```
         AOPTN CROSSREF                                                 A2100010
A21B     TITLE 'ABSOLUTE LOADER FOR CURRENT SYSTEMS SIMULATORS'         A2100020
*************************************************************************A2100030
*                                                                       * A2100040
*                    ABSOLUTE  LOADER  PROGRAM                          * A2100050
*                                                                       * A2100060
*                              FOR                                      * A2100070
*                                                                       * A2100080
*               IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620               * A2100090
*                                                                       * A2100100
*               ----------------------------------------                * A2100110
*                                                                       * A2100120
*                           INTRODUCTION                                * A2100130
*                                                                       * A2100140
*  THIS PROGRAM LOADS THE ASSEMBLED PROGRAMS  INTO  SYSTEM/360  MAIN    * A2100150
*  STORAGE AT THE ADDRESSES ASSIGNED BY THE ASSEMBLER.  THEN,  CONT-    * A2100160
*  ROL IS TRANSFERRED TO A GIVEN SECTION FOR EXECUTION OF  THE  PRO-    * A2100170
*  GRAM LOADED.                                                         * A2100180
*                                                                       * A2100190
*  THE ABSOLUTE LOADER ALSO ALLOWS THE USER TO MAKE CERTAIN ADDITIONS   * A2100200
*  OR CORRECTIONS TO THE SIMULATOR AT THE TIME OF LOADING.              * A2100210
*                                                                       * A2100220
*  THE ABSOLUTE LOADER USES TWO CONTIGUOUS BUFFERS OF 120 BYTES  AND    * A2100230
*  80 BYTES RESPECTIVELY.  THESE BUFFERS PRECEDE THE ABSOLUTE LOADER    * A2100240
*  PROGRAM.  AT THE TIME OF LOADING, THE FIRST BUFFER  RECEIVES  THE    * A2100250
*  DATA TO BE LOADED AT ADDRESSES 0 TO 128. THIS BUFFER  REPRESENTS     * A2100260
*  THE PSW ZONE.  THE SECOND BUFFER IS AN INPUT BUFFER WHICH  PARTLY    * A2100270
*  OVERLAYS THE AREA OF THE INITIALIZATION ROUTINE.                     * A2100280
*                                                                       * A2100290
*  THIS LOADER IS SPECIALLY DESIGNED TO LOAD THE FOLLOWING PROGRAMS=    * A2100300
*                                                                       * A2100310
*               CONTROL PROGRAM                                         * A2100320
*               I/O SUPPORT PACKAGE PROGRAM                             * A2100330
*               SYSTEM/360 INITIALIZATION PROGRAM                       * A2100340
*               RELOCATING LOADER PROGRAM                               * A2100350
*  AN LDR CARD IS PLACED BETWEEN THE END CARD  OF  THE  I/O SUPPORT     * A2100360
*  PACKAGE PROGRAM AND THE  1ST  CARD OF THE INITIALIZATION PROGRAM.    * A2100370
*  THE LAST CARD IN THE DECK IS AN LDT CARD.                            * A2100380
*                                                                       * A2100390
*  CONTROL IS GIVEN TO THE ABSOLUTE LOADER AT THE  END  OF  THE  IPL    * A2100400
*  PROCEDURE. THE CPU PROCEEDS UNDER CONTROL OF THE PSW FETCHED FROM    * A2100410
*  LOCATION 0. THIS PSW IS DISABLED FOR ALL INTERRUTIONS EXCEPT  MA-    * A2100420
*  CHINE CHECK INTERRUPTS. ITS ADDRESS FIELD CONTAINS THE ADDRESS OF    * A2100430
*  ENTRY POINT TO THE ABSOLUTE LOADER.                                  * A2100440
*                                                                       * A2100450
*  THE ABSOLUTE LOADER USES A THIRD BUFFER OF 8 BYTES,  IN  LOCATION    * A2100460
*  LDBUFF, TO SAVE THE FOLLOWING DATA                                   * A2100470
*        ADDRESS OF THE LAST BYTE OF THE CONTROL PROGRAM, WHICH  IS     * A2100480
*              THE CONTENTS OF LOCATION COUNTER AT THE TIME THE 1ST     * A2100490
*              END CARD IS ENCOUNTERED.                                 * A2100500
*        ADDRESS OF THE ENTRY POINT TO THE RELOCATING LOADER, WHICH     * A2100510
*              IS THE CONTENTS OF ADDRESS  FIELD  IN THE  LAST  END     * A2100520
*              CARD ENCOUNTERED.                                        * A2100530
*                                                                       * A2100540
*                                                        .../...        * A2100550
```

SOURCE SUPPLIED ON OPTIONAL MATERIAL TAF

1620

extra

extra

extra

```
*                                                                     * A2100560
*********************************************************************** A2100570
*********************************************************************** A2100580
*                                                                     * A2100590
*  .../...                                                            * A2100600
*                                                                     * A2100610
*  THE LOADER TRANSFERS THESE DATA AND THE CURRENT VALUE OF LOCATION  * A2100620
*  COUNTER TO THE SYSTEM/360 INITIALIZATION PROGRAM.                  * A2100630
*  NOTE- LOCCTR CONTAINS THE ADDRESS OF THE LAST BYTE OF THE I/O      * A2100640
*        SUPPORT PACKAGE PROGRAM.                                     * A2100650
*                                                                     * A2100660
*  SINCE THE ABSOLUTE LOADER IS NOT A SELF-LOADING PROGRAM, THE LOAD- * A2100670
*  ING FUNCTION IS OBTAINED BY INSERTING ONE PROGRAM LOADING CARD IN  * A2100680
*  THE CARD DECK, AS DESCRIBED IN THE FOLLOWING.                      * A2100690
*                                                                     * A2100700
*********************************************************************** A2100710
        SPACE                                                          A2100720
*********************************************************************** A2100730
*                                                                     * A2100740
*                        SELF LOADING DECK STRUCTURE                  * A2100750
*                                                                     * A2100760
*  IN ORDER TO EXECUTE THE LOADING FUNCTION, THE CARD DECK OF THE AB- * A2100770
*  SOLUTE LOADER MUST CONTAIN THE FOLLOWING CARDS =                   * A2100780
*                                                                     * A2100790
*      - 1 INITIAL PROGRAM LOADING CARD  - IPL                        * A2100800
*      - N TEXT CARDS - THESE CARDS ARE TAKEN FROM THE OBJECT   DECK  * A2100810
*      - 1 RLD  CARD  - OF THE ABSOLUTE LOADER.                       * A2100820
*      - 1 END  CARD  -                                               * A2100830
*                                                                     * A2100840
*      THE IPL CARD IS PRODUCED AS OUTPUT BY THE BOS/360    ASSEMBLY * A2100850
*      PROGRAM AND HAS THE FOLLOWING FORMAT.                          * A2100860
*                                                                     * A2100870
*      COL 1 -8 '00   04   00   00   00   A( ABSLOD  )' IPL  PSW      * A2100880
*      COL 9 -16'02   A(ABSLOD-16)   60   00   00   50' READ CCW      * A2100890
*      COL 17-24'02   A(  BUFF   )   20   00   00   50' READ CCW      * A2100900
*      COL 65-72'40   40   40   40   C9   D7   D3   40' IPL  ID.      * A2100910
*      COL 73-80'C1   F2   F1   C2   F0   F0   F0   F1' A21B0001      * A2100920
*                                                                     * A2100930
*  THE SELF LOADING DECK   IS  OBTAINED  BY REMOVING FROM THE OBJECT  * A2100940
*  DECK  THE CARD  PRECEDING  THE CARD  WITH IDENTIFICATION  IPL  IN  * A2100950
*  COLUMNS  69-71.                                                    * A2100960
*                                                                     * A2100970
*********************************************************************** A2100980
        EJECT                                                          A2100990
ABSLOD  START 26960                    *                               A2101000
BUFF    EQU   ABSLOD+256               INPUT BUFFER USED TO LOAD       A2101010
*                                      * ABSOLUTE LOADER PROGRAM       A2101020
        SPACE                                                          A2101030
*********************************************************************** A2101040
*                                                                     * A2101050
*  3  STATEMENTS  FOLLOW WHICH DO NOT  APPEAR IN THE SOURCE LISTING.  * A2101060
*  THE FORMAT OF THESE INSTRUCTION STATEMENTS IS =                    * A2101070
*                                                                     * A2101080
*  NAME  OPERATION  OPERAND                                COL. 71    * A2101090
*  ----  ---------  -------                                -------    * A2101100
```

```
*                                                                     * A2101110
*        PRINT      OFF                                               * A2101120
*                                                                     * A2101130
*        PUNCH      'CONTENTS OF IPL  CARD COLUMNS  1-55        X     * A2101140
*                   CONTENTS  OF IPL  CARD COLUMNS 56-80'             * A2101150
*                                                                     * A2101160
*  THE PUNCH ASSEMBLER INSTRUCTION CAUSES THE DATA IN THE OPERAND TO  * A2101170
*  BE PUNCHED IN A CARD. THE OPERAND IS WRITTEN  AS A STRING  OF  80  * A2101180
*  CHARACTERS.  THE  POSITION  IMMEDIATELY  TO  THE  RIGHT  OF   THE  * A2101190
*  QUOTATION MARK IS REGARDED AS COLUMN 1 OF THE CARD TO BE PUNCHED.  * A2101200
*                                                                     * A2101210
*  NOTE- THE LOCATIONS SPECIFIED BY THE OPERAND OF THE PUNCH  STATE-  * A2101220
*        MENT  DEPENDS ON THE ASSEMBLY  LOCATION  SPECIFIED  BY  THE  * A2101230
*        OPERAND OF THE START STATEMENT.                              * A2101240
*                                                                     * A2101250
*********************************************************************** A2101260
         SPACE                                                        A2101270
         PRINT OFF                                                    A2101280
         PUNCH '       &    -  &   &   &                            XA2101290
                            IPL A21B0001'                             A2101300
         PRINT ON                                                     A2101310
         SPACE                                                        A2101320
*********************************************************************** A2101330
*                                                                     * A2101340
*                 DEFINITION OF PROGRAM STATUS WORDS                  * A2101350
*                                                                     * A2101360
*********************************************************************** A2101370
         SPACE                                                        A2101380
PGOPSW   EQU   40                    PROGRAM OLD PSW                 A2101390
IOOPSW   EQU   56                    I/O OLD PSW                     A2101400
LDRCSW   EQU   64                    LOADER CSW                      A2101410
LDRCAW   EQU   72                    LOADER CAW                      A2101420
EXNPSW   EQU   88                    EXTERNAL NEW PSW                A2101430
PGNPSW   EQU   104                   PROGRAM NEW PSW                 A2101440
MKNPSW   EQU   112                   MACHINE CHECK NEW PSW           A2101450
IONPSW   EQU   120                   I/O NEW PSW                     A2101460
         EJECT                                                        A2101470
*********************************************************************** A2101480
*                                                                     * A2101490
*              ROUTINE TO LOAD THE ABSOLUTE LOADER                    * A2101500
*                                                                     * A2101510
*  ENTRY POINT= ABSLOD                                                * A2101520
*  CONTROL IS GIVEN TO THIS ROUTINE AT THE END OF THE IPL PROCEDURE.  * A2101530
*  AT THAT TIME= THE FIRST 56 BYTES ARE PLACED IN STORE  (IPL CCW 1)  * A2101540
*                THE CONTENTS OF SECOND TXT CARD IS STORED  IN  ONE   * A2101550
*                INPUT BUFFER (BUFF). (IPL CCW 2).                    * A2101560
*  THE ROUTINE = 1. MOVE THE 56 BYTES OF TEXT FROM  BUFF TO STORAGE.  * A2101570
*                2. READ ONE CARD OR CARD IMAGE AND, UPON FINDING  A  * A2101580
*                   TXT, END OR OTHER CARD  PERFORMS OPERATION 1, 3,  * A2101590
*                   OR 2.                                            * A2101600
*                3. TERMINATES THE LOADING PROCESS AND BRANCHES   TO  * A2101610
*                   LOCATION LENTRY  FOR EXECUTION OF  THE  ABSOLUTE  * A2101620
*                   LOADER PROGRAM.                                  * A2101630
*  AFTERWARDS, THE STORAGE AREA OCCUPIED BY THIS ROUTINE IS OVERLAID   * A2101640
*                                                                     * A2101650
```

```
*********************************************************************** A2101660
         SPACE                                                          A2101670
         BALR  LBASRG,0              *                                  A2101680
         USING *,LBASRG              *                                  A2101690
IPLCTL   LH    LWK3RG,BUFF+10        BYTE COUNT                         A2101700
         BCTR  LWK3RG,0              *                                  A2101710
         STC   LWK3RG,IPLMVE+1       *                                  A2101720
         L     LOCCTR,BUFF+4         TEXT ADDRESS                       A2101730
IPLMVE   MVC   0(1,LOCCTR),BUFF+16   MOVE TEXT FROM CARD TO STORE       A2101740
         LA    LWK1RG,IPLCCW         READ CARD(IMAGE) CCW               A2101750
         ST    LWK1RG,LDRCAW         *                                  A2101760
         LH    LWK2RG,2              PICK-UP IPL DEVICE ADDRESS         A2101770
SIO      SIO   0(LWK2RG)             SIO INPUT DEVICE                   A2101780
         BC    6,SIO                 CC= 1,2. BUSY OR CSW STORED        A2101790
         BC    1,WAIT                CC= 3. ERROR, ENTER WAIT STATE.    A2101800
BUSY     TIO   0(LWK2RG)             CC=0,OPERATION STARTED.            A2101810
         BC    2,BUSY                *                                  A2101820
         BC    9,WAIT                *                                  A2101830
         TM    LDRCSW+4,X'04'        CSW STORED. IS DE PRESENT          A2101840
         BC    8,BUSY                NO, BRANCH. YES,                   A2101850
         TM    LDRCSW+5,X'BF'        IS ANY CHANNEL ERROR CONDITION     A2101860
         BC    5,WAIT                * PRESENT. YES,ERROR,WAIT.  NO,    A2101870
         TM    LDRCSW+4,X'83'        IS ATTENTION,UC OR UE PRESENT      A2101880
         BC    5,WAIT                * YES,ERROR,WAIT.                  A2101890
         CLC   BUFF(4),TXT           IS CARD READ A TXT CARD.           A2101900
         BC    8,IPLCTL              * YES, GO TO PROCESS IT.           A2101910
         BC    15,LDEND              * NO, BRANCH                       A2101920
IPLCCW   CCW   X'02',BUFF,X'20',80   * READ CCW (80 BYTES)              A2101930
TXT      DC    X'02E3E7E3'           TXT CARD TYPE                      A2101940
WAIT     MVI   1,X'02'               SET WAIT STATE BIT ON IN IPL PSW   A2101950
         LPSW  0                     * AND LOAD IT.                     A2101960
LDEND    CLC   BUFF(4),END           IS CARD READ THE ABSOLUTE LOADER   A2101970
         BC    8,LENTRY              * END CARD. YES, BRANCH TO LOADE   A2101980
         BC    15,SIO                NO, IGNORE CARD AND GET NEXT ONE   A2101990
END      DC    X'02C5D5C4'           END CARD TYPE                      A2102000
         ORG   ABSLOD+336            *                                  A2102010
         EJECT                                                          A2102020
*********************************************************************** A2102030
*                                                                     * A2102040
*                        INITIALIZATION ROUTINE                       * A2102050
*                                                                     * A2102060
*  ENTRY POINT = 'LENTRY'                                             * A2102070
*  THIS ROUTINE SETS PARAMETERS FOR MACHINE-CHECK AND PROGRAM INTER-  * A2102080
*  RUPTIONS.                                                          * A2102090
*  WHEN A MACHINE-CHECK INTERRUPTION OCCURS, A PSW FOR WHICH I/O,EX-  * A2102100
*  TERNAL AND FURTHER MACHINE CHECK INTERRUPTIONS ARE DISABLED,  AND  * A2102110
*  IN WHICH THE WAIT STATE BIT IS ONE, WILL BE LOADED.  THE  ADDRESS  * A2102120
*  FIELD OF THIS PSW WILL CONTAIN ALL ZEROS.                          * A2102130
*                                                                     * A2102140
*  PROGRAM INTERRUPTION PROCESSING                                    * A2102150
*                                                                     * A2102160
*  WHEN THE FIRST PROGRAM INTERRUPTION OCCURS A NEW PROGRAM PSW WITH  * A2102170
*  I/O AND EXTERNAL INTERRUPTIONS DISABLED WILL  BE LOADED.  CONTROL  * A2102180
*  WILL RETURN TO THE INSTRUCTION WITH ADDRESS CLEAR3.  THE  ROUTINE  * A2102190
*  WILL SET THE WAIT STATE BIT ON IN THE PROGRAM NEW PSW. ANY SUBSE-  * A2102200
```

```
*  QUENT PROGRAM INTERRUPTION WILL CAUSE THE  SYSTEM  TO ENTER   THE  * A2102210
*  WAIT STATE, ALL INTERRUPTS, EXCEPT MACHINE-CHECK, DISABLED.        * A2102220
*  IF THE FIRST PROGRAM INTERRUPTION RESULTS FROM AN ADDRESSING  EX-  * A2102230
*  CEPTION (WORD OUTSIDE THE AVAILABLE STORAGE  FOR  THE  PARTICULAR  * A2102240
*  INSTALLATION) CONTROL WILL RETURN TO THE INSTRUCTION WITH ADDRESS  * A2102250
*  CLEAR4. OTHERWISE THE PROGRAM NEW PSW WILL BE LOADED.              * A2102260
*                                                                     * A2102270
*  THE ROUTINE  ADJUSTS  THE  READ  ROUTINE  TO  LOAD FROM THE INPUT  * A2102280
*  DEVICE USED DURING THE LOADER IPL PROCEDURE.                       * A2102290
*                                                                     * A2102300
*  FINALLY  THE ROUTINE SETS CORE STORAGE TO ZERO  FROM LOCATION 180  * A2102310
*  TO THE END OF STORAGE AVAILABLE EXCEPT FOR THE AREA  OCCUPIED  BY  * A2102320
*  THE ABSOLUTE LOADER PROGRAM.                                       * A2102330
*                                                                     * A2102340
*  THIS ROUTINE IS ENTERED ONCE DURING THE EXECUTION OF  THE  LOADER  * A2102350
*  PROGRAM. THE STORAGE AREA OCCUPIED BY THIS ROUTINE IS OVERLAID BY  * A2102360
*  THE INPUT BUFFER TO SAVE STORAGE SPACE.                            * A2102370
*                                                                     * A2102380
********************************************************************** A2102390
         SPACE                                                        A2102400
         USING GETCRD-208,LBASRG       *                              A2102410
LENTRY   BALR  LBASRG,0                *                              A2102420
LBEGIN   L     LBASRG,ALALFA-LBEGIN(0,LBASRG) LOAD BASE REGISTER      A2102430
         XC    EXNPSW(40),EXNPSW       CLEAR NEW PSW AREA             A2102440
         OI    MKNPSW+1,X'06'          SET WAIT-MACH CHECK BITS ON    A2102450
         MVC   PGNPSW(8),PRGPSW        BUILD UP PROGRAM NEW PSW       A2102460
         LA    LOCCTR,INITAD           INITIALIZE LOCATION COUNTER    A2102470
         SPACE 1                                                      A2102480
LENTRE   MVC   SIORDR+2(2),2           PICK UP IPL DEVICE ADDRESS     A2102490
         SPACE 1                                                      A2102500
CLEAR0   LA    LWK1RG,INITAD           STARTING AND ENDING ADDRESSES  A2102510
         L     LWK2RG,ENDAD            OF FIRST AREA TO BE CLEARED    A2102520
CLEAR1   CLR   LWK1RG,LWK2RG           IS AREA CLEARED                A2102530
         BC    2,CLEAR2                YES,BRANCH TO CLEAR2           A2102540
CLEAR    XC    0(256,LWK1RG),0(LWK1RG) NO,CLEAR 256 BYTES AT A TIME   A2102550
         LA    LWK1RG,256(LWK1RG)      INCREMENT STARTING ADDR. BY 256 A2102560
         BC    15,CLEAR1               AND RESUME PROCESSING          A2102570
CLEAR2   LA    LWK2RG,CLEAR0           CALCULATE  LENGTH   OF          A2102580
         SR    LWK2RG,LWK1RG           SECOND AREA TO BE CLEARED AND  A2102590
         STC   LWK2RG,CLEAR4+1         STORE  IT  AND                 A2102600
         LR    LWK2RG,LWK1RG           STARTING ADDRESS               A2102610
         LA    LWK1RG,LDREND           STARTING ADDR. OF THIRD AREA TO A2102620
         MVI   CLEAR1+3,X'00'          BE CLEARED(BETWEEN LOADER AND  A2102630
         BC    15,CLEAR1               END OF STORAGE AVAILABLE)      A2102640
CLEAR3   OI    PGNPSW+1,X'02'          SET WAIT BIT FOR NEXT PR.INT. ON A2102650
         CLI   PGOPSW+3,X'05'          TEST IF ADDRESSING INTERRUPT.  A2102660
         BC    8,CLEAR4                YES,BRANCH TO CLEAR4           A2102670
         LPSW  PGNPSW                  NO,LOAD PROGRAM NEW PSW        A2102680
CLEAR4   XC    0(0,LWK2RG),0(LWK2RG)   CLEAR SECOND AREA              A2102690
         SPACE 2                                                      A2102700
********************************************************************** A2102710
*                                                                    * A2102720
*                      CARD ANALYSIS  ROUTINE                        * A2102730
*                                                                    * A2102740
*  ENTRY POINT = 'GETCRD'                                            * A2102750
```

```
*  THIS ROUTINE SELECTS THE APPROPRIATE ROUTINE TO PROCESS THE  CARD  * A2102760
*  WHICH HAS BEEN READ.                                               * A2102770
*  THIS ROUTINE OBTAINS CARD IMAGES THROUGH THE  I/O ROUTINE  LDREAD  * A2102780
*  AND,UPON FINDING A TXT,REP,END,LDR OR LDT CARD IMAGE,LINKS,TO THE  * A2102790
*  TXT,REP,END,LDR OR LDT CARD ROUTINE. ALL OTHER CARDS ARE IGNORED.  * A2102800
*  THE DIFFERENT CARD TYPES ARE TESTED IN ORDER OF DECREASING   FRE-  * A2102810
*  QUENCY.                                                            * A2102820
*  THE ROUTINE MAY BE ENTERED FROM THE INITIALIZATION ROUTINE,  FROM  * A2102830
*  ITSELF AND FROM THE TXT,REP,LDR AND END CARD ROUTINES.             * A2102840
*                                                                     * A2102850
********************************************************************** A2102860
         SPACE                                                         A2102870
         CNOP  0,4                                                     A2102880
GETCRD   BAL   LEXIT3,LDREAD          GET CARD IMAGE                   A2102890
         L     LWK1RG,LDXBUF          CARD TYPE (COL.1-4) TO WORK.REG. A2102900
         CL    LWK1RG,LTXTCD          IS CARD TYPE = TXT               A2102910
         BC    8,LENTXT               YES,BRANCH TO TXT CARD ROUTINE   A2102920
         CL    LWK1RG,LREPCD          NO,IS CARD TYPE = REP            A2102930
         BC    8,LENREP               YES,BRANCH TO REP CARD ROUTINE   A2102940
         CL    LWK1RG,LENDCD          NO,IS CARD TYPE = END            A2102950
         BC    8,LENEND               YES,BRANCH TO END CARD ROUTINE   A2102960
         CL    LWK1RG,LLDRCD          NO,IS CARD TYPE = LDR            A2102970
         BC    8,LENLDR               YES,BRANCH TO LDR CARD ROUTINE   A2102980
         CL    LWK1RG,LLDTCD          NO,IS CARD TYPE  = LDT           A2102990
         BC    6,GETCRD               IGNORE OTHER CARDS               A2103000
         EJECT                                                         A2103010
********************************************************************** A2103020
*                                                                    * A2103030
*                    LDT CARD PROCESSING  ROUTINE                    * A2103040
*                                                                    * A2103050
*  ENTRY POINT = 'LENLDT'                                            * A2103060
*  THIS ROUTINE TERMINATES THE LOADING PROCESS. THE  SEQUENCE  FIRST  * A2103070
*  LOOKS FOR  AN ENTRY POINT TO THE PROGRAM.  THEN, PROGRAM DATA ARE  * A2103080
*  SAVED AND CONTROL IS GIVEN TO THE ENTRY POINT FOR EXECUTION OF THE * A2103090
*  PROGRAM BY LOADING A PSW IMAGE WHICH CONTAINS  THE ENTRY  ADDRESS. * A2103100
*  THAT PSW IMAGE BECOMES THE CURRENT PSW.WHEN CONTROL IS TRANSFERRED * A2103110
*  THE SYSTEM IS IN PROBLEM STATE. ALL INTERRUPTS EXCEPT MACHINE AND  * A2103120
*  PROGRAM CHECK ARE DISABLED.                                       * A2103130
*  THE ROUTINE IS ENTERED FROM THE CARD ANALYZIS ROUTINE.            * A2103140
*                                                                    * A2103150
********************************************************************** A2103160
         SPACE                                                         A2103170
         CLI   LDXBUF+5,X'40'         DOES LDT  CARD CONTAIN  ADDRESS A2103180
         BC    8,LENTRL               NO,GO TO SAVE DATA              A2103190
LENLDT   MVC   LDTPSW+5(3),LDXBUF+5   YES,PLACE ADDRESS IN TRANSF.PSW A2103200
LENTRL   LM    2,3,LDBUFF             GL. REGISTERS 1 TO 3 CONTAIN    A2103210
         LA    1,1(LOCCTR)            DATA FOR LOADED PROGRAMS        A2103220
         L     4,ALALFA               MOVE PSW IMAGE ZONE             A2103230
         MVC   24(104,0),24(4)        TO  PSW  AREA                   A2103240
         LPSW  LDTPSW                 END OF LOADING - TRANSFER       A2103250
*                                     CONTROL TO LOADED PROGRAM       A2103260
         SPACE 2                                                       A2103270
********************************************************************** A2103280
*                                                                    * A2103290
*                    TXT CARD PROCESSING ROUTINE                     * A2103300
```

```
*                                                                     * A2103310
*  ENTRY POINT = 'LENTXT'                                             * A2103320
*  THIS ROUTINE TESTS THE CARD-SPECIFIED TEXT ADDRESS.                * A2103330
*                                                                     * A2103340
*  IF THIS ADDRESS IS LOWER THAN 128,  THE PART OF THE TEXT CARD  TO  * A2103350
*  BE LOADED BEFORE LOCATION 128 IS PLACED INTO A BUFFER STARTING AT  * A2103360
*  ADDRESS LALPHA. THE OTHER PART OF THE TEXT CARD IS STORED STARTING * A2103370
*  FROM LOCATION 128.                                                 * A2103380
*                                                                     * A2103390
*  IF THE STARTING ADDRESS IS GREATER THAN OR EQUAL TO 128, THE ROU-  * A2103400
*  TINE CHECKS IF THE TEXT OVERLAYS THE LOADER PROGRAM.  IF THERE IS  * A2103410
*  NO OVERLAY, THE TEXT IS PLACED INTO MAIN STORAGE AND THE LOCATION  * A2103420
*  COUNTER IS UPDATED.  IT ALWAYS CONTAINS THE HIGHEST ADDRESS OCCU-  * A2103430
*  PLED. IN CASE OF OVERLAY, AN ERROR WAIT PSW IS LOADED  WITH   THE  * A2103440
*  ERROR INDICATION '2' IN ITS ADDRESS FIELD.                         * A2103450
*  THIS ROUTINE HAS TWO ENTRY POINTSO LOCATION  LENTXT  AND LOCATION  * A2103460
*  LENTX1. LOCATION LENTXT IS THE ENTRY POINT FROM THE CARD  ANALYSIS * A2103470
*  ROUTINE. LOCATION LENTX1 IS THE ENTRY  POINT  FROM  THE REP  CARD  * A2103480
*  ROUTINE.                                                           * A2103490
*                                                                     * A2103500
*  NOTE = WHEN THE LDR CARD HAS BEEN ENCOUNTERED, THE LOCATION COUN-  * A2103510
*  TER WILL NOT BE UPDATED  AND  THE ADDRESS INSPECTION  FOR  LOADER  * A2103520
*  OVERLAY WILL NOT BE PERFORMED.                                     * A2103530
*                                                                     * A2103540
*********************************************************************** A2103550
         SPACE                                                         A2103560
LENTXT   NI    LDXBUF+4,X'00'          TXT CARD - GET STARTING ADDRESS A2103570
         L     LWK1RG,LDXBUF+4         AND BYTE COUNT                  A2103580
         LH    LWK2RG,LDXBUF+10        *                               A2103590
LENTX1   LTR   LWK2RG,LWK2RG           IS BYTE COUNT EQUAL TO ZERO     A2103600
         BC    8,GETCRD                YES,GET NEXT CARD               A2103610
         BCTR  LWK2RG,0                NO,DECREMENT WORK.REG. BY 1     A2103620
         STC   LWK2RG,LODTXT+1         *                               A2103630
         AR    LWK2RG,LWK1RG           CALCULATE HIGHEST ADDRESS TEXT  A2103640
*                                      WILL OCCUPY                     A2103650
         LH    LWK3RG,PSWZON           LOAD 128 (END OF PSW  AREA)     A2103660
         SR    LWK3RG,LWK1RG           IS START.ADDRESS GREATER THAN OR A2103670
*                                      EQUAL TO 128                    A2103680
TSTADR   BC    12,LENTRM               YES,GO TO TEST ADDRESS          A2103690
         SH    LWK2RG,PSWZON           NO,IS HIGHEST ADDR.LESS THAN 128 A2103700
         BC    4,LENTRN                YES,BRANCH TO LENTRN            A2103710
         BCTR  LWK3RG,0                NO,CALCULATE AND STORE TEXT     A2103720
         STC   LWK3RG,LODTXT+1         LENGTH TO BE LOADED  IN PSW     A2103730
*                                      IMAGE ZONE                      A2103740
         STC   LWK2RG,MOVTXT+1         STORE TEXT LENGTH TO BE LOADED  A2103750
*                                      AT ADDRESS 128                  A2103760
         LA    LWK3RG,LDXBUF+17(LWK3RG) *LOAD ADDRESS OF TEXT IN INPUT  A2103770
*                                      BUFFER                          A2103780
MOVTXT   MVC   128(1,0),0(LWK3RG)      PLACE TEXT IN STORAGE           A2103790
LENTRN   A     LWK1RG,ALALFA           ADDRESS OF IMAGE ZONE FOR FIRST A2103800
*                                      PART OF TEXT                    A2103810
LODTXT   MVC   0(1,LWK1RG),LDXBUF+16   TRANSFER TEXT FROM CARD TO STOR. A2103820
         BC    15,GETCRD               BRANCH TO CARD ANALYSIS ROUTINE A2103830
LENTRM   CLR   LWK2RG,LOCCTR           IS HIGHEST ADDRESS GREATER THAN A2103840
*                                      LOCCTR                          A2103850
```

```
CHANGE   BC    12,LODTXT              NO,BRANCH TO  LODTXT              A2103860
         LR    LOCCTR,LWK2RG          YES,SAVE ADDRESS IN LOCCTR       A2103870
         CL    LWK2RG,ALALFA          IS LOADER OVERLAID               A2103880
         BC    4,LODTXT               NO, TRANSFER TEXT TO STORAGE     A2103890
         MVI   STOPSW+7,X'02'         YES,BRANCH TO ERROR WAIT-ERR2    A2103900
LDSTOP   LPSW  STOPSW                 *                                A2103910
         EJECT                                                          A2103920
*********************************************************************** A2103930
*                                                                     * A2103940
*                     REP CARD PROCESSING ROUTINE                     * A2103950
*                                                                     * A2103960
*  ENTRY POINT = 'LENREP'                                             * A2103970
*  THIS ROUTINE CONVERTS THE REP CARD TO THE FORMAT OF A  TXT  CARD.  * A2103980
*  THE ROUTINE LINKS TO THE LHEXB  ROUTINE TO CONVERT  BOTH  THE AD-  * A2103990
*  DRESS AND THE TEXT OF THE CORRECTIONS FROM HEXADECIMAL TO  BINARY  * A2104000
*  FORM. CONTROL IS THEN GIVEN TO THE TXT CARD PROCESSING ROUTINE.    * A2104010
*  THIS ROUTINE  HAS  ONE ENTRY POINT, LOCATION  LENREP,  WHICH   IS  * A2104020
*  ALWAYS ENTERED FROM THE CARD ANALYSIS ROUTINE.                     * A2104030
*  DURING THE PERIODS THE REP CARD ROUTINE IS LINKED TO  THE  LHEXB1  * A2104040
*  ROUTINE, IT IS SUBJECT TO THE ERROR WAIT OF THAT ROUTINE. IF  THE  * A2104050
*  REP CARD IS INCORRECTLY DEFINED IT ALSO EXITS  TO  THE SAME ERROR  * A2104060
*  WAIT (LOCATION LHEXB5).                                            * A2104070
*                                                                     * A2104080
*********************************************************************** A2104090
         SPACE 1                                                        A2104100
LENREP   LA    LEXIT1,LDXBUF+6        CONVERT REP CARD-SPECIFIED       A2104110
         LA    LEXIT2,6               CORRECTION ADDRESS TO BINARY     A2104120
         BAL   LEXIT3,LHEXB1          *                                A2104130
         LR    LWK1RG,LEXIT4          AND STORE IT IN WORK.REG. 1      A2104140
         LA    LEXIT1,LDXBUF+16       ADDR.OF FIRST CORRECTION BYTE    A2104150
         CLI   0(LEXIT1),X'40'        IS FIRST BYTE BLANK              A2104160
         BC    8,GETCRD               YES,BRANCH TO CARD ANALYSIS ROUT A2104170
LENRE1   LA    LWK2RG,2               NO,UPDATE BYTE COUNT REGISTER    A2104180
LENRE2   LA    LEXIT2,4               CONVERT 4 CORRECTION BYTES       A2104190
         BAL   LEXIT3,LHEXB1          TO HEXADECIMAL                   A2104200
         STH   LEXIT4,LDXBUF+14(LWK2RG) AND STORE IN INPUT BUFFER      A2104210
         CLI   0(LEXIT1),C','         IS NEXT BYTE A COMMA             A2104220
         BC    7,LENRE3               NO,BRANCH  TO TEST FOR BLANK     A2104230
         LA    LEXIT1,1(LEXIT1)       YES,UPDATE CORRECTION ADDRESS    A2104240
         AH    LWK2RG,LENRE1+2        INCREMENT BYTE COUNT BY 2        A2104250
         BC    15,LENRE2              RETURN TO  LENRE2                A2104260
LENRE3   CLI   0(LEXIT1),X'40'        IS NEXT BYTE BLANK               A2104270
         BC    7,LHEXB5               NO,INCORRECT REP CARD            A2104280
         BC    15,LENTX1              YES,BRANCH TO TXT CARD PROCESS.  A2104290
         EJECT                                                          A2104300
*********************************************************************** A2104310
*                                                                     * A2104320
*                     LDR CARD PROCESSING  ROUTINE                    * A2104330
*                                                                     * A2104340
*  ENTRY POINT = 'LENLDR'                                             * A2104350
*  THIS ROUTINE STOPS INCREMENTING OF THE LOCATION COUNTER  'LOCCTR'  * A2104360
*  BEFORE GIVING CONTROL BACK TO THE CARD ANALYSIS ROUTINE.           * A2104370
*                                                                     * A2104380
*********************************************************************** A2104390
         SPACE                                                          A2104400
```

```
LENLDR   MVC   TSTADR(4),CHANGE       MODIFY TXT CARD PROCESSING ROUT. A2104410
         BC    15,GETCRD              BRANCH TO CARD ANALYSIS ROUTINE  A2104420
         SPACE 2                                                       A2104430
*************************************************************************A2104440
*                                                                      * A2104450
*                     END CARD PROCESSING  ROUTINE                     * A2104460
*                                                                      * A2104470
*  ENTRY POINT = 'LENEND'                                              * A2104480
*  THIS ROUTINE SAVES THE CONTENTS OF THE LOCATION  COUNTER  AT  THE   * A2104490
*  TIME THE FIRST END CARD IS ENCOUNTERED. IT ALSO SAVES THE ADDRESS   * A2104500
*  FIELD OF THE LAST END CARD ENCOUNTERED. THEN,IT DETERMINES AN EN-   * A2104510
*  TRY POINT FOR PROGRAM EXECUTION.                                    * A2104520
*  THIS ROUTINE IS ALWAYS ENTERED FROM AND EXITS TO THE CARD  ANALY-   * A2104530
*  SIS ROUTINE.                                                        * A2104540
*                                                                      * A2104550
*************************************************************************A2104560
         SPACE 1                                                       A2104570
LENEND   TM    CPREND,X'20'           IS FIRST CARD END SWITCH OFF     A2104580
         BC    1,LENEN1               NO,BRANCH TO LENEN1              A2104590
         ST    LOCCTR,LDBUFF          YES,STORE ADDRESS FIELD          A2104600
         OI    CPREND,X'20'           SET FIRST CARD END SWITCH ON     A2104610
LENEN1   MVC   LDBUFF+5(3),LDXBUF+5   STORE ADDRESS FIELD              A2104620
         TM    LDENTR,X'10'           HAS AN ENTRY POINT BEEN FOUND    A2104630
         BC    1,GETCRD               YES,GO TO CARD ANALYSIS ROUTINE  A2104640
         CLI   LDXBUF+5,X'40'         NO,TEST IF ADDRESS IN CARD IS    A2104650
*                                      BLANK                           A2104660
         BC    8,GETCRD               YES,GO TO CARD ANALYSIS ROUTINE  A2104670
         OI    LDENTR,X'10'           SET SWITCH ENTRY POINT FOUND ON  A2104680
         EX    0,LENLDT               PLACE ADDRESS IN TRANSFER PSW    A2104690
         BC    15,GETCRD              BRANCH TO CARD ANALYSIS ROUTINE  A2104700
         EJECT                                                         A2104710
*************************************************************************A2104720
*                                                                      * A2104730
*                     READ TAPE AND CARDS  ROUTINE                     * A2104740
*                                                                      * A2104750
*  ENTRY POINT = 'LDREAD'                                              * A2104760
*  THIS ROUTINE READS 80 BYTES FROM CARDS OR MAGNETIC  TAPE  RECORDS   * A2104770
*  AND TESTS THE CHANNEL AND DEVICE STATUS BITS.  IF DATA HAVE  BEEN   * A2104780
*  READ SUCCESSFULLY, CONTROL IS GIVEN BACK TO THE CALLING SEQUENCE.   * A2104790
*  OTHERWISE, AN ERROR WAIT PSW IS LOADED AND THE LOADING PROCESS IS   * A2104800
*  TERMINATED WITHOUT ANY POSSIBILITY OF RETRY. THE ERROR INDICATION   * A2104810
*  '1' IS PLACED IN THE ADDRESS FIELD OF THE ERROR WAIT PSW.           * A2104820
*                                                                      * A2104830
*************************************************************************A2104840
         SPACE 1                                                       A2104850
LDREAD   SSM   STOPSW                 DISABLE I/O EXT.INTERRUPTIONS    A2104860
         LA    LWK1RG,LRDCCW          LOAD ADDRESS OF CCW              A2104870
         ST    LWK1RG,LDRCAW          AND STORE IT IN CAW              A2104880
         LA    LWK1RG,RDIRUP          SET UP NEW I/O PSW FOR I/O       A2104890
         ST    LWK1RG,IONPSW+4        INTERRUPTIONS                    A2104900
SIORDR   SIO   0                      SIO READER                       A2104910
         BC    8,WAITRD               CC = 0, BRANCH TO WAITRD         A2104920
         BC    3,LDSTOP               CC = 2 OR 3, ERROR WAIT-ERR1     A2104930
         TM    LDRCSW+5,X'06'         TEST FOR CHANNEL OR INTERFACE    A2104940
*                                      CONTROL  CHECK                  A2104950
```

```
         BC    5,LDSTOP              YES,BRANCH TO ERROR WAIT-ERR1    A2104960
         TM    LDRCSW+4,X'10'        NO,TEST FOR BUSY CONDITION       A2104970
         BC    1,SIORDR              YES,START I/O OPERATION AGAIN    A2104980
         BC    15,LDSTOP             NO,BRANCH TO ERROR WAIT - ERR1   A2104990
RDIRUP   TM    INTFLG,X'80'          IS INTERRUPTION EXPECTED         A2105000
         BC    12,RETINT             NO, BRANCH TO RETINT             A2105010
         CLC   IOOPSW+2(2),SIORDR+2  YES,IS INPUT DEVICE INTERRUPTED  A2105020
         BC    6,RETINT              NO,BRANCH TO RETINT              A2105030
         TM    LDRCSW+5,X'0E'        YES,TEST FOR CCC,CDC,ICC         A2105040
         BC    5,LDSTOP              YES,BRANCH TO ERROR WAIT-ERR1    A2105050
         TM    LDRCSW+4,X'04'        NO,TEST IF DE  IS PRESENT        A2105060
         BC    14,STSTAT             NO,SAVE STATUS                   A2105070
         OC    DEVCSW(1),LDRCSW+4    YES,SAVE STATUS                  A2105080
         TM    DEVCSW,X'02'          TEST IF UC IS PRESENT            A2105090
         BC    1,LDSTOP              YES,BRANCH TO ERROR WAIT-ERR1    A2105100
         NI    INTFLG,X'7F'          NO,CLEAR INTERRUPTION FLAG       A2105110
         BCR   15,LEXIT3             READING COMPLETED-RETURN TO CAL- A2105120
*                                    LER                              A2105130
RETINT   LPSW  IOOPSW                RETURN TO POINT OF INTERRUPTION  A2105140
STSTAT   MVC   DEVCSW(1),LDRCSW+4    STORE DEVICE STATUS BITS         A2105150
WAITRD   OI    INTFLG,X'80'          SET INTERRUPTION FLAG ON         A2105160
         LPSW  RDWAIT                LOAD READ WAIT PSW               A2105170
         EJECT                                                        A2105180
*************************************************************************A2105190
*                                                                     * A2105200
*               HEXADECIMAL-BINARY  CONVERSION ROUTINE                * A2105210
*                                                                     * A2105220
*  ENTRY POINT = LHEXB1                                               * A2105230
*  THIS ROUTINE CONVERTS A SPECIFIED NUMBER OF CHARACTERS OF HEXADE-  * A2105240
*  CIMAL DATA TO BINARY FORM. IT IS ENTERED FROM THE REP CARD ROUTI-   * A2105250
*  NE AT LOCATION LHEXB1 FOR CONVERSION OF AN ADDRESS OR OF A   TEXT  * A2105260
*  CORRECTION   OR AT LOCATION LHEXB5 IF THE REP CARD IS INCORRECTLY  * A2105270
*  DEFINED.                                                           * A2105280
*  THE MAXIMUM FIELD LENGTH HANDLED  BY CALL IS 8 BYTES.              * A2105290
*                                                                     * A2105300
*  CALLING SEQUENCE                                                   * A2105310
*      L       LEXIT1,START OF FIELD ADDRESS                          * A2105320
*      L       LEXIT2,FIELD LENGTH IN BYTES                           * A2105330
*      BAL     LEXIT3,LHEXB1                                          * A2105340
*                                                                     * A2105350
*  ANSWER RETURNED IN  LEXIT4                                         * A2105360
*  ADDRESS UPDATED IN  LEXIT1                                         * A2105370
*                                                                     * A2105380
*  THE CHARACTER VALIDITY CHECK DETERMINES WHETHER THE VALID CHARAC-  * A2105390
*  IS  ALPHABETIC (A - F) OR NUMERIC (1 - 9).                         * A2105400
*  IF THE ROUTINE ENCOUNTERS AN INVALID  HEXADECIMAL  CHARACTER,  IT  * A2105410
*  EXITS TO LOCATION LHEXB5. AN ERROR WAIT PSW WITH  ALL  INTERRUPTS  * A2105420
*  DISABLED AND THE ERROR INDICATION  '3'  IN  ITS ADDRESS FIELD  IS  * A2105430
*  LOADED.                                                            * A2105440
*  IF THE ROUTINE ENCOUNTERS NO INVALID CHARACTER, IT EXITS  TO  THE  * A2105450
*  ADDRESS CONTAINED IN GENERAL REGISTER LEXIT3.                      * A2105460
*                                                                     * A2105470
*************************************************************************A2105480
         SPACE                                                        A2105490
LHEXB1   SR    LEXIT4,LEXIT4         CLEAR OUTPUT REGISTER            A2105500
```

```
         SR    LWK3RG,LWK3RG          CLEAR WORKING REGISTER           A2105510
LHEXB2   CLI   0(LEXIT1),C'0'         IS BYTE VALUE GREATER THAN 0     A2105520
         BC    4,LHEXB4               NO,CHARACTER NOT NUMERIC,BRANCH  A2105530
         CLI   0(LEXIT1),C'9'         YES,IS BYTE VALUE GREATER THAN 9 A2105540
         BC    3,LHEXB5               YES,INVALID CHARACTER,BRANCH     A2105550
         NI    0(LEXIT1),X'0F'        NO,CLEAR HIGH ORDER BITS (ZONE)  A2105560
         IC    LWK3RG,0(LEXIT1,0)     LOAD BYTE INTO WORKING REGISTER  A2105570
LHEXB3   SLL   LEXIT4,4               SHIFT REGISTER 4 BYTES LEFT      A2105580
         AR    LEXIT4,LWK3RG          ADD NEW HEXAD. DIGIT TO REGISTER A2105590
         LA    LEXIT1,1(LEXIT1)       LOAD ADDRESS OF NEXT BYTE        A2105600
         BCT   LEXIT2,LHEXB2          BRANCH IF ANY BYTES ARE LEFT     A2105610
         BCR   15,LEXIT3              CONVERSION END-RETURN TO CALLER  A2105620
LHEXB4   CLI   0(LEXIT1),C'A'         IS BYTE VALUE GREATER THAN A     A2105630
         BC    4,LHEXB5               NO,INVALID CHARACTER,BRANCH      A2105640
         CLI   0(LEXIT1),C'F'         YES,IS BYTE VALUE GREATER THAN F A2105650
         BC    3,LHEXB5               YES,INVALID CHARACTER,BRANCH     A2105660
         IC    LWK3RG,0(LEXIT1,0)     NO,INSERT BYTE IN WORK.REG.      A2105670
         SH    LWK3RG,CONSB7          CONVERT ALPHABETIC TO BINARY     A2105680
         BC    15,LHEXB3              BRANCH TO LHEXB3                 A2105690
LHEXB5   MVI   STOPSW+7,X'03'         INCORRECT REP CARD OR HEXADECI-  A2105700
*                                     MAL CHARACTER                    A2105710
         BC    15,LDSTOP              BRANCH TO ERROR WAIT-ERR3        A2105720
         EJECT                                                         A2105730
*********************************************************************** A2105740
*                                                                     * A2105750
*                    DEFINITION OF CONSTANTS                          * A2105760
*                                                                     * A2105770
*********************************************************************** A2105780
         SPACE                                                        A2105790
         DS    0D                     *                                A2105800
RDWAIT   DC    X'FE06'                READ WAIT PSW (I/O INTERRUPTIONS A2105810
         DC    XL6'0'                 ENABLED,WAIT STATE BIT ON). THIS A2105820
*                                     PSW IS LOADED AFTER A READ OPE-  A2105830
*                                     RATION HAS BEEN STARTED.         A2105840
PRGPSW   DC    X'0004000000'          PROGRAM NEW PSW.                 A2105850
         DC    AL3(CLEAR3)            INTERRUPTION ADDRESS FOR THE 1ST A2105860
*                                     CHECK (ADDRESSING EXCEPTION CAU- A2105870
*                                     SED BY CLEAR SUBRTN). WAIT BIT   A2105880
*                                     IS SET ON FOR THE FOLLOWING.     A2105890
STOPSW   DC    X'00060000'            ERROR WAIT PSW (WAIT STATE BIT   A2105900
         DC    X'00000001'            ON, ALL INTERRUPTS DISABLED).    A2105910
*                                     THIS PSW IS LOADED WHEN LOADER   A2105920
*                                     ENCOUNTERS AN I/O ERROR OR END   A2105930
*                                     OF FILE,CONDITION, AN ATTEMPTED  A2105940
*                                     LOADER OVERLAY ERROR CONDITION   A2105950
*                                     OR AN INCORRECT REP CARD. THE    A2105960
*                                     ERROR INDICATION (1,2 OR 3) IS   A2105970
*                                     PLACED IN ADDRESS FIELD OF PSW.  A2105980
LDTPSW   DC    X'000500000F'          TRANSFER PSW LOADED AT END OF    A2105990
         DC    XL3'0'                 LOADING (PROBLEM STATE, ALL IN-  A2106000
*                                     TERRUPTS EXCEPT MACHINE AND PRO- A2106010
*                                     GRAM CHECKS DISABLED).           A2106020
LRDCCW   DC    X'02'                  READ SELECT STACKER 1-CCW        A2106030
         DC    AL3(LDXBUF)            ADDRESS OF INPUT BUFFER          A2106040
         DC    X'20000050'            SLI FLAG ON - COUNT = 80         A2106050
```

```
LTXTCD   DC    X'02E3E7E3'            TXT CARD TYPE                      A2106060
LREPCD   DC    X'02D9C5D7'            REP CARD TYPE                      A2106070
LENDCD   DC    X'02C5D5C4'            END CARD TYPE                      A2106080
LLDRCD   DC    X'02D3C4D9'            LDR CARD TYPE                      A2106090
LLDTCD   DC    X'02D3C4E3'            LDT CARD TYPE                      A2106100
CONSB7   DC    X'00B7'                DECIMAL 183                        A2106110
DEVCSW   DC    X'00'                  DEVICE STATUS BITS                 A2106120
PASFLG   DC    X'00'                  BIT 7 - GENERAL PURPOSE FLAG       A2106130
         EJECT                                                           A2106140
*********************************************************************** A2106150
*                                                                     * A2106160
*                    DEFINITION OF EQUIVALENTS                        * A2106170
*                                                                     * A2106180
*********************************************************************** A2106190
         SPACE 1                                                        A2106200
INTFLG   EQU   PASFLG                BIT 0-INTERRUPTION FLAG USED IN    A2106210
*                                          READ ROUTINE.              A2106220
CPREND   EQU   PASFLG                BIT 2-1ST END CARD ADDRESS SAVED   A2106230
*                                          USED IN END CARD ROUTINE.  A2106240
LDENTR   EQU   PASFLG                BIT 3-FLAG FOR ENTRY POINT FOUND   A2106250
*                                          USED IN END CARD ROUTINE   A2106260
LDXBUF   EQU   GETCRD-80             INPUT BUFFER (80 BYTES). OVERLAY   A2106270
*                                    INITIALIZATION ROUTINE.            A2106280
ALALFA   DC    A(LDXBUF-128)         PSW IMAGE AREA (USED TO SAVE       A2106290
*                                    PSWS IN CONTROL AFTER LOADING).    A2106300
INITAD   EQU   384                   FIRST BYTE AFTER LOG-OUT AREA      A2106310
LDBUFF   EQU   8                     CONTROL DATA BUFFER (8 BYTES).     A2106320
PSWZON   EQU   MOVTXT+2              DECIMAL 128                        A2106330
ENDAD    DC    A(CLEAR0-256)         *                                  A2106340
         SPACE 2                                                        A2106350
*********************************************************************** A2106360
*                                                                     * A2106370
*                       REGISTER  ASSIGNMENT                          * A2106380
*                                                                     * A2106390
*********************************************************************** A2106400
         SPACE                                                          A2106410
LBASRG   EQU   15                    BASE REGISTER                      A2106420
LEXIT1   EQU   11                    RETURN REGISTER 1                  A2106430
LEXIT2   EQU   12                    RETURN REGISTER 2                  A2106440
LEXIT3   EQU   13                    RETURN REGISTER 3                  A2106450
LEXIT4   EQU   14                    RETURN REGISTER 4                  A2106460
LWK1RG   EQU   1                     WORKING REGISTER 1                 A2106470
LWK2RG   EQU   2                     WORKING REGISTER 2                 A2106480
LWK3RG   EQU   3                     WORKING REGISTER 3                 A2106490
LOCCTR   EQU   4                     LOCATION COUNTER REGISTER          A2106500
LDREND   EQU   ENDAD                 END OF LOADER                      A2106510
         SPACE                                                          A2106520
         END                                                            A2106530
         AOPTN CROSSREF                                                 A2200010
A22B     TITLE 'CONTROL PROGRAM FOR CURRENT SYSTEMS SIMULATORS V-1,L-1' A2200020
CONTPR   START 0                                                        A2200030
         USING *,0                                                      A2200040
         SPACE 2                                                        A2200050
*********************************************************************** A2200060
*                                                                     * A2200070
```

```
*                            CONTROL PROGRAM                            * A2200080
*                                                                       * A2200090
*                                  FOR                                  * A2200100
*                                                                       * A2200110
*                 IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620             * A2200120
*                                                                       * A2200130
*                 -----------------------------------------             * A2200140
*                                                                       * A2200150
*  THE CONTROL PROGRAM CONSISTS OF ROUTINES TO PERFORM THE FOLLOWING    * A2200160
*  FUNCTIONS=                                                           * A2200170
*                                                                       * A2200180
*      MACHINE-CHECK INTERRUPTION PROCESSING                            * A2200190
*      SUPERVISOR-CALL (SVC) INTERRUPTION PROCESSING                    * A2200200
*      PROGRAM INTERRUPTION PROCESSING                                  * A2200210
*      EXTERNAL INTERRUPTION PROCESSING                                 * A2200220
*      I/O DEVICE VERIFICATION                                          * A2200230
*      I/O REQUESTS                                                     * A2200240
*      I/O INTERRUPTION PROCESSING                                      * A2200250
*      SETTING UP SEREP INTERFACE                                       * A2200260
*      COMMUNICATION WITH THE OPERATOR'S CONSOLE                        * A2200270
*                                                                       * A2200280
*  THE FUNCTIONS OF THE CONTROL PROGRAM ARE DESCRIBED WITH THE APPRO-   * A2200290
*  PRIATE ROUTINES.                                                     * A2200300
*                                                                       * A2200310
*  TO FACILITATE THE  UNDERSTANDING OF THE FUNCTIONS,  A DESCRIPTION    * A2200320
*  OF THE FUNCTION WILL BE GIVEN, FOLLOWED BY A DETAILED EXPLANATION    * A2200330
*  OF THE CORRESPONDING ROUTINE(S).                                     * A2200340
*                                                                       * A2200350
*  THE CONTROL PROGRAM WILL OPERATE IN THE SUPERVISOR STATE, WHEREAS    * A2200360
*  THE SIMULATOR WILL OPERATE IN THE PROBLEM STATE.   ANY ATTEMPT TO    * A2200370
*  EXECUTE A PRIVILEGED INSTRUCTION WITHIN THE SIMULATOR  WILL CAUSE    * A2200380
*  A PROGRAM INTERRUPTION.                                              * A2200390
*                                                                       * A2200400
************************************************************************* A2200410
         EJECT                                                            A2200420
************************************************************************* A2200430
*                                                                       * A2200440
*  TO PERFORM  ITS FUNCTIONS,  THE CONTROL PROGRAM USES  THREE MAJOR    * A2200450
*  ELEMENTS= SVC CALLING SEQUENCES, CHANNEL CONTROL BLOCKS AND  UNIT    * A2200460
*  CONTROL BLOCKS.                                                      * A2200470
*                                                                       * A2200480
*  THE SVC CALLING SEQUENCES CONSIST OF AN SVC INSTRUCTION, FOLLOWED    * A2200490
*  BY A CERTAIN NUMBER OF PARAMETERS.  THESE SEQUENCES ARE USED  TO     * A2200500
*  TRANSFER CONTROL TO AND FROM  THE VARIOUS ROUTINES IN THE CONTROL    * A2200510
*  PROGRAM.                                                             * A2200520
*                                                                       * A2200530
*  THE CONTROL PROGRAM CONTAINS  ONE CHANNEL CONTROL BLOCK (CCB) FOR    * A2200540
*  EACH AVAILABLE SYSTEM/360 CHANNEL.   THE CCB CONTAINS INFORMATION    * A2200550
*  DESCRIBING  THE STATUS OF  THE CHANNEL,  PLUS A LIST OF THOSE DE-    * A2200560
*  VICES WHICH ARE ATTACHED TO THE CHANNEL.                             * A2200570
*                                                                       * A2200580
*  THE CONTROL  PROGRAM CONTAINS  ONE UNIT CONTROL BLOCK  (UCB)  FOR    * A2200590
*  EACH AVAILABLE  SYSTEM/360 DEVICE.   THE UCB CONTAINS INFORMATION    * A2200600
*  DESCRIBING THE NATURE AND STATUS OF THE DEVICE.                      * A2200610
*                                                                       * A2200620
```

```
*  THREE GENERAL REGISTERS  I, J AND K  ARE ASSIGNED TO CONTAIN  THE  * A2200630
*  ADDRESSES OF THE FIRST BYTE OF AN  SVC CALLING SEQUENCE, OF A UCB  * A2200640
*  AND OF A CCB, RESPECTIVELY.                                        * A2200650
*                                                                     * A2200660
*  USING I, J OR K, ANY ELEMENT IN AN SVC CALLING SEQUENCE, IN A UCB  * A2200670
*  OR IN A CCB  MAY BE EXPRESSED AS A DISPLACEMENT PLUS THE CONTENTS  * A2200680
*  OF I, J OR K.                                                      * A2200690
*                                                                     * A2200700
*                                *****                                * A2200710
*                                                                     * A2200720
*  THE FOLLOWING NOTES  ILLUSTRATE THE NOTATION  USED THROUGHOUT THE  * A2200730
*  CONTROL PROGRAM DOCUMENTATION.                                     * A2200740
*                                                                     * A2200750
*      I        DENOTES,  WHEN THERE IS NO RISK OF AMBIGUITY,  EITHER * A2200760
*               A GENERAL REGISTER ITSELF OR ITS CONTENTS.             * A2200770
*                                                                     * A2200780
*     (I)       DENOTES,  WHEN THERE IS A RISK OF AMBIGUITY, THE CON- * A2200790
*               TENTS OF A GENERAL REGISTER.   IN THIS CASE,  I  WILL * A2200800
*               DENOTE EXCLUSIVELY THE GENERAL REGISTER ITSELF.       * A2200810
*                                                                     * A2200820
*      XXX(I)   DENOTES THE SYSTEM/360 ADDRESS (I)+XXX,  WHERE XXX IS * A2200830
*               A DISPLACEMENT.   WHEN THERE IS NO RISK OF AMBIGUITY, * A2200840
*               THIS EXPRESSION WILL ALSO BE USED TO INDICATE THE IN- * A2200850
*               FORMATION AT THIS ADDRESS.                            * A2200860
*                                                                     * A2200870
*     (XXX(I)) DENOTES THE CONTENTS OF THE BYTE,  HALFWORD,  WORD OR  * A2200880
*               DOUBLE WORD AT ADDRESS XXX(I).   THIS EXPRESSION WILL * A2200890
*               BE USED IN CASES OF AMBIGUITY,  IN WHICH CASE  XXX(I) * A2200900
*               WILL BE USED  EXCLUSIVELY TO DENOTE  A SYSTEM/360 AD- * A2200910
*               DRESS.                                                * A2200920
*                                                                     * A2200930
********************************************************************** A2200940
         EJECT                                                         A2200950
********************************************************************** A2200960
*                                                                     * A2200970
*      GENERAL REGISTER ASSIGNMENT                                    * A2200980
*                                                                     * A2200990
********************************************************************** A2201000
         SPACE                                                         A2201010
I        EQU   1                    ADDRESS OF 1ST BYTE OF SVC        A2201020
*                                     CALLING SEQUENCE                A2201030
J        EQU   2                    ADDRESS OF 1ST BYTE OF UCB        A2201040
DEVICE   EQU   3                    ADDRESS OF SYSTEM/360 DEVICE      A2201050
BASE     EQU   3                    WORKING REGISTER                  A2201060
INTCDE   EQU   3                    INTERRUPTION CODE IN SVC INSTR.   A2201070
POINTR   EQU   3                    POINTER TO NEXT UCB IN CHAIN      A2201080
BUFF     EQU   3                    ADDRESS OF MESSAGE BUFFER         A2201090
CHPTR    EQU   3                    POINTER TO NEXT UCB ON CHANNEL    A2201100
WORKA    EQU   3                    WORKING REGISTER                  A2201110
WORK     EQU   4                    WORKING REGISTER                  A2201120
L        EQU   4                    COUNT OF WAITING INTERRUPTIONS    A2201130
P        EQU   4                    USED TO SAVE THE CHANNEL NUMBER   A2201140
*                                     WHEN DETERMINING THE INDEX J    A2201150
*                                     OF A UCB                        A2201160
LINKA    EQU   4                    RETURN ADDRESS FOR SUBROUTINE     A2201170
```

```
*                                       CALL (LEVEL 2)                  A2201180
LINK     EQU   5                      RETURN ADDRESS FOR SUBROUTINE     A2201190
*                                       CALL (LEVEL 1)                  A2201200
K        EQU   6                      ADDRESS OF 1ST BYTE OF CCB        A2201210
*                                       (IOQBEG)                        A2201220
BASEGR   EQU   7                      USED TO BRANCH TO THE SYSTEM/360  A2201230
*                                       MAIN STORAGE DUMP PROGRAM OR    A2201240
*                                       TO THE I/O SUPPORT PACKAGE      A2201250
*                                       PROGRAM.                        A2201260
         EJECT                                                          A2201270
*********************************************************************** A2201280
*                                                                     * A2201290
*      DEFINITION OF CHANNEL AND DEVICE SYMBOLIC STATUS BITS          * A2201300
*                                                                     * A2201310
*********************************************************************** A2201320
         SPACE                                                          A2201330
A        EQU   X'80'                  ATTENTION                         A2201340
SM       EQU   X'40'                  STATUS MODIFIER                   A2201350
CUE      EQU   X'20'                  CONTROL UNIT END                  A2201360
B        EQU   X'10'                  BUSY                              A2201370
CE       EQU   X'08'                  CHANNEL END                       A2201380
DE       EQU   X'04'                  DEVICE END                        A2201390
UC       EQU   X'02'                  UNIT CHECK                        A2201400
UE       EQU   X'01'                  UNIT EXCEPTION                    A2201410
PCI      EQU   X'80'                  PROGRAM-CONTROLLED INTERRUPTION   A2201420
IL       EQU   X'40'                  INCORRECT LENGTH                  A2201430
PC       EQU   X'20'                  PROGRAM CHECK                     A2201440
PRC      EQU   X'10'                  PROTECTION CHECK                  A2201450
DC       EQU   X'08'                  CHANNEL DATA CHECK                A2201460
CCC      EQU   X'04'                  CHANNEL CONTROL CHECK             A2201470
IC       EQU   X'02'                  INTERFACE CONTROL CHECK           A2201480
CHC      EQU   X'01'                  CHAINING CHECK                    A2201490
UCORUE   EQU   X'03'                  UNIT CHECK/EXCEPTION              A2201500
AORDE    EQU   X'84'                  ATTENTION/DEVICE END              A2201510
PORPRC   EQU   X'30'                  PROGRAM/PROTECTION CHECK          A2201520
DESTAT   EQU   X'FF'                  ANY DEVICE STATUS BIT             A2201530
CDICCC   EQU   X'0E'                  CHANNEL DATA/INTERFACE/CHANNEL    A2201540
*                                       CONTROL CHECK                   A2201550
CHSTAT   EQU   X'FF'                  ANY CHANNEL STATUS BIT            A2201560
CHERST   EQU   X'1F'                  PRC+DC+CCC+IC+CH                  A2201570
CHNRST   EQU   X'BF'                  CHANNEL STATUS BITS, EXCEPT IL    A2201580
         EJECT                                                          A2201590
*********************************************************************** A2201600
*                                                                     * A2201610
*  CHANNEL CONTROL BLOCKS                                             * A2201620
*                                                                     * A2201630
*                                  *****                              * A2201640
*                                                                     * A2201650
*      DEFINITION OF PARAMETERS RELATED TO THE CCB'S                  * A2201660
*  THESE PARAMETERS ARE  EXPRESSED AS DISPLACEMENTS  RELATIVE TO THE  * A2201670
*  ADDRESS  K  OF THE CCB.                                            * A2201680
*                                                                     * A2201690
*********************************************************************** A2201700
         SPACE                                                          A2201710
IOQBEG   EQU   0                      TWO WORDS USED WHEN CHAINING I/O  A2201720
```

```
IOQEND   EQU   4                        REQUESTS FOR A PARTICULAR      A2201730
*                                       CHANNEL                        A2201740
         SPACE                                                         A2201750
*                                      (IOQBEG(K)) IS THE ADDRESS OF   A2201760
*                                       THE FIRST UCB ON THE CHAIN, OR A2201770
*                                       ZERO IF THERE ARE NO UCB'S ON  A2201780
*                                       THE CHAIN                      A2201790
*                                      (IOQEND(K)) IS THE ADDRESS OF   A2201800
*                                       THE WORD, DEVCHN, INTO WHICH   A2201810
*                                       WILL BE PLACED THE ADDRESS OF  A2201820
*                                       THE NEXT UCB TO BE ADDED TO    A2201830
*                                       THE CHAIN                      A2201840
         SPACE 2                                                       A2201850
*********************************************************************** A2201860
*  CHAINING OF I/O REQUESTS  IS MORE FULLY EXPLAINED  IN CONJUNCTION  * A2201870
*  WITH THE  STACK  AND  UNSTAK  ROUTINES.                            * A2201880
*********************************************************************** A2201890
         SPACE 2                                                       A2201900
DEVTAB   EQU   8                       LIST OF DEVICES ATTACHED TO THE A2201910
*                                       CHANNEL                        A2201920
         SPACE                                                         A2201930
*             EACH WORD IN THE LIST CORRESPONDS TO A DEVICE ATTACHED    A2201940
*             TO THE CHANNEL                                           A2201950
*             BITS 5-7 OF EACH WORD CONTAINS THE SYSTEM/360 ADDRESS OF A2201960
*             THE DEVICE, EXCLUDING THE CHANNEL PART                   A2201970
*             BITS 8-31 OF EACH WORD CONTAINS THE ADDRESS OF THE ASSO- A2201980
*             CIATED UNIT CONTROL BLOCK (UCB)                          A2201990
*             DEVTAB IS TERMINATED BY A WORD CONTAINING ZEROS          A2202000
         EJECT                                                         A2202010
*********************************************************************** A2202020
*                                                                     * A2202030
*  UNIT CONTROL BLOCKS                                                * A2202040
*                                                                     * A2202050
*                                  *****                              * A2202060
*                                                                     * A2202070
*      DEFINITION OF PARAMETERS RELATED TO THE UCB'S                  * A2202080
*  THESE PARAMETERS ARE  EXPRESSED AS DISPLACEMENTS  RELATIVE TO THE  * A2202090
*  ADDRESS  J  OF THE UCB.                                            * A2202100
*                                                                     * A2202110
*********************************************************************** A2202120
         SPACE                                                         A2202130
DEVTYP   EQU   0                       FOUR CHARACTERS WHICH DEFINE THE A2202140
*                                       TYPE OF DEVICE (USED DURING    A2202150
*                                       I/O VERIFICATION)              A2202160
DEV360   EQU   4                       TWO BYTES CONTAINING THE SYSTEM/ A2202170
*                                       360 ADDRESS OF THE DEVICE      A2202180
DEVSPF   EQU   6                       ONE BYTE WHICH DEFINES THE      A2202190
*                                       SPECIAL FEATURES OF THE        A2202200
*                                       DEVICE (USED DURING I/O        A2202210
*                                       VERIFICATION)                  A2202220
BORCH    EQU   7                       ONE BYTE WHICH DEFINES THE      A2202230
*                                       STATUS OF THE DEVICE           A2202240
         SPACE                                                         A2202250
*             THE BYTE IS DECOMPOSED THUS=                             A2202260
         SPACE                                                         A2202270
```

```
*                THE SIX LEFTMOST BITS ARE ALWAYS ZERO                  A2202280
*                THE TWO RIGHTMOST BITS TAKE THE VALUES=                A2202290
         SPACE                                                          A2202300
*                      00     DEVICE AVAILABLE                         A2202310
*                      01     DEVICE CHAINED                           A2202320
*                      11     DEVICE BUSY                              A2202330
*                      10     DEVICE BUSY, BUT UNIT CHECK ENCOUNTERED  A2202340
         SPACE                                                          A2202350
DEVSVC   EQU   8                    A ONE-WORD QUANTITY USED ONLY      A2202360
*                                     WHEN THE DEVICE IS BUSY OR       A2202370
*                                     CHAINED                          A2202380
*                                   CONTAINS THE ADDRESS  I  OF THE    A2202390
*                                     FIRST BYTE OF THE SVC CALLING    A2202400
*                                     SEQUENCE ASSOCIATED WITH THE     A2202410
*                                     DEVICE                           A2202420
         SPACE                                                          A2202430
DEVCHN   EQU   12                   A ONE-WORD QUANTITY USED WHEN      A2202440
*                                     CHAINING SVC 1 OR SVC 2          A2202450
*                                     CALLING SEQUENCES                A2202460
*                                   CONTAINS THE ADDRESS OF THE NEXT   A2202470
*                                     UCB ON THE CHAIN, OR ZERO IF     A2202480
*                                     THERE ARE NO MORE UCB'S ON THE   A2202490
*                                     CHAIN                            A2202500
DEVINT   EQU   16                   A ONE-WORD QUANTITY CONTAINING     A2202510
*                                     THE ADDRESS OF THE FIRST         A2202520
*                                     BYTE OF THE EXIT SEQUENCE USED   A2202530
*                                     BY THE ROUTINE WHICH TREATS      A2202540
*                                     I/O INTERRUPTIONS                A2202550
         SPACE 3                                                        A2202560
UCBSNS   EQU   20                   THREE BYTES RESERVED FOR THE       A2202570
*                                     4TH, 5TH AND 6TH SENSE BYTES     A2202580
*                                     ARISING FROM A SENSE             A2202590
*                                     OPERATION PERFORMED AS A         A2202600
*                                     RESULT OF A UNIT CHECK           A2202610
*                                     CONDITION DETECTED DURING THE    A2202620
*                                     EXECUTION OF AN I/O REQUEST      A2202630
*                                     FOR THE DEVICE                   A2202640
*                                   IF ONE OR MORE OF THESE SENSE      A2202650
*                                     BYTES DOES NOT OCCUR, THE COR-   A2202660
*                                     RESPONDING BYTE(S) IN UCBSNS     A2202670
*                                     WILL BE SET TO ZERO              A2202680
         SPACE                                                          A2202690
INVST    EQU   23                   A ONE-BYTE QUANTITY USED TO MASK   A2202700
*                                     OUT THE DEVICE STATUS BYTE OF    A2202710
*                                     THE CSW                          A2202720
*                                   CONTAINS ONES ONLY IN THOSE BITS   A2202730
*                                     WHICH CORRESPOND TO ANY DEVICE   A2202740
*                                     STATUS BITS WHICH SHOULD NOT     A2202750
*                                     OCCUR FOR THIS TYPE OF DEVICE    A2202760
         SPACE                                                          A2202770
DEVATT   EQU   24                   A ONE-WORD QUANTITY USED IN        A2202780
*                                     CONNECTION WITH UNACCOMPANIED    A2202790
*                                     ATTENTION INTERRUPTIONS FOR      A2202800
*                                     THE DEVICE                       A2202810
*                                   IF AN ATTENTION INTERRUPTION       A2202820
```

```
*                                       CANNOT OCCUR FOR THIS DEVICE,  A2202830
*                                       THIS WORD MAY BE OMITTED FROM  A2202840
*                                       THE UCB                        A2202850
         SPACE                                                         A2202860
BCHMSK   EQU   X'01'              FLAGS AND MASKS                      A2202870
BSYFLG   EQU   X'03'              *                                    A2202880
FREFLG   EQU   X'00'              *                                    A2202890
CHNFLG   EQU   X'01'              *                                    A2202900
SNSFLG   EQU   X'02'              *                                    A2202910
         EJECT                                                        A2202920
*************************************************************************A2202930
*                                                                     * A2202940
*      DEFINITION OF PARAMETERS IN AN I/O REQUEST SEQUENCE            * A2202950
*  THESE PARAMETERS ARE  EXPRESSED AS DISPLACEMENTS  RELATIVE TO THE  * A2202960
*  ADDRESS  I  OF THE SVC CALLING SEQUENCE.                           * A2202970
*                                                                     * A2202980
*************************************************************************A2202990
         SPACE                                                        A2203000
CAWADD   EQU   4                  DENOTES THE ADDRESS OF THE FIRST   A2203010
*                                   CCW TO BE EXECUTED               A2203020
         SPACE                                                        A2203030
STATUS   EQU   8                  ONE BYTE CONTAINING TWO            A2203040
*                                   HEXADECIMAL DIGITS,              A2203050
*                                   LABELED ERRTYP AND STRTBT        A2203060
         SPACE                                                        A2203070
ERRTYP   EQU   STATUS             ON RECEIPT OF AN I/O REQUEST,      A2203080
*                                   ERRTYP IS SET TO ZERO            A2203090
         SPACE                                                        A2203100
*                                  WHEN ALL ACTIVITY RELATED TO       A2203110
*                                    THIS REQUEST HAS BEEN TERMIN-    A2203120
*                                    ATED, ERRTYP TAKES THE           A2203130
*                                    FOLLOWING VALUES=                A2203140
         SPACE                                                        A2203150
*                                     1 - THE DEVICE,CONTROL UNIT,    A2203160
*                                         SUBCHANNEL OR CHANNEL IS    A2203170
*                                         NON-OPERATIONAL             A2203180
*                                     2 - NO UCB EXISTS FOR THIS      A2203190
*                                         DEVICE                      A2203200
*                                     3 - A PROGRAM CHECK OR PROTEC-  A2203210
*                                         TION CHECK HAS BEEN         A2203220
*                                         DETECTED BY THE CHANNEL     A2203230
*                                    IF NONE OF THESE CONDITIONS      A2203240
*                                    ARISES, ERRTYP REMAINS ZERO      A2203250
         SPACE                                                        A2203260
STRTBT   EQU   STATUS             ON RECEIPT OF AN I/O REQUEST,      A2203270
*                                   STRTBT IS SET TO ZERO            A2203280
*                                  IT WILL BE SET TO ONE WHEN THE     A2203290
*                                    PHYSICAL I/O OPERATION HAS       A2203300
*                                    BEEN INITIALIZED AT THE DEVICE   A2203310
         SPACE                                                        A2203320
SNSADD   EQU   9                  THREE BYTES RESERVED FOR THE       A2203330
*                                   1ST, 2ND AND 3RD SENSE BYTES     A2203340
*                                   ARISING FROM A SENSE             A2203350
*                                   OPERATION PERFORMED AS A         A2203360
*                                   RESULT OF A UNIT CHECK           A2203370
```

```
*                                       CONDITION DETECTED DURING THE  A2203380
*                                       EXECUTION OF AN I/O REQUEST    A2203390
*                                     ON RECEIPT OF AN I/O REQUEST     A2203400
*                                       THESE BYTES ARE SET TO ZERO    A2203410
         EJECT                                                         A2203420
SVCCSW   EQU   12                      A DOUBLE-WORD QUANTITY USED TO  A2203430
*                                       ACCUMULATE CHANNEL STATUS      A2203440
*                                       INFORMATION                    A2203450
*                                     ON RECEIPT OF AN I/O REQUEST,    A2203460
*                                       THE CONTENTS OF SVCCSW ARE SET A2203470
*                                       TO ZERO                        A2203480
*                                     CHANNEL STATUS INFORMATION       A2203490
*                                       GENERATED IN THE CSW BY THE    A2203500
*                                       EXECUTION OF AN I/O REQUEST IS A2203510
*                                       ACCUMULATED IN SVCCSW          A2203520
*                                     IF CHANNEL AND DEVICE STATUS IS  A2203530
*                                       GENERATED ON MORE THAN ONE     A2203540
*                                       OCCASION DURING THE            A2203550
*                                       EXECUTION OF A CHAIN OF I/O    A2203560
*                                       COMMANDS, THE CONTROL PROGRAM  A2203570
*                                       WILL ACCUMULATE THE LOGICAL    A2203580
*                                       'OR' OF THIS STATUS INFORMA-   A2203590
*                                       TION IN THE APPROPRIATE BYTES  A2203600
*                                       OF SVCCSW                      A2203610
         SPACE                                                         A2203620
SVCPSW   EQU   20                      A DOUBLE WORD IN WHICH IS PLACED A2203630
*                                       THE INPUT/OUTPUT OLD PSW       A2203640
*                                       GENERATED BY THE LAST I/O      A2203650
*                                       INTERRUPTION RELATED TO THE    A2203660
*                                       REQUEST                        A2203670
         SPACE 2                      *                                A2203680
*********************************************************************** A2203690
*                                                                     * A2203700
*      DEFINITION OF SENSE BITS USED BY READ/WRITE CONSOLE ROUTINES   * A2203710
*                                                                     * A2203720
*********************************************************************** A2203730
         SPACE                                                         A2203740
INTREQ   EQU   X'40'                   INTERVENTION REQUIRED          A2203750
BUSOUT   EQU   X'20'                   BUS OUT CHECK                  A2203760
EQUCHK   EQU   X'10'                   EQUIPMENT CHECK                A2203770
CR       EQU   X'80'                   COMMAND REJECT                 A2203780
         EJECT                                                         A2203790
IPLPSW   DS    D                       INITIAL PROGRAM LOADING PSW    A2203800
IPLCC1   DS    D                       INITIAL PROGRAM LOADING CCW1   A2203810
IPLCC2   DS    D                       INITIAL PROGRAM LOADING CCW2   A2203820
OEXPSW   DS    D                       EXTERNAL OLD PSW               A2203830
OSVPSW   DS    D                       SUPERVISOR CALL OLD PSW        A2203840
OPRPSW   DS    D                       PROGRAM OLD PSW                A2203850
OMCPSW   DC    X'0000FF0000000000'     MACHINE-CHECK OLD PSW          A2203860
OIOPSW   DS    D                       INPUT/OUTPUT OLD PSW           A2203870
CSW      DS    D                       CHANNEL STATUS WORD            A2203880
CAW      DS    F                       CHANNEL ADDRESS WORD           A2203890
         DS    F                                                      A2203900
         SPACE                                                        A2203910
TIMER    DC    X'FFFFFF00'             TIMER-SET TO MAX VALUE         A2203920
```

```
         DS    F                                                          A2203930
         SPACE                                                            A2203940
*****************************************************************       A2203950
*              EXTERNAL NEW PSW                                 *       A2203960
*****************************************************************       A2203970
         SPACE                                                            A2203980
NEXPSW   DC    X'00'                 I/O AND EXTERNAL INTERRUPTIONS     A2203990
*                                      DISABLED                         A2204000
         DC    X'04'                 ALLOW MACHINE CHECK INTERRUPTION   A2204010
         DC    H'0'                                                     A2204020
         DC    A(EXTRET)             EXTERNAL INTERRUPTION ADDRESS      A2204030
         SPACE                                                          A2204040
*****************************************************************       A2204050
*              SUPERVISOR CALL NEW PSW                          *       A2204060
*****************************************************************       A2204070
         SPACE                                                          A2204080
NSVPSW   DC    X'00'                 I/O AND EXTERNAL INTERRUPTIONS     A2204090
*                                      DISABLED                         A2204100
         DC    X'04'                 ALLOW MACHINE CHECK INTERRUPTION   A2204110
         DC    H'0'                                                     A2204120
         DC    A(SVCINT)             SVC INTERRUPTION ADDRESS           A2204130
         SPACE                                                          A2204140
*****************************************************************       A2204150
*              PROGRAM NEW PSW                                  *       A2204160
*****************************************************************       A2204170
         SPACE                                                          A2204180
NPRPSW   DC    X'00'                 I/O AND EXTERNAL INTERRUPTIONS     A2204190
*                                      DISABLED                         A2204200
         DC    X'06'                 ALLOW MACHINE CHECK INTERRUPTION   A2204210
*                                      AND WAIT                         A2204220
         DC    H'0'                                                     A2204230
         DC    A(PRINT)              INTERRUPTION ADDRESS WHEN NOT IN   A2204240
*                                      WAIT STATE                       A2204250
         EJECT                                                          A2204260
*****************************************************************       A2204270
*                                                               *       A2204280
*  MACHINE-CHECK INTERRUPTION PROCESSING=                       *       A2204290
*                                                               *       A2204300
*  WHEN A MACHINE-CHECK INTERRUPTION OCCURS, THE CONTROL PROGRAM  IS     A2204310
*  ENTERED.  IT WILL RESPOND TO THE INTERRUPTION BY  SETTING UP  THE     A2204320
*  STANDARD SEREP INTERFACE FOR A MACHINE CHECK.                *       A2204330
*                                                               *       A2204340
*  A PSW FOR WHICH I/O, EXTERNAL AND FURTHER MACHINE-CHECK INTERRUP-     A2204350
*  TIONS ARE DISABLED, AND IN WHICH THE WAIT STATE BIT IS ONE,  WILL     A2204360
*  BE LOADED.  THE LAST BYTE OF THIS PSW (ADDRESS FIELD, BITS 56-63)     A2204370
*  WILL CONTAIN ALL ONES.                                       *       A2204380
*                                                               *       A2204390
*  THE WAIT LIGHT ON THE SYSTEM CONTROL PANEL WILL BE SET ON AND OP-     A2204400
*  ERATOR INTERVENTION WILL BE AWAITED.                         *       A2204410
*                                                               *       A2204420
*****************************************************************       A2204430
         SPACE 2                                                        A2204440
*****************************************************************       A2204450
*              MACHINE-CHECK NEW PSW                            *       A2204460
*****************************************************************       A2204470
```

```
         SPACE                                                          A2204480
NMCPSW   DC    X'00'                 I/O AND EXTERNAL INTERRUPTIONS    A2204490
*                                      DISABLED                        A2204500
         DC    X'02'                 STOP MACHINE CHECK INTERRUPTIONS  A2204510
*                                      AND WAIT                        A2204520
         DC    X'0000'                                                 A2204530
         DC    A(255)                                                  A2204540
         SPACE                                                          A2204550
*****************************************************************     A2204560
*              INPUT/OUTPUT NEW PSW                             *     A2204570
*****************************************************************     A2204580
         SPACE                                                          A2204590
NIOPSW   DC    X'00'                 I/O AND EXTERNAL INTERRUPTIONS    A2204600
*                                      DISABLED                        A2204610
         DC    X'04'                 ALLOW MACHINE-CHECK INTERRUPTION  A2204620
         DC    H'0'                                                    A2204630
         DC    A(IOINT)              INPUT/OUTPUT INTERRUPTION ADDR.   A2204640
         SPACE                                                          A2204650
SCAN     DS    64F                   DIAGNOSTIC SCAN-OUT AREA          A2204660
MESPSW   DS    D                     PSW IN MESAGE/COMAND ROUTINES     A2204670
SVCGR    DS    7F                    SAVE AREA FOR GENERAL REGISTERS   A2204680
*                                      1-7 AFTER SVC INTERRUPTION      A2204690
IOGR     DS    7F                    SAVE AREA FOR GENERAL REGISTERS   A2204700
*                                      1-7 AFTER I/O INTERRUPTION      A2204710
MESSGR   DS    7F                    SAVE AREA FOR GENERAL REGISTERS   A2204720
*                                      1-7 FOR MESAGE/COMAND ROUTINES  A2204730
         EJECT                                                          A2204740
*****************************************************************     A2204750
*                                                               *     A2204760
*  PROCESSING OF SUPERVISOR CALL INTERRUPTIONS                  *     A2204770
*                                                               *     A2204780
*  INCLUDED IN THE ROUTINE IS AN SVC TABLE OF 20 SINGLE-WORD ENTRIES  * A2204790
*  CORRESPONDING, IN ORDER, TO THE ALLOWED VALUES  0-19 OF THE INTER- * A2204800
*  RUPTION CODE IN AN SVC INSTRUCTION.  THESE INTERRUPTION CODES AND  * A2204810
*  THEIR CORRESPONDING FUNCTIONS ARE GIVEN IN THE FOLLOWING TABLE.    * A2204820
*                                                               *     A2204830
*****************************************************************     A2204840
         SPACE 2                                                        A2204850
*****************************************************************     A2204860
*                                                               *     A2204870
*           INTERRUPTION CODE                FUNCTION           *     A2204880
*           ----------------                 --------           *     A2204890
*                                                               *     A2204900
*                 0                  I/O DEVICE VERIFICATION    *     A2204910
*                 1                  I/O REQUEST AND INTERRUPT  *     A2204920
*                                      AT CHANNEL END           *     A2204930
*                 2                  I/O REQUEST AND CONTINUE   *     A2204940
*                 3                  RETURN TO POINT OF         *     A2204950
*                                      INTERRUPTION             *     A2204960
*                 4                  WRITE MESSAGE              *     A2204970
*                 5                  SET COMMAND PARAMETERS     *     A2204980
*                 6                  SET PROGRAM INTERRUPTION   *     A2204990
*                                      RETURN                   *     A2205000
*                 7                  SET UP SEREP INTERFACE     *     A2205010
*                 8                  DISABLE                    *     A2205020
```

```
*                  9                ENABLE                          * A2205030
*                 10                SET EXTERNAL INTERRUPTION       * A2205040
*                                     RETURN                        * A2205050
*                 11                I/O REQUEST AND WAIT            * A2205060
*                 12                SYSTEM/360 DUMP                 * A2205070
*                 13                REWIND                          * A2205080
*                 14                REWIND AND UNLOAD               * A2205090
*                 15                DISABLE (CONSOLE)               * A2205100
*                 16                ENABLE  (CONSOLE)               * A2205110
*                 17                LOGICAL I/O REQUEST             * A2205120
*                 18                LOGICAL I/O REQUEST             * A2205130
*                 19                SET WAIT STATE                  * A2205140
*                                                                   * A2205150
********************************************************************* A2205160
        SPACE 2                                                       A2205170
********************************************************************* A2205180
*                                                                   * A2205190
*  IF THE INTERRUPTION CODE IS GREATER THAN 19,  A PROGRAM INTERRUP- * A2205200
*  TION IS  ARTIFICIALLY CREATED.   THE INTERRUPTION CODE PORTION OF * A2205210
*  THE PROGRAM OLD PSW IS SET TO INDICATE AN OPERATION EXCEPTION.    * A2205220
*                                                                   * A2205230
*                                                          .../...  * A2205240
*                                                                   * A2205250
********************************************************************* A2205260
        EJECT                                                         A2205270
********************************************************************* A2205280
*                                                                   * A2205290
*  .../...                                                          * A2205300
*                                                                   * A2205310
*  OTHERWISE, CONTROL IS GIVEN TO THE APPROPRIATE ROUTINE,  VIA  THE* A2205320
*  SVC TABLE.                                                       * A2205330
*                                                                   * A2205340
*  THE GENERAL AND FLOATING-POINT REGISTERS  MAY CONTAIN  ANY  VALUE* A2205350
*  WHEN AN SVC CALLING SEQUENCE IS PRESENTED TO THE CONTROL PROGRAM. * A2205360
*  WHEN CONTROL IS RETURNED TO THE SIMULATOR, THE CONTENTS  OF THESE* A2205370
*  REGISTERS WILL BE UNCHANGED.                                     * A2205380
*                                                                   * A2205390
*  FOR THE MAJORITY OF ITS FUNCTIONS,  THE CONTROL  PROGRAM  MUST BE* A2205400
*  GIVEN A NUMBER OF PARAMETERS.  THE VALUES OF THESE PARAMETERS ARE* A2205410
*  SET UP IN THE BYTES IMMEDIATELY FOLLOWING THE SVC INSTRUCTION. AN* A2205420
*  SVC INSTRUCTION, TOGETHER WITH ITS NECESSARY PARAMETERS IS REFER-* A2205430
*  RED TO AS AN SVC CALLING SEQUENCE.                               * A2205440
*                                                                   * A2205450
*  AN SVC CALLING SEQUENCE IS OF THE GENERAL FORM=                  * A2205460
*                                                                   * A2205470
*             CNOP    X,X                                           * A2205480
*       I     SVC     INTCDE                                        * A2205490
*             DC OR DS                                              * A2205500
*             ...                                                   * A2205510
*             NORMAL AND EXCEPTIONAL RETURN ADDRESSES               * A2205520
*                                                                   * A2205530
*  THE CALLING SEQUENCES FOR EACH VALUE OF INTCDE ARE FOUND WITH THE* A2205540
*  APPROPRIATE ROUTINES.                                            * A2205550
*                                                                   * A2205560
********************************************************************* A2205570
```

```
         SPACE 2                                                        A2205580
*****************************************************************     A2205590
*                                                               *     A2205600
*      SVC INTERRUPTION PROCESSING ROUTINE                      *     A2205610
*                                                               *     A2205620
*****************************************************************     A2205630
         SPACE                                                          A2205640
SVCINT   STM   1,7,SVCGR             (GENERAL REGISTERS 1-7) TO SVCGR   A2205650
         L     I,OSVPSW+4            OSVPSW(A)-2 TO I                   A2205660
         SH    I,DEC2                *                                  A2205670
         SR    INTCDE,INTCDE         2.(INTERRUPTION CODE) TO INTCDE    A2205680
         IC    INTCDE,1(I)           *                                  A2205690
         SLA   INTCDE,1              *                                  A2205700
         LH    BASEGR,SVCTAB(INTCDE) IS INTCDE LESS THAN 20             A2205710
         CH    INTCDE,DEC40          *                                  A2205720
         BCR   4,BASEGR              YES-BRANCH THROUGH SVCTAB          A2205730
SVCERR   MVC   OPRPSW(8),OSVPSW      NO-TREAT AS PROGRAM INTERRUPTION   A2205740
         MVC   OPRPSW+2(2),INTCD1    OSVPSW TO OPRPSW,1 TO INT. CODE    A2205750
         LM    1,7,SVCGR             RELOAD GENERAL REGISTERS 1-7       A2205760
         LPSW  NPRPSW                PROGRAM INTERRUPTION               A2205770
SVCTAB   DC    YL2(VERIFY)           I/O VERIFICATION                   A2205780
         DC    YL2(IOCONT)           I/O REQUEST-INTERRUPT AT CHANNEL   A2205790
*                                      END                              A2205800
         DC    YL2(IOCONT)           I/O REQUEST-CONTINUE               A2205810
         DC    YL2(RETURN)           RETURN TO CALLER                   A2205820
         DC    YL2(MESAGE)           WRITE MESSAGE                      A2205830
         DC    YL2(SETCOM)           SET COMMAND PARAMETERS             A2205840
         DC    YL2(PRSET)            SET PROGRAM INTERRUPTION RETURN    A2205850
         DC    YL2(SEREP)            SET UP SEREP INTERFACE             A2205860
         DC    YL2(DISABL)           DISABLE                            A2205870
         DC    YL2(ENABLE)           ENABLE                             A2205880
         DC    YL2(EXTSET)           SET EXTERNAL INTERRUPTION RETURN   A2205890
         DC    YL2(IOWAIT)           I/O REQUEST-WAIT                   A2205900
DMP360   DC    H'3440'               SYSTEM/360 DUMP ADDRESS            A2205910
         DC    YL2(IOCONT)           REWIND                             A2205920
         DC    YL2(IOCONT)           REWIND AND UNLOAD                  A2205930
         DC    YL2(DISCSL)           DISABLE (CONSOLE)                  A2205940
         DC    YL2(ENACSL)           ENABLE (CONSOLE)                   A2205950
         DC    H'3488'               I/O PACKAGE ADDRESS (FOR LOGICAL   A2205960
         DC    H'3488'                  I/O REQUEST)                    A2205970
         DC    YL2(WAITST)           SET WAIT STATE                     A2205980
         EJECT                                                          A2205990
*****************************************************************     A2206000
*                                                               *     A2206010
*      SVC 6, SET PARAMETERS FOR PROGRAM INTERRUPTION           *     A2206020
*                                                               *     A2206030
*****************************************************************     A2206040
         SPACE                                                          A2206050
PRSET    NI    NPRPSW+1,X'FD'        0 TO WAIT BIT IN NPRPSW            A2206060
         MVC   PRRETA(3),3(I)        (I+2) TO PRRET                     A2206070
         LA    I,6(I)                I+6 TO PRPSW                       A2206080
         ST    I,PRPSWA              *                                  A2206090
PRSETA   LA    I,8(I)                I+14 TO OSVPSW(A)                  A2206100
         BC    15,VERIFB             GO TO I/O VERIFICATION ROUTINE     A2206110
         SPACE 2                                                        A2206120
```

```
*********************************************************************** A2206130
*                                                                     * A2206140
*      SVC 10, SET PARAMETERS FOR TIMER INTERRUPTION                  * A2206150
*                                                                     * A2206160
*********************************************************************** A2206170
         SPACE                                                        A2206180
EXTSET   MVC   NEXPSW+5(3),EXTSWA      1 TO EXTSW                    A2206190
         MVC   TIMINA(8),2(I)          (I+2) TO TIMINA,KEYINA        A2206200
         LA    I,10(I)                 I+10 TO EXTPSW                A2206210
         ST    I,EXTPSW                *                             A2206220
         BC    15,PRSETA               I+18 TO OSVPSW(A), GO TO I/O  A2206230
*                                       VERIFICATION ROUTINE          A2206240
         EJECT                                                        A2206250
*********************************************************************** A2206260
*                                                                     * A2206270
*  DISABLE AND ENABLE FUNCTIONS                                       * A2206280
*                                                                     * A2206290
*      DISABLE=                                                       * A2206300
*      THE SVC CALLING SEQUENCE   SVC 8  WILL CAUSE I/O AND EXTERNAL  * A2206310
*      INTERRUPTIONS  TO BE DISABLED.   I.E., THE SYSTEM MASK WILL BE * A2206320
*      SET TO THE VALUE X'00'.                                        * A2206330
*                                                                     * A2206340
*  THE DISABLED STATE MAY BE SET UP EITHER BY THE 'DISABLE' SVC CAL-  * A2206350
*  LING SEQUENCE OR AS THE RESULT OF AN INTERRUPTION.                 * A2206360
*                                                                     * A2206370
*      ENABLE=                                                        * A2206380
*      THE SVC CALLING SEQUENCE   SVC 9  WILL CAUSE I/O AND EXTERNAL  * A2206390
*      INTERRUPTIONS TO BE ENABLED.   I.E., THE  SYSTEM MASK WILL BE  * A2206400
*      SET TO THE VALUE X'FF'.                                        * A2206410
*                                                                     * A2206420
*      CONSOLE DISABLE=                                               * A2206430
*  THE SVC CALLING SEQUENCE SVC 15 WILL CAUSE AN ATTENTION INTERRUP-  * A2206440
*  TION RESULTING FROM  AN OPERATOR COMMAND ON  THE CONSOLE KEYBOARD  * A2206450
*  TO BE IGNORED.                                                     * A2206460
*                                                                     * A2206470
*      CONSOLE ENABLE=                                                * A2206480
*  THE SVC CALLING SEQUENCE SVC 16  WILL CAUSE SUCH ATTENTION INTER-  * A2206490
*  RUPTIONS TO BE ACCEPTED IF AN SVC 5 (SET COMMAND PARAMETERS) CAL-  * A2206500
*  LING SEQUENCE HAS BEEN PREVIOUSLY SUBMITTED.                       * A2206510
*                                                                     * A2206520
*********************************************************************** A2206530
         SPACE 2                                                      A2206540
*********************************************************************** A2206550
*                                                                     * A2206560
*      SVC 8, DISABLE                                                 * A2206570
*                                                                     * A2206580
*********************************************************************** A2206590
         SPACE                                                        A2206600
DISABL   NI    OSVPSW,X'00'            INHIBIT I/O AND EXTERNAL INT. A2206610
DISRET   BC    15,ENARET               RETURN TO CALLER              A2206620
         SPACE 2                                                      A2206630
*********************************************************************** A2206640
*                                                                     * A2206650
*      SVC 9, ENABLE                                                  * A2206660
*                                                                     * A2206670
```

```
*********************************************************************** A2206680
         SPACE                                                          A2206690
ENABLE   OI    OSVPSW,X'FF'          ALLOW I/O AND EXTERNAL INT.       A2206700
ENARET   LM    1,7,SVCGR             RESTORE GENERAL REGISTERS 1-7     A2206710
         LPSW  OSVPSW                RETURN TO CALLER                  A2206720
         EJECT                                                          A2206730
*********************************************************************** A2206740
*                                                                     * A2206750
*      SVC 15, DISABLE (CONSOLE)                                      * A2206760
*                                                                     * A2206770
*********************************************************************** A2206780
         SPACE                                                          A2206790
DISCSL   OI    ATTSW,X'01'           SET ATTSW=1                       A2206800
         BC    15,ENARET             RETURN TO CALLER                  A2206810
         SPACE 2                                                        A2206820
*********************************************************************** A2206830
*                                                                     * A2206840
*      SVC 16, ENABLE (CONSOLE)                                       * A2206850
*                                                                     * A2206860
*********************************************************************** A2206870
         SPACE                                                          A2206880
ENACSL   NI    ATTSW,X'00'           SET ATTSW=0                       A2206890
         BC    15,ENARET             RETURN TO CALLER                  A2206900
         EJECT                                                          A2206910
*********************************************************************** A2206920
*                                                                     * A2206930
*      SVC 19, SET WAIT STATE                                         * A2206940
*                                                                     * A2206950
*  THE SVC CALLING SEQUENCE SVC 19 WILL SET THE WAIT STATE BIT 'ON'   * A2206960
*  IN THE CURRENT PSW.  ALL I/O AND EXTERNAL  INTERRUPTIONS  WILL BE  * A2206970
*  ENABLED.                                                           * A2206980
*                                                                     * A2206990
*  WHEN AN I/O OR EXTERNAL INTERRUPTION OCCURS,  THE CONTROL PROGRAM  * A2207000
*  WILL BE ENTERED.  THE WAIT STATE BIT WILL BE SET 'OFF' IN THE OLD  * A2207010
*  PSW AT THE POINT  OF INTERRUPTION  AND CONTROL  WILL BE  RETURNED  * A2207020
*  EITHER TO THE POINT OF INTERRUPTION  BY LOADING THE OLD PSW OR TO  * A2207030
*  A PRE-DETERMINED LOCATION.  THE OLD PSW AT THE POINT OF INTERRUP-  * A2207040
*  TION IS ALSO STORED IN A PRE-DETERMINED LOCATION.                  * A2207050
*                                                                     * A2207060
*********************************************************************** A2207070
         SPACE 2                                                        A2207080
WAITST   OI    OSVPSW+1,X'02'        1 TO WAIT STATE BIT IN OSVPSW     A2207090
         BC    15,ENABLE             GO TO ENABLE                      A2207100
         EJECT                                                          A2207110
*********************************************************************** A2207120
*                                                                     * A2207130
*  SIMULATOR INTERRUPTION AND RETURN                                  * A2207140
*                                                                     * A2207150
*  WHEN AN INTERRUPTION OCCURS, CONTROL WILL BE GIVEN TO THE CONTROL  * A2207160
*  PROGRAM.   AS A RESULT,  THE CONTROL  PROGRAM  MAY RETURN CONTROL  * A2207170
*  EITHER  TO THE POINT OF INTERRUPTION OR TO A PRE-DETERMINED LOCA-  * A2207180
*  TION.                                                              * A2207190
*                                                                     * A2207200
*  IN THE LATTER CASE, THE OLD PSW AT THE POINT OF INTERRUPTION WILL  * A2207210
*  BE STORED IN A DOUBLE WORD  AT A PRE-DETERMINED ADDRESS.   IN AD-  * A2207220
```

```
*  DITION, ALL I/O AND EXTERNAL INTERRUPTIONS WILL BE DISABLED.       * A2207230
*                                                                     * A2207240
*  CONTROL MAY BE RETURNED TO  THE POINT OF INTERRUPTION BY USING AN  * A2207250
*  SVC CALLING SEQUENCE OF THE FORM=                                  * A2207260
*                                                                     * A2207270
*              CNOP    2,4                                            * A2207280
*       I      SVC     3                                              * A2207290
*              DC      A(RETPSW)                                      * A2207300
*                                                                     * A2207310
*  WHERE RETPSW DENOTES THE PRE-DETERMINED ADDRESS AT WHICH THE CON-  * A2207320
*  TROL PROGRAM HAS STORED THE OLD PSW.                               * A2207330
*                                                                     * A2207340
*  THE CURRENT  PSW  WILL BE REPLACED  BY THE CONTENTS OF THE DOUBLE  * A2207350
*  WORD WITH ADDRESS  RETPSW, THUS RETURNING CONTROL TO THE POINT OF  * A2207360
*  INTERRUPTION.                                                      * A2207370
*                                                                     * A2207380
********************************************************************** A2207390
         SPACE 2                                                       A2207400
********************************************************************** A2207410
*                                                                     * A2207420
*      ROUTINE TO TREAT RETURN CALLING SEQUENCE                       * A2207430
*                                                                     * A2207440
********************************************************************** A2207450
         SPACE                                                         A2207460
RETURN   L     WORK,2(I)               (RETPSW) TO (TEMP)             A2207470
         MVC   OSVPSW(8),0(WORK)       *                              A2207480
         BC    15,ENARET               RETURN TO CALLER               A2207490
         EJECT                                                       A2207500
********************************************************************** A2207510
*                                                                     * A2207520
*  INTERRUPTION PROCESSING                                            * A2207530
*                                                                     * A2207540
*                                  *****                              * A2207550
*                                                                     * A2207560
*  PROGRAM INTERRUPTION PROCESSING                                    * A2207570
*                                                                     * A2207580
*  WHEN A PROGRAM INTERRUPTION OCCURS,  THE CONTROL  PROGRAM WILL BE  * A2207590
*  ENTERED.                                                           * A2207600
*                                                                     * A2207610
*  AT THE TIME WHEN THE  CONTROL PROGRAM  IS LOADED INTO  SYSTEM/360  * A2207620
*  MAIN STORAGE, PROGRAM INTERRUPTIONS  ARE PROCESSED IN THE FOLLOW-  * A2207630
*  ING WAY=                                                           * A2207640
*                                                                     * A2207650
*      A PSW FOR WHICH I/O AND EXTERNAL  INTERRUPTIONS ARE ENA-       * A2207660
*      BLED, AND IN WHICH THE WAIT STATE BIT IS ONE AND THE IN-       * A2207670
*      TERRUPTION CODE CONTAINS ALL ZEROS, WILL BE LOADED.            * A2207680
*                                                                     * A2207690
*      THE WAIT LIGHT ON THE SYSTEM CONTROL PANEL WILL BE TURNED      * A2207700
*      ON AND, APART FROM THE PROCESSING OF I/O AND EXTERNAL IN-      * A2207710
*      TERRUPTIONS, OPERATOR INTERVENTION WILL BE AWAITED.            * A2207720
*                                                                     * A2207730
*  BY SUBMITTING AN SVC CALLING SEQUENCE OF THE FORM=                 * A2207740
*                                                                     * A2207750
*              CNOP    2,8                                            * A2207760
*       I      SVC     6                                              * A2207770
```

```
*             DC      A(PRRET)                                          * A2207780
*      PRPSW  DS      D                                                 * A2207790
*                                                                       * A2207800
*  ANY SUBSEQUENT  PROGRAM  INTERRUPTIONS WILL  BE PROCESSED  IN THE  * A2207810
*  FOLLOWING WAY=                                                     * A2207820
*                                                                       * A2207830
*  THE  PROGRAM OLD PSW  IS PLACED IN THE  DOUBLE WORD  WITH ADDRESS  * A2207840
*  PRPSW,  I/O AND EXTERNAL INTERRUPTIONS ARE DISABLED,  AND CONTROL  * A2207850
*  IS RETURNED TO THE INSTRUCTION WITH ADDRESS PRRET.                 * A2207860
*                                                                       * A2207870
*                                  *****                                * A2207880
*                                                                       * A2207890
*  EXTERNAL INTERRUPTION PROCESSING                                   * A2207900
*                                                                       * A2207910
*  WHEN A EXTERNAL INTERRUPTION OCCURS, THE CONTROL  PROGRAM WILL BE  * A2207920
*  ENTERED.                                                           * A2207930
*                                                                       * A2207940
*  AT THE TIME WHEN THE  CONTROL PROGRAM  IS LOADED INTO  SYSTEM/360  * A2207950
*  MAIN STORAGE, AND WHEN AN EXTERNAL INTERRUPTION OCCURS DUE TO  AN  * A2207960
*  EXTERNAL SIGNAL, EXTERNAL INTERRUPTIONS ARE IGNORED.   THE  EXTER- * A2207970
*  NAL OLD PSW IS LOADED INTO THE PSW.                                * A2207980
*                                                                       * A2207990
*                                                            .../...  * A2208000
*                                                                       * A2208010
************************************************************************ A2208020
        EJECT                                                           A2208030
************************************************************************ A2208040
*                                                                       * A2208050
*  .../...                                                            * A2208060
*                                                                       * A2208070
*  BY SUBMITTING AN SVC CALLING SEQUENCE OF THE FORM=                 * A2208080
*                                                                       * A2208090
*             CNOP    6,8                                             * A2208100
*      I      SVC     10                                              * A2208110
*             DC      A(TIMINT)                                       * A2208120
*             DC      A(KEYINT)                                       * A2208130
*      EXTPSW DS      D                                               * A2208140
*                                                                       * A2208150
*  ANY SUBSEQUENT EXTERNAL INTERRUPTIONS  RELATED TO THE TIMER OR TO  * A2208160
*  THE INTERRUPT KEY  ON THE SYSTEM CONTROL PANEL  WILL BE PROCESSED  * A2208170
*  IN THE FOLLOWING WAY=                                              * A2208180
*                                                                       * A2208190
*  THE EXTERNAL OLD PSW  IS PLACED IN THE  DOUBLE WORD  WITH ADDRESS  * A2208200
*  EXTPSW,  I/O AND EXTERNAL INTERRUPTIONS ARE DISABLED, AND CONTROL  * A2208210
*  IS RETURNED EITHER TO THE INSTRUCTION WITH ADDRESS TIMINT  (TIMER  * A2208220
*  INTERRUPTION)  OR TO THE INSTRUCTION WITH ADDRESS KEYINT  (INTER-  * A2208230
*  RUPT KEY INTERRUPTION).                                            * A2208240
*                                                                       * A2208250
*  IF THE VALUES  TIMINT  OR  KEYINT  ARE ZERO, THE CORRESPONDING IN- * A2208260
*  TERRUPTION WILL BE IGNORED.                                        * A2208270
*                                                                       * A2208280
************************************************************************ A2208290
        SPACE 2                                                         A2208300
************************************************************************  A2208310
*                                                                       * A2208320
```

```
*      PROCESSING OF PROGRAM INTERRUPTIONS                              * A2208330
*                                                                       * A2208340
************************************************************************ A2208350
         SPACE                                                          A2208360
PRINT    ST    BASE,PRBASE           ENTER HERE AFTER SVC 6             A2208370
         L     BASE,PRPSWA           OPRPSW TO (PRPSW)                  A2208380
         MVC   0(8,BASE),OPRPSW      *                                  A2208390
         L     BASE,PRBASE           *                                  A2208400
         MVC   OPRPSW+5(3),PRRETA    PRRET TO OPRPSW(A)                 A2208410
         NI    OPRPSW,X'00'          DISABLE I/O AND EXTERNAL INT.      A2208420
         LPSW  OPRPSW                LOAD OPRPSW-INTERRUPT              A2208430
         EJECT                                                          A2208440
************************************************************************ A2208450
*                                                                       * A2208460
*      PROCESSING OF EXTERNAL INTERRUPTIONS                             * A2208470
*                                                                       * A2208480
************************************************************************ A2208490
         SPACE                                                          A2208500
EXTINT   STM   BASE,WORK,EXBASE      ENTER HERE AFTER SVC 10            A2208510
         L     BASE,EXTPSW           *                                  A2208520
         TM    OEXPSW+3,X'80'        TEST FOR TIMER INTERRUPTION        A2208530
         BC    5,EXTIMR              YES-BRANCH                         A2208540
         TM    OEXPSW+3,X'40'        NO-TEST FOR INTERRUPT KEY          A2208550
         BC    8,EXTRET              NO-RETURN TO POINT OF INT.         A2208560
         LA    WORK,KEYINA           SAVE KEYINT                        A2208570
         CLC   KEYINA(3),KEYINA+1    IS INTERRUPT KEY EXPECTED          A2208580
         BC    15,EXTOUT             *                                  A2208590
EXTIMR   LA    WORK,TIMINA           SAVE TIMINT                        A2208600
         CLC   TIMINA(3),TIMINA+1    IS TIMER INTERRUPTION EXPECTED     A2208610
EXTOUT   BC    8,EXTRET              NO-RETURN TO POINT OF INT.         A2208620
         NI    OEXPSW+1,X'FD'        0 TO WAIT STATE BIT IN OEXPSW      A2208630
         MVC   0(8,BASE),OEXPSW      YES-SAVE OLD EXTPSW                A2208640
         MVC   OEXPSW+5(3),1(WORK)   TIMINT OR KEYINT TO OEXPSW(A)      A2208650
         LM    BASE,WORK,EXBASE      *                                  A2208660
         NI    OEXPSW,X'00'          DISABLE I/O AND EXTERNAL INTER-    A2208670
*                                     RUPTIONS - ENTER HERE BEFORE       A2208680
*                                     SVC 10                             A2208690
EXTRET   LPSW  OEXPSW                LOAD OEXPSW - INTERRUPT            A2208700
         EJECT                                                          A2208710
************************************************************************ A2208720
*                                                                       * A2208730
*  I/O DEVICE VERIFICATION                                              * A2208740
*                                                                       * A2208750
*  THE FOLLOWING  SVC CALLING SEQUENCE  WILL ENABLE THE SIMULATOR TO    * A2208760
*  VERIFY THAT AN  I/O DEVICE  WITH A PARTICULAR  SYSTEM/360 ADDRESS    * A2208770
*  HAS THE CORRECT PROPERTIES.                                          * A2208780
*                                                                       * A2208790
*             CNOP    0,4                                               * A2208800
*       I     SVC     0                                                 * A2208810
*       DEV360 DC     X'0DDD'                                           * A2208820
*       DEVTYP DC     C'TTTT'                                           * A2208830
*       DEVSPF DC     X'55'                                             * A2208840
*              DC     AL3(ERROR)                                        * A2208850
*       I+12                                                            * A2208860
*                                                                       * A2208870
```

```
*  WHEN THE CONTROL PROGRAM IS ENTERED  WITH THIS  CALLING SEQUENCE,  * A2208880
*  THE  UNIT CONTROL BLOCK  (UCB)  ASSOCIATED WITH THE DEVICE HAVING  * A2208890
*  ADDRESS  DDD  IS EXAMINED.                                         * A2208900
*  IF THIS DEVICE IS OF THE TYPE  TTTT  AND HAS SPECIAL FEATURES COR- * A2208910
*  RESPONDING TO  SS,  CONTROL IS RETURNED  TO THE SIMULATOR  AT AD-  * A2208920
*  DRESS  I+12.                                                       * A2208930
*                                                                     * A2208940
*  IF THE  UCB  DOES NOT EXIST,  OR IF ONE OR BOTH OF  TTTT  AND  SS  * A2208950
*  ARE  INVALID,  CONTROL IS  RETURNED TO THE  SIMULATOR  AT ADDRESS  * A2208960
*  ERROR.                                                             * A2208970
*                                                                     * A2208980
*  THE BIT STRUCTURE  OF THE QUANTITY  SS  FOR THE DEVICES SUPPORTED  * A2208990
*  BY THE  CONTROL PROGRAM IS PRESENTED IN THE FOLLOWING TABLE.   SS  * A2209000
*  CONSISTS OF ONE BYTE, THE BITS OF WHICH ARE DENOTED, FROM LEFT TO  * A2209010
*  RIGHT, AS  B0,B1,...,B7.   BITS WHICH ARE NOT  MENTIONED  IN THIS  * A2209020
*  TABLE HAVE THE VALUE 0.                                            * A2209030
*                                                                     * A2209040
********************************************************************* A2209050
        EJECT                                                         A2209060
********************************************************************* A2209070
*                                                                     * A2209080
*             DEVICE        SPECIAL FEATURE BYTE (SS)                 * A2209090
*             (TTTT)                                                  * A2209100
*             ------        -------------------------                 * A2209110
*                                                                     * A2209120
*              1442         B7=0   NO CARD-IMAGE FEATURE              * A2209130
*              2501,2520    B7=1   CARD-IMAGE FEATURE                 * A2209140
*                                                                     * A2209150
*              1443         B7=0   120 PRINT POSITIONS                * A2209160
*                           B7=1   144 PRINT POSITIONS                * A2209170
*                                                                     * A2209180
*              2540         B7=0   NO COLUMN BINARY FEATURE           * A2209190
*                           B7=1   COLUMN BINARY FEATURE              * A2209200
*                                                                     * A2209210
*              1403         B7=0   100 PRINT POSITIONS                * A2209220
*                           B7=1   132 PRINT POSITIONS                * A2209230
*                                                                     * A2209240
*              1052         ALL ZEROS                                 * A2209250
*                                                                     * A2209260
*              2671         ALL ZEROS                                 * A2209270
*                                                                     * A2209280
*              2311         ALL ZEROS                                 * A2209290
*                                                                     * A2209300
*              2400         B7=1   7-TRACK TAPES                      * A2209310
*                           B6=0   NO DATA CONVERTER                  * A2209320
*                           B6=1   DATA CONVERTER                     * A2209330
*                           B5=1   9-TRACK TAPES                      * A2209340
*                                                                     * A2209350
********************************************************************* A2209360
        SPACE 2                                                       A2209370
********************************************************************* A2209380
*                                                                     * A2209390
*      ROUTINE TO VERIFY TYPE AND CHARACTERISTICS OF A DEVICE         * A2209400
*                                                                     * A2209410
********************************************************************* A2209420
```

```
         SPACE                                                          A2209430
VERIFY   MVC   TEMP(4),8(I)          (TEMP)=                           A2209440
*                                    (DEVSPF.OR.(1...1.XOR.SPFEAT))    A2209450
         BAL   LINK,GETUCB           GET INDEX PAIR (J,K) OF UCB       A2209460
         BC    15,VERIFA             ERROR-UCB DOES NOT EXIST          A2209470
         BC    15,VERIFA             ERROR-UCB DOES NOT EXIST    V1L1  A2209471
         CLC   DEVTYP(4,J),4(I)      IS TYPE=DEVTYP(J)                 A2209480
         BC    7,VERIFA              NO-ERROR,RETURN TO CALLER         A2209490
         XI    TEMP,X'FF'            YES-VERIFY SPECIAL FEATURES       A2209500
         OC    TEMP(1),DEVSPF(J)     *                                 A2209510
         TM    TEMP,X'FF'            IS (TEMP)=1....1                  A2209520
         BC    12,VERIFA             NO-ERROR,RETURN TO CALLER         A2209530
         LA    I,12(I)               YES-VERIFICATION O.K.,RETURN TO   A2209540
VERIFB   ST    I,TEMP                CALLER                            A2209550
VERIFA   MVC   OSVPSW+5(3),TEMP+1    RETURN TO CALLER                  A2209560
         BC    15,ENARET             *                                 A2209570
         EJECT                                                          A2209580
*********************************************************************** A2209590
*                                                                     * A2209600
*      ROUTINE TO DETERMINE INDEX J OF UCB AND INDEX K OF CCB,        * A2209610
*                                GIVEN THE DEVICE ADDRESS             * A2209620
*                                                                     * A2209630
*  THIS ROUTINE DETERMINES,  FROM A PARTICULAR SYSTEM/360 DEVICE AD-  * A2209640
*  DRESS, THE CORRESPONDING INDEX PAIR (J,K).   IT MAY BE ENTERED BY  * A2209650
*  ONE OF TWO CALLING SEQUENCES,                                      * A2209660
*                                                                     * A2209670
*              BAL     LINK,INTUCB                                    * A2209680
*                - OR -                                               * A2209690
*              BAL     LINK,GETUCB                                    * A2209700
*                                                                     * A2209710
*  IN THE FIRST CASE, THE DEVICE ADDRESS WILL BE IN  OIOPSW,  AND IN  * A2209720
*  THE SECOND CASE, THE DEVICE ADDRESS WILL BE AT ADDRESS  I+2.       * A2209730
*                                                                     * A2209740
*  IF A UNIT CONTROL BLOCK  DOES NOT EXIST FOR THIS DEVICE,  CONTROL  * A2209750
*  WILL BE RETURNED  TO ADDRESS  LINK.   OTHERWISE,  CONTROL WILL BE  * A2209760
*  RETURNED TO ADDRESS LINK+4.                                        * A2209770
*                                                                     * A2209780
*  THE ROUTINE USES THE CHANNEL TABLE AND THE CHANNEL CONTROL BLOCK.  * A2209790
*                                                                     * A2209800
*********************************************************************** A2209810
         SPACE                                                          A2209820
INTUCB   LA    I,OIOPSW              OIOPSW TO I                       A2209830
GETUCB   SR    P,P                   4*CHANNEL NUMBER TO P             A2209840
         IC    P,2(I)                *                                 A2209850
         CLI   2(I),X'07'            IS CHANNEL NUMBER LESS THAN 7     A2209860
         BCR   10,LINK               NO-ERROR,RETURN TO (LINK)         A2209870
         SLL   P,2                   CHTAB(P) TO K                     A2209880
         L     K,CHTAB(P)            *                                 A2209890
         LTR   CHPTR,K               IS (K)=0                          A2209900
         BCR   8,LINK                YES ERROR CHANNEL NOT FOUND V1L1  A2209910
*        BCR   8,LINK                YES-ERROR,RETURN  (DELETED) V1L1  A2209915
GETUCA   L     WORK,DEVTAB(CHPTR)    NO-IS LIST EXHAUSTED              A2209920
         LTR   WORK,WORK             *                                 A2209930
         BC    8,4(LINK)             YES ERROR UNIT NOT FOUND    V1L1  A2209940
*        BCR   8,LINK                YES-ERROR,RETURN  (DELETED) V1L1  A2209945
```

```
         CLC   DEVTAB(1,CHPTR),3(I)    NO-IS NEXT DEVICE CORRECT          A2209950
         BC    8,GETUCC                YES-RETURN                         A2209960
         LA    CHPTR,4(CHPTR)          NO-CHPTR+4 TO CHPTR                A2209970
         BC    15,GETUCA               TRY NEXT DEVICE                    A2209980
GETUCC   L     J,DEVTAB(CHPTR)         DEVICE FOUND,(CHPTR) TO J          A2209990
         BC    15,8(LINK)              RETURN LINK+8                V1L1  A2210000
*        BC    15,4(LINK)              RETURN TO (LINK)+4(DELETED) V1L1  A2210005
         EJECT                                                            A2210010
*********************************************************************** A2210020
*                                                                     * A2210030
*  I/O REQUESTS=                                                      * A2210040
*                                                                     * A2210050
*  THREE TYPES OF I/O REQUESTS MAY BE SUBMITTED TO THE CONTROL  PRO-  * A2210060
*  GRAM.   THESE THREE TYPES ARE=                                     * A2210070
*                                                                     * A2210080
*      I/O REQUEST AND WAIT                                           * A2210090
*             THE CONTROL PROGRAM  WILL RETURN CONTROL TO THE SIMU-   * A2210100
*             LATOR ONLY WHEN ALL ACTIVITY RELATIVE TO THE  I/O OP-   * A2210110
*             ERATION HAS BEEN TERMINATED.                            * A2210120
*                                                                     * A2210130
*      I/O REQUEST AND CONTINUE                                       * A2210140
*             THE CONTROL PROGRAM  WILL RETURN CONTROL TO THE SIMU-   * A2210150
*             LATOR  AS SOON AS POSSIBLE  AFTER HAVING ACCEPTED THE   * A2210160
*             REQUEST.  I/O INTERRUPTIONS RELATED TO SUCH A REQUEST   * A2210170
*             WILL INTERRUPT  THE SIMULATOR  AND WILL TRANSFER CON-   * A2210180
*             TROL TO THE  CONTROL  PROGRAM.   THE CONTROL  PROGRAM   * A2210190
*             WILL PRESERVE ALL  INFORMATION RELATED TO THE I/O IN-   * A2210200
*             TERRUPTION AND,  IF THIS  INFORMATION  INDICATES THAT   * A2210210
*             THE I/O REQUEST IS TERMINATED, WILL RETURN CONTROL TO   * A2210220
*             THE SIMULATOR AT A PRE-DETERMINED LOCATION.    OTHER-   * A2210230
*             WISE,  CONTROL WILL BE  RETURNED TO THE  SIMULATOR AT   * A2210240
*             THE POINT OF INTERRUPTION.                              * A2210250
*                                                                     * A2210260
*      I/O REQUEST AND INTERRUPT AT CHANNEL END                       * A2210270
*             THIS REQUEST IS  SIMILAR TO THE  I/O REQUEST AND CON-   * A2210280
*             TINUE, EXCEPT THAT, IN THE ABSENCE OF UNUSUAL  CONDI-   * A2210290
*             TIONS, THE CONDITION CHANNEL END WILL ALSO CAUSE  THE   * A2210300
*             CONTROL PROGRAM TO RETURN CONTROL TO THE SIMULATOR AT   * A2210310
*             A PRE-DETERMINED LOCATION.                              * A2210320
*                                                                     * A2210330
*                                 *****                               * A2210340
*                                                                     * A2210350
*  I/O REQUEST AND WAIT                                               * A2210360
*                                                                     * A2210370
*  THE SVC CALLING SEQUENCE FOR AN I/O REQUEST AND WAIT HAS THE FOL-  * A2210380
*  LOWING FORM=                                                       * A2210390
*                                                                     * A2210400
*             CNOP    4,8                                             * A2210410
*       I     SVC     11                                              * A2210420
*      DEV360 DC      X'0DDD'                                         * A2210430
*      CAWADD DC      A(CCWADD)                                       * A2210440
*      STATUS DS      C                                               * A2210450
*      SNSADD DS      3C                                              * A2210460
*      SVCCSW DS      D                                               * A2210470
*      EXCRET ANY FOUR-BYTE INSTRUCTION (ADDRESS I+20)                 * A2210480
```

```
*       NRMRET ANY INSTRUCTION          (ADDRESS I+24)                  * A2210490
*                                                                       * A2210500
*                                                                .../...* A2210510
*                                                                       * A2210520
************************************************************************ A2210530
         EJECT                                                           A2210540
************************************************************************ A2210550
*                                                                       * A2210560
*  .../...                                                              * A2210570
*                                                                       * A2210580
*  IN THIS CALLING SEQUENCE,  DDD  DENOTES THE SYSTEM/360 ADDRESS OF    * A2210590
*  THE DEVICE FOR WHICH THE I/O REQUEST IS INTENDED, AND CAWADD  DE-    * A2210600
*  NOTES THE ADDRESS OF THE FIRST  CCW TO BE EXECUTED.                  * A2210610
*                                                                       * A2210620
*  THERE IS NO RESTRICTION ON THE CCW'S WHICH MAY BE PRESENTED.   IN    * A2210630
*  PARTICULAR,  A STRING OF CCW'S  CONNECTED BY EITHER DATA CHAINING    * A2210640
*  OR COMMAND CHAINING IS PERMITTED.                                    * A2210650
*                                                                       * A2210660
*  THE BYTE AT ADDRESS  STATUS  IS TREATED BY THE CONTROL PROGRAM AS    * A2210670
*  TWO HEXADECIMAL DIGITS,  STRTBT  AND  ERRTYP.                        * A2210680
*  ON RECEIPT OF THE I/O REQUEST, STRTBT AND ERRTYP WILL BOTH BE SET    * A2210690
*  TO ZERO.                                                             * A2210700
*  STRTBT IS SIGNIFICANT ONLY IN THE I/O REQUEST AND CONTINUE/INTER-    * A2210710
*  RUPT AT CHANNEL END CALLING SEQUENCES.   IN THE  I/O  REQUEST AND    * A2210720
*  WAIT, IT WILL ALWAYS CONTAIN THE VALUE ONE (PHYSICAL I/O OPERATION   * A2210730
*  INITIALIZED AT THE DEVICE) WHEN CONTROL IS RETURNED TO THE  SIMU-    * A2210740
*  LATOR.                                                               * A2210750
*                                                                       * A2210760
*  CONTROL WILL BE RETURNED TO THE SIMULATOR AT ADDRESS  NRMRET WITH    * A2210770
*  ERRTYP=0  (NO EXCEPTIONAL CONDITIONS ENCOUNTERED)  OR AT  ADDRESS    * A2210780
*  EXCRET FOR THE FOLLOWING CONDITIONS=                                 * A2210790
*                                                                       * A2210800
*     -- NO UCB EXISTS FOR THIS DEVICE                                  * A2210810
*           ERRTYP=2                                                    * A2210820
*                                                                       * A2210830
*     -- THE DEVICE  OR ITS ASSOCIATED  CONTROL UNIT,  SUBCHANNEL OR    * A2210840
*        CHANNEL IS IN A NON-OPERATIONAL STATE                          * A2210850
*           ERRTYP=1                                                    * A2210860
*                                                                       * A2210870
*     -- A PROGRAM CHECK OR PROTECTION CHECK  CONDITION HAS BEEN DE-    * A2210880
*        TECTED BY THE CHANNEL                                          * A2210890
*           ERRTYP=3                                                    * A2210900
*                                                                       * A2210910
*     -- A UNIT CHECK CONDITION HAS OCCURRED                            * A2210920
*           ERRTYP=0                                                    * A2210930
*                                                                       * A2210940
*           A SENSE OPERATION WILL BE PERFORMED ON THE DEVICE AND       * A2210950
*             A MAXIMUM OF 3 BYTES OF  SENSE INFORMATION WILL BE        * A2210960
*             STORED, STARTING AT ADDRESS  SNSADD                       * A2210970
*                                                                       * A2210980
*     -- A UNIT EXCEPTION OR CHAINING CHECK CONDITION HAS OCCURRED      * A2210990
*           ERRTYP=0                                                    * A2211000
*                                                                       * A2211010
*     NOTE= THE FIRST TWO OF THESE  EXCEPTIONAL CONDITIONS ARE MUTU-    * A2211020
*           ALLY EXCLUSIVE, BUT THE LAST THREE MAY OCCUR CONCURRENT-    * A2211030
```

```
*          LY.                                                          * A2211040
*                                                                       * A2211050
*                                                                .../.. * A2211060
*                                                                       * A2211070
************************************************************************ A2211080
        EJECT                                                            A2211090
************************************************************************ A2211100
*                                                                       * A2211110
*  .../...                                                              * A2211120
*                                                                       * A2211130
*  EXTERNAL AND I/O INTERRUPTIONS RELATED TO OTHER I/O REQUESTS WILL     * A2211140
*  BE ALLOWED TO OCCUR  DURING THE PERIOD IN WHICH  THE CONTROL PRO-    * A2211150
*  GRAM IS WAITING FOR THE I/O REQUEST AND WAIT TO TERMINATE.   SUCH     * A2211160
*  INTERRUPTIONS WILL BE PROCESSED NORMALLY.                            * A2211170
*                                                                       * A2211180
*  AN I/O REQUEST AND WAIT CALLING SEQUENCE IS NOT ALLOWED WHEN  THE    * A2211190
*  SIMULATOR IS IN THE DISABLED STATE.                                  * A2211200
*                                                                       * A2211210
*                                  *****                                * A2211220
*                                                                       * A2211230
*  I/O REQUEST AND CONTINUE                                             * A2211240
*                                                                       * A2211250
*  THE SVC CALLING SEQUENCE FOR AN  I/O REQUEST AND CONTINUE HAS THE    * A2211260
*  FOLLOWING FORM=                                                      * A2211270
*                                                                       * A2211280
*             CNOP    4,8                                               * A2211290
*       I     SVC     2                                                 * A2211300
*       DEV360 DC     X'0DDD'                                           * A2211310
*       CAWADD DC     A(CCWADD)                                         * A2211320
*       STATUS DS     C                                                 * A2211330
*       SNSADD DS     3C                                                * A2211340
*       SVCCSW DS     D                                                 * A2211350
*       SVCPSW DS     D                                                 * A2211360
*              DC     A(NRMRET)                                         * A2211370
*              DC     A(EXCRET)                                         * A2211380
*       ACCRET ANY INSTRUCTION (ADDRESS I+36)                            * A2211390
*                                                                       * A2211400
*  IN THIS CALLING SEQUENCE,  DDD  DENOTES THE SYSTEM/360 ADDRESS OF    * A2211410
*  THE DEVICE FOR WHICH THE I/O REQUEST IS INTENDED, AND CAWADD  DE-    * A2211420
*  NOTES THE ADDRESS OF THE FIRST  CCW TO BE EXECUTED.                  * A2211430
*                                                                       * A2211440
*  THERE IS NO RESTRICTION ON THE CCW'S WHICH MAY BE PRESENTED.   IN    * A2211450
*  PARTICULAR,  A STRING OF CCW'S  CONNECTED BY EITHER DATA CHAINING    * A2211460
*  OR COMMAND CHAINING IS PERMITTED.                                    * A2211470
*                                                                       * A2211480
*  THE BYTE AT ADDRESS  STATUS  IS TREATED BY THE CONTROL PROGRAM AS    * A2211490
*  TWO HEXADECIMAL DIGITS,  STRTBT  AND  ERRTYP.                        * A2211500
*  ON RECEIPT OF THE I/O REQUEST, STRTBT AND ERRTYP WILL BOTH BE SET    * A2211510
*  TO ZERO.                                                             * A2211520
*  STRTBT WILL BE SET TO  ONE  ONLY WHEN THE  PHYSICAL I/O OPERATION    * A2211530
*  HAS BEEN INITIALIZED AT THE DEVICE.   WHEN CONTROL IS RETURNED TO    * A2211540
*  THE SIMULATOR AT ADDRESS  ACCRET  (REQUEST ACCEPTED), STRTBT WILL    * A2211550
*  INDICATE WHETHER OR NOT THE PHYSICAL I/O OPERATION HAS BEEN  INI-    * A2211560
*  TIALIZED.                                                            * A2211570
*                                                                       * A2211580
```

```
*                                                                 .../...  *
*                                                                          *
****************************************************************************
         EJECT
****************************************************************************
*                                                                          *
*  .../...                                                                 *
*                                                                          *
*  IF ANY OF THE ASSOCIATED CHANNEL, SUBCHANNEL, CONTROL UNIT OR DE-      *
*  VICE IS BUSY, THUS PRECLUDING INITIALIZATION OF THE  I/O REQUEST,      *
*  THE CONTROL PROGRAM WILL PLACE THIS REQUEST ON A QUEUE UNTIL  ALL      *
*  PARTS OF THE DEVICE PATH BECOME FREE.                                  *
*                                                                          *
*  CONTROL WILL BE RETURNED TO THE  SIMULATOR  UNDER ONE OF THE FOL-      *
*  LOWING CONDITIONS=                                                      *
*                                                                          *
*     -- PHYSICAL I/O OPERATION STARTED                                    *
*              STRTBT=1                                                    *
*              RETURN TO ADDRESS  ACCRET                                   *
*                                                                          *
*     -- DEVICE  PATH  (CHANNEL, SUBCHANNEL, CONTROL UNIT OR DEVICE)       *
*          BUSY                                                            *
*              I/O REQUEST ADDED TO REQUEST QUEUE                          *
*              STRTBT=0                                                    *
*              RETURN TO ADDRESS  ACCRET                                   *
*                                                                          *
*     -- PHYSICAL I/O OPERATION STARTED AND TERMINATED WITH NO EXCEP-     *
*          TIONAL CONDITIONS                                               *
*              STRTBT=1                                                    *
*              PLACE  OSVPSW  AT  SVCPSW                                   *
*              ADDRESS  ACCRET  TO THE ADDRESS PART OF  SVCPSW             *
*              RETURN TO ADDRESS  NRMRET  WITH ALL I/O AND  EXTERNAL       *
*                INTERRUPTIONS DISABLED                                    *
*                                                                          *
*     -- EXCEPTIONAL CONDITION HAS PREVENTED THE STARTING OF THE I/O      *
*          OPERATION                                                       *
*            - OR -                                                        *
*        I/O OPERATION  STARTED AND  TERMINATED WITH  AN EXCEPTIONAL       *
*          CONDITION                                                       *
*              STRTBT=1                                                    *
*              PLACE  OSVPSW  AT  SVCPSW                                   *
*              ADDRESS  ACCRET  TO THE ADDRESS PART OF  SVCPSW             *
*              RETURN TO ADDRESS  EXCRET  WITH ALL I/O AND  EXTERNAL       *
*                INTERRUPTIONS DISABLED                                    *
*                                                                          *
*  AN I/O INTERRUPTION  RELATED TO  THIS REQUEST  WILL INTERRUPT THE      *
*  SIMULATOR AND WILL GIVE CONTROL TO THE CONTROL PROGRAM.    IF EX-      *
*  AMINATION OF THE INTERRUPTION INDICATES THAT ALL ACTIVITY RELATED      *
*  TO THE REQUEST  HAS NOT TERMINATED,  CONTROL WILL BE  RETURNED TO      *
*  THE POINT OF INTERRUPTION.  OTHERWISE, CONTROL WILL BE  RETURNED       *
*  TO ONE OF THE ADDRESSES  NRMRET  OR  EXCRET, ACCORDING TO THE CON-     *
*  DITIONS DESCRIBED BELOW.  THE INPUT/OUTPUT OLD PSW WILL BE PLACED      *
*  IN THE DOUBLE WORD WITH ADDRESS  SVCPSW, AND ALL I/O AND EXTERNAL      *
*  INTERRUPTIONS WILL BE DISABLED.                                         *
*                                                                          *
```

```
*                                                              .../...  * A2212140
*                                                                       * A2212150
************************************************************************* A2212160
        EJECT                                                             A2212170
************************************************************************* A2212180
*                                                                       * A2212190
*  .../...                                                              * A2212200
*                                                                       * A2212210
*  THE FOLLOWING EXCEPTIONAL CONDITIONS MAY OCCUR DURING THE  EXECU-    * A2212220
*  TION OF AN I/O REQUEST=                                              * A2212230
*                                                                       * A2212240
*     -- NO UCB EXISTS FOR THIS DEVICE                                  * A2212250
*           ERRTYP=2                                                    * A2212260
*                                                                       * A2212270
*     -- THE DEVICE  OR ITS ASSOCIATED  CONTROL UNIT,  SUBCHANNEL OR   * A2212280
*        CHANNEL IS IN A NON-OPERATIONAL STATE                          * A2212290
*           ERRTYP=1                                                    * A2212300
*                                                                       * A2212310
*     -- A PROGRAM CHECK OR PROTECTION CHECK  CONDITION HAS BEEN DE-    * A2212320
*        TECTED BY THE CHANNEL                                          * A2212330
*           ERRTYP=3                                                    * A2212340
*                                                                       * A2212350
*     -- A UNIT CHECK CONDITION HAS OCCURRED                            * A2212360
*           ERRTYP=0                                                    * A2212370
*                                                                       * A2212380
*           A SENSE OPERATION WILL BE PERFORMED ON THE DEVICE AND       * A2212390
*             A MAXIMUM OF 3 BYTES OF  SENSE INFORMATION WILL BE        * A2212400
*             STORED, STARTING AT ADDRESS  SNSADD                       * A2212410
*                                                                       * A2212420
*     -- A UNIT EXCEPTION OR CHAINING CHECK CONDITION HAS OCCURRED      * A2212430
*           ERRTYP=0                                                    * A2212440
*                                                                       * A2212450
*     NOTE= THE FIRST TWO OF THESE  EXCEPTIONAL CONDITIONS ARE MUTU-    * A2212460
*           ALLY EXCLUSIVE, BUT THE LAST THREE MAY OCCUR CONCURRENT-    * A2212470
*           LY.                                                         * A2212480
*                                                                       * A2212490
*  AN I/O REQUEST AND CONTINUE CALLING SEQUENCE FOR A DEVICE IN  THE    * A2212500
*  BUSY OR CHAINED STATE IS NOT ALLOWED WHEN THE SIMULATOR IS IN THE    * A2212510
*  DISABLED STATE.                                                      * A2212520
*                                                                       * A2212530
*                                *****                                  * A2212540
*                                                                       * A2212550
*                                                              .../...  * A2212560
*                                                                       * A2212570
************************************************************************* A2212580
        EJECT                                                             A2212590
************************************************************************* A2212600
*                                                                       * A2212610
*  .../...                                                              * A2212620
*                                                                       * A2212630
*  I/O REQUEST AND INTERRUPT AT CHANNEL END                             * A2212640
*                                                                       * A2212650
*  THE  SVC CALLING SEQUENCE  FOR  AN  I/O REQUEST  AND INTERRUPT AT    * A2212660
*  CHANNEL END HAS THE FOLLOWING FORM=                                  * A2212670
*                                                                       * A2212680
```

```
*             CNOP    4,8                                               * A2212690
*       I     SVC     1                                                 * A2212700
*       DEV360 DC     X'0DDD'                                           * A2212710
*       CAWADD DC     A(CCWADD)                                         * A2212720
*       STATUS DS     C                                                 * A2212730
*       SNSADD DS     3C                                                * A2212740
*       SVCCSW DS     D                                                 * A2212750
*       SVCPSW DS     D                                                 * A2212760
*             DC      A(NRMRET)                                         * A2212770
*             DC      A(EXCRET)                                         * A2212780
*       ACCRET ANY INSTRUCTION (ADDRESS I+36)                           * A2212790
*                                                                       * A2212800
*  THIS CALLING SEQUENCE PERFORMS THE SAME FUNCTIONS AS THE I/O  RE-    * A2212810
*  QUEST AND CONTINUE, EXCEPT FOR THE FOLLOWING ADDITIONAL  FACILITY    * A2212820
*  OFFERED BY THE I/O REQUEST AND INTERRUPT AT CHANNEL END =            * A2212830
*                                                                       * A2212840
*  WHEN A CHANNEL END CONDITION OCCURS WITHOUT THE DEVICE END CONDI-    * A2212850
*  TION, A TEST IS MADE FOR THE PRESENCE OF A UNIT EXCEPTION OR UNIT    * A2212860
*  CHECK CONDITION.                                                     * A2212870
*                                                                       * A2212880
*  IF NEITHER OF THESE  CONDITIONS IS PRESENT,  THE INPUT/OUTPUT OLD    * A2212890
*  PSW IS PLACED IN THE DOUBLE WORD WITH ADDRESS SVCPSW, AND CONTROL    * A2212900
*  IS RETURNED  TO LOCATION  NRMRET  WITH I/O AND EXTERNAL INTERRUP-    * A2212910
*  TIONS DISABLED.                                                      * A2212920
*                                                                       * A2212930
*  OTHERWISE (CHANNEL END ACCOMPANIED BY  UNIT CHECK OR  UNIT EXCEP-    * A2212940
*  TION), CONTROL IS RETURNED TO THE POINT OF INTERRUPTION.    THUS,     * A2212950
*  ON DEVICES FOR WHICH CHANNEL END AND DEVICE END OCCUR SEPARATELY,    * A2212960
*  TWO RETURNS TO  NRMRET MAY BE EFFECTED.                              * A2212970
*                                                                       * A2212980
*                                  *****                                * A2212990
*                                                                       * A2213000
*  CHANNEL STATUS INFORMATION                                           * A2213010
*                                                                       * A2213020
*  INFORMATION FROM THE CSW'S WHICH MAY BE GENERATED AS A CONSEQUENCE   * A2213030
*  OF THE EXECUTION OF AN  I/O REQUEST IS ACCUMULATED  IN THE DOUBLE    * A2213040
*  WORD  SVCCSW.   ON RECEIPT OF AN I/O REQUEST, THE CONTROL PROGRAM    * A2213050
*  SETS THE CONTENTS OF  SVCCSW  TO ZERO.                               * A2213060
*                                                                       * A2213070
*                                                              .../...  * A2213080
*                                                                       * A2213090
************************************************************************ A2213100
        EJECT                                                             A2213110
************************************************************************ A2213120
*                                                                       * A2213130
*  .../...                                                              * A2213140
*                                                                       * A2213150
*  THE EXECUTION OF A CHAIN OF I/O COMMANDS WILL PRODUCE AT MOST ONE    * A2213160
*  NON-ZERO VALUE FOR EACH OF THE COMMAND ADDRESS AND COUNT PARTS OF    * A2213170
*  A CSW.   (THE CONTROL PROGRAM  WILL IGNORE  ANY SCW IN  WHICH THE    * A2213180
*  PROGRAM  - CONTROLLED INTERRUPTION IS THE ONLY STATUS BIT PRESENT.)  * A2213190
*  THE VALUES OF THESE  QUANTITIES WILL BE SET  INTO THE APPROPRIATE    * A2213200
*  BYTES OF SVCCSW.                                                     * A2213210
*                                                                       * A2213220
*  IF NON-ZERO VALUES OF THE TWO STATUS BYTES ARE  PRODUCED   DURING    * A2213230
```

```
*  THE EXECUTION OF A CHAIN OF I/O COMMANDS, THE CONTROL PROGRAM WILL * A2213240
*  ACCUMULATE THE LOGICAL 'OR' OF THIS STATUS INFORMATION IN THE AP- * A2213250
*  PROPRIATE BYTES OF  SVCCSW.                                       * A2213260
*                                                                    * A2213270
*  IF,  WHEN A CHAIN OF  I/O COMMANDS  HAS TERMINATED,  A UNIT CHECK * A2213280
*  STATUS BIT IS PRESENT IN SVCCSW, THE CONTROL PROGRAM WILL PERFORM * A2213290
*  A SENSE  OPERATION  AND WILL  PLACE A  MAXIMUM OF THREE  BYTES OF * A2213300
*  SENSE INFORMATION STARTING AT ADDRESS SNSADD.                     * A2213310
*                                                                    * A2213320
*  WHEN CONTROL  IS RETURNED  TO THE SIMULATOR  WITH  ERRTYP=1 OR 2, * A2213330
*  SVCCSW WILL ALWAYS CONTAIN ZERO, INDICATING THAT NO I/O OPERATION * A2213340
*  HAS STARTED FOR THE I/O REQUEST.                                  * A2213350
*                                                                    * A2213360
*  WHEN CONTROL  IS RETURNED WITH  ERRTYP=3,  THE I/O  OPERATION HAS * A2213370
*  COMPLETELY  TERMINATED,  AND  SVCCSW  (AND,  IN THE CASE  OF UNIT * A2213380
*  CHECK, THE BYTES AT SNSADD)  WILL DESCRIBE THE STATE OF THIS TER- * A2213390
*  MINATION.                                                         * A2213400
*                                                                    * A2213410
*                                 *****                              * A2213420
*                                                                    * A2213430
*  CHAINING OF I/O REQUESTS                                          * A2213440
*                                                                    * A2213450
*  THE PARTICULAR STATE  OF A DEVICE AT ANY  MOMENT IS DETERMINED BY * A2213460
*  THE CONTROL PROGRAM FROM INFORMATION  IN ITS ASSOCIATED UNIT CON- * A2213470
*  TROL BLOCK.   THE CONTROL PROGRAM WILL TREAT EACH DEVICE AS BEING * A2213480
*  IN ONE OF THE FOLLOWING THREE STATES=                             * A2213490
*                                                                    * A2213500
*      BUSY         THE CONTROL PROGRAM HAS STARTED ACTIVITY FOR SOME * A2213510
*                   I/O REQUEST, BUT THIS ACTIVITY HAS NOT YET TERMI- * A2213520
*                   NATED                                            * A2213530
*                                                                    * A2213540
*      CHAINED      NOT BUSY - AN SVC 1 OR SVC 2 CALLING SEQUENCE HAS * A2213550
*                   BEEN RECEIVED  FOR THE DEVICE,  BUT IT CANNOT YET * A2213560
*                   BE EXECUTED                                      * A2213570
*                                                                    * A2213580
*      AVAILABLE    NOT BUSY AND NOT CHAINED                         * A2213590
*                                                                    * A2213600
*  ANY I/O INTERRUPTION, EXCEPT AN ATTENTION INTERRUPTION,  RECEIVED * A2213610
*  FOR A DEVICE WHICH  IS IN THE AVAILABLE OR  CHAINED STATE WILL BE * A2213620
*  IGNORED.                                                          * A2213630
*                                                            .../... * A2213640
*                                                                    * A2213650
********************************************************************** A2213660
********************************************************************** A2213670
*                                                                    * A2213680
*  .../...                                                           * A2213690
*                                                                    * A2213700
*  AVAILABLE STATE                                                   * A2213710
*                                                                    * A2213720
*          I/O REQUEST AND WAIT                                      * A2213730
*                                                                    * A2213740
*          IF THE DEVICE FOR WHICH THE REQUEST IS RECEIVED IS IN THE * A2213750
*          AVAILABLE STATE,  THE CONTROL PROGRAM  WILL TRY  TO START * A2213760
*          THE CORRESPONDING  I/O OPERATION.   IF THE STATUS  OF ANY * A2213770
*          OF THE CHANNEL,  SUBCHANNEL,  CONTROL UNIT OR DEVICE PRE- * A2213780
```

```
*          CLUDES INITIATION  OF THE OPERATION,  THE CONTROL PROGRAM  * A2213790
*          WILL 'CYCLE'(*) ON THE SVC CALLING SEQUENCE UNTIL THE OP-  * A2213800
*          ERATION IS ACCEPTED.  OTHERWISE,  THE OPERATION  WILL BE   * A2213810
*          STARTED,  THE BUSY STATE WILL BE SET AND THE CONTROL PRO-  * A2213820
*          GRAM WILL CYCLE ON THE SVC CALLING SEQUENCE UNTIL ALL RE-  * A2213830
*          LATED I/O INTERRUPTIONS HAVE BEEN RECEIVED AND PROCESSED.  * A2213840
*                                                                     * A2213850
*  (*) TO CYCLE MEANS  THAT THE CONTROL  PROGRAM WILL  PLACE THE AD-  * A2213860
*      DRESS OF  THE SVC INSTRUCTION  INTO THE  ADDRESS PART  OF THE  * A2213870
*      SUPERVISOR-CALL OLD PSW AND WILL THEN LOAD THIS PSW.    THUS,  * A2213880
*      THE CONTROL PROGRAM WILL RETURN TO THE SVC INSTRUCTION, WHICH  * A2213890
*      WILL BE REPEATED UNTIL THE OPERATION CAN BE ACCEPTED.          * A2213900
*                                                                     * A2213910
*          I/O REQUEST AND CONTINUE/INTERRUPT AT CHANNEL END          * A2213920
*                                                                     * A2213930
*          IF THE DEVICE FOR WHICH THE REQUEST IS RECEIVED IS IN THE  * A2213940
*          AVAILABLE STATE,  THE CONTROL  PROGRAM WILL TRY  TO START  * A2213950
*          THE CORRESPONDING I/O OPERATION.   IT WILL SET EITHER THE  * A2213960
*          BUSY STATE (OPERATION STARTED)  OR THE CHAINED STATE (OP-  * A2213970
*          ERATION WAITING) AND WILL RETURN CONTROL TO THE SIMULATOR  * A2213980
*          AT ADDRESS ACCRET.                                         * A2213990
*                                                                     * A2214000
*                                                                     * A2214010
*  BUSY OR CHAINED STATE                                              * A2214020
*                                                                     * A2214030
*  ANY I/O  REQUEST RECEIVED  FOR A DEVICE  IN THE  BUSY OR  CHAINED  * A2214040
*  STATE WILL CAUSE THE CONTROL PROGRAM TO CYCLE ON THE NEW SVC CAL-   * A2214050
*  LING SEQUENCE.                                                     * A2214060
*                                                                     * A2214070
*                                  *****                              * A2214080
*                                                                     * A2214090
*  SETTING UP OF THE SEREP INTERFACE                                  * A2214100
*                                                                     * A2214110
*  IF CERTAIN  'CATASTROPHIC'  CONDITIONS ARE ENCOUNTERED DURING THE  * A2214120
*  EXECUTION OF AN  I/O REQUEST,  THERE WILL BE  NO RETURN  FROM THE  * A2214130
*  CONTROL PROGRAM  TO THE SIMULATOR.  INSTEAD,  THE STANDARD  SEREP  * A2214140
*  INTERFACE WILL BE SET UP.                                          * A2214150
*                                                                     * A2214160
*                                                          .../...    * A2214170
*                                                                     * A2214180
********************************************************************** A2214190
        EJECT                                                          A2214200
********************************************************************** A2214210
*                                                                     * A2214220
*  .../...                                                            * A2214230
*                                                                     * A2214240
*  DURING THE EXECUTION OF AN I/O REQUEST, ONE OF THE FOLLOWING CON-  * A2214250
*  DITIONS MAY OCCUR=                                                 * A2214260
*                                                                     * A2214270
*     -- ONE OR MORE  OF THE  CHANNEL  STATUS  INDICATIONS - CHANNEL  * A2214280
*        CONTROL CHECK, INTERFACE CONTROL CHECK, CHANNEL DATA CHECK -  * A2214290
*        IS DETECTED                                                  * A2214300
*     -- A CHANNEL OR DEVICE  STATUS INDICATION WHICH SHOULD NOT OC-  * A2214310
*        CUR IS DETECTED                                              * A2214320
*     -- A SENSE OPERATION  CANNOT BE PERFORMED ON A DEVICE  (SUCH A  * A2214330
```

```
*         SENSE OPERATION IS ATTEMPTED  EACH TIME THE EXECUTION OF AN  * A2214340
*         I/O REQUEST GIVES RISE TO A UNIT CHECK CONDITION)            * A2214350
*                                                                      * A2214360
*  IN ALL THE ABOVE  SITUATIONS,  THE CONTROL PROGRAM WILL SET UP IN   * A2214370
*  MAIN STORAGE THE ELEMENTS NECESSARY FOR THE STANDARD SEREP INTER-   * A2214380
*  FACE.  IT WILL LOAD A PSW IN WHICH I/O AND EXTERNAL INTERRUPTIONS   * A2214390
*  ARE DISABLED,  IN WHICH  THE WAIT STATE BIT IS ONE  AND FOR WHICH   * A2214400
*  THE INTERRUPTION CODE BITS ARE ALL ONES.                            * A2214410
*                                                                      * A2214420
*  IN ALL OTHER CASES, CONTROL WILL BE RETURNED TO THE SIMULATOR.      * A2214430
*                                                                      * A2214440
*  THE SIMULATOR  MAY FIND IT NECESSARY,  AS A RESULT  OF THE CONDI-   * A2214450
*  TIONS UNDER WHICH  AN I/O REQUEST  HAS TERMINATED,  TO SET UP THE   * A2214460
*  SEREP INTERFACE (FOR EXAMPLE, THE CONDITION ERRTYP=1 MAY BE INTER-  * A2214470
*  PRETED AS A SEREP CONDITION).                                       * A2214480
*                                                                      * A2214490
*  THE FOLLOWING CALLING SEQUENCE SHOULD BE USED IN THE SIMULATOR TO   * A2214500
*  REQUEST THAT THE CONTROL PROGRAM SET UP THE STANDARD SEREP INTER-   * A2214510
*  FACE=                                                               * A2214520
*                                                                      * A2214530
*              CNOP    2,4                                             * A2214540
*       I      SVC     7                                               * A2214550
*       TYPE   DC      X'TT'                                           * A2214560
*              DC      AL3(IOREQ)                                      * A2214570
*                                                                      * A2214580
*  TT  DENOTES THE TYPE OF INTERFACE WHICH IS REQUIRED.   THUS=        * A2214590
*                                                                      * A2214600
*       TT=0F  INDICATES A CHANNEL FAILURE                             * A2214610
*       TT=1F  INDICATES A DEVICE FAILURE                              * A2214620
*       TT=3F  INDICATES A DEVICE-NOT-OPERATIONAL CONDITION            * A2214630
*                                                                      * A2214640
*  IOREQ  DENOTES THE ADDRESS OF THE  SVC INSTRUCTION IN THE CALLING   * A2214650
*  SEQUENCE OF THAT I/O REQUEST WHICH GAVE RISE TO THIS SEREP CONDI-   * A2214660
*  TION.                                                               * A2214670
*                                                                      * A2214680
*                              *****                                   * A2214690
*                                                                      * A2214700
*                                                          .../...     * A2214710
*                                                                      * A2214720
************************************************************************ A2214730
        EJECT                                                           A2214740
************************************************************************ A2214750
*                                                                      * A2214760
*  .../...                                                             * A2214770
*                                                                      * A2214780
*  REWIND AND REWIND AND UNLOAD CALLING SEQUENCES                      * A2214790
*                                                                      * A2214800
*  WHEN AN I/O REQUEST AND CONTINUE  CALLING SEQUENCE IS USED TO RE-   * A2214810
*  WIND OR TO  REWIND AND UNLOAD A  2400 SERIES  MAGNETIC TAPE UNIT,   * A2214820
*  THE OPERATION WILL NORMALLY BE TERMINATED  (AND THE SIMULATOR IN-    * A2214830
*  TERRUPTED) ONLY WHEN THE  DEVICE END  SIGNAL IS RECEIVED FROM THE   * A2214840
*  TAPE UNIT.                                                          * A2214850
*                                                                      * A2214860
*  THE TWO SVC CALLING SEQUENCES BELOW  WILL ENABLE THE CONTROL PRO-   * A2214870
*  GRAM TO TERMINATE  THE OPERATION WHEN THE  CHANNEL END  SIGNAL IS   * A2214880
```

```
*  RECEIVED.    IN THIS CASE,  I/O INTERRUPTIONS  FOR THE  TAPE UNIT  * A2214890
*  WHICH OCCUR AFTER THE  CHANNEL END  SIGNAL HAS BEEN RECEIVED WILL  * A2214900
*  BE IGNORED.                                                        * A2214910
*                                                                     * A2214920
*  THE FOLLOWING  SVC CALLING SEQUENCES ARE USED FOR  THE REWIND AND  * A2214930
*  THE REWIND AND UNLOAD FUNCTIONS=                                   * A2214940
*                                                                     * A2214950
*             CNOP    4,8                                             * A2214960
*       I     SVC     13  (REWIND)                                    * A2214970
*            -OR-                                                     * A2214980
*       I     SVC     14  (REWIND AND UNLOAD)                         * A2214990
*       DEV360 DC     X'0DDD'                                         * A2215000
*       CAWADD DC     A(CCWADD)                                       * A2215010
*       STATUS DS     C                                               * A2215020
*       SNSADD DS     3C                                              * A2215030
*       SVCCSW DS     D                                               * A2215040
*       SVCPSW DS     D                                               * A2215050
*              DC     A(NRMRET)                                       * A2215060
*              DC     A(EXCRET)                                       * A2215070
*       ACCRET ANY INSTRUCTION (ADDRESS I+36)                         * A2215080
*                                                                     * A2215090
*  DDD REPRESENTS THE SYSTEM/360 ADDRESS OF THE 2400 SERIES MAGNETIC  * A2215100
*  TAPE UNIT INVOLVED IN THE OPERATION,  AND  CAWADD DENOTES THE AD-  * A2215110
*  DRESS OF THE FIRST CCW TO BE EXECUTED.                             * A2215120
*                                                                     * A2215130
*  WHEN A CHANNEL END CONDITION OCCURS WITHOUT DEVICE END,  THE FOL-  * A2215140
*  LOWING TESTS ARE MADE=                                             * A2215150
*                                                                     * A2215160
*      REWIND              HAS A UNIT EXCEPTION OR A UNIT CHECK CON-  * A2215170
*                          DITION OCCURRED                            * A2215180
*      REWIND AND UNLOAD   HAS A UNIT  EXCEPTION  CONDITION OCCURRED  * A2215190
*                                                                     * A2215200
*  IF NOT, CONTROL IS RETURNED TO THE SIMULATOR AT  LOCATION  NRMRET  * A2215210
*  AND THE DEVICE END CONDITION WILL BE IGNORED.                      * A2215220
*                                                                     * A2215230
*  OTHERWISE,  THE  TERMINATION  OF  THIS  OPERATION IS IDENTICAL TO  * A2215240
*  THAT OF THE I/O REQUEST AND CONTINUE OPERATION.                    * A2215250
*                                                          .../...    * A2215260
*                                                                     * A2215270
********************************************************************** A2215280
         EJECT                                                         A2215290
********************************************************************** A2215300
*                                                                     * A2215310
*  .../...                                                            * A2215320
*                                                                     * A2215330
*  THE CONTROL PROGRAM  MAKES  NO CHECK FOR THE VALIDITY OF THE COM-  * A2215340
*  MAND CODE IN THE SIMULATOR-PROVIDED CCW.  THUS,  IN THE SIMULATOR  * A2215350
*  A COMMAND  CODE  CORRESPONDING  TO THE OPERATION  TO BE PERFORMED  * A2215360
*  MUST BE PLACED  IN THE CCW.  IF NOT,  THE  CONTROL  PROGRAM  WILL  * A2215370
*  TREAT THE CALLING SEQUENCE  AS AN I/O REQUEST  AND CONTINUE,  BUT  * A2215380
*  WILL TERMINATE THE OPERATION AS A REWIND OR REWIND AND UNLOAD.     * A2215390
*                                                                     * A2215400
********************************************************************** A2215410
         EJECT                                                         A2215420
********************************************************************** A2215430
```

```
*                                                                     * A2215440
*      I/O REQUEST AND WAIT ROUTINE                                   * A2215450
*                                                                     * A2215460
*  THIS ROUTINE CONSISTS OF TWO DISTINCT PARTS.   IN THE FIRST PART,  * A2215470
*  THE I/O OPERATION IS STARTED (IF POSSIBLE) AND ANY IMMEDIATE CON-  * A2215480
*  DITIONS ARE PROCESSED.   CONTROL IS THEN RETURNED  TO THE CALLING  * A2215490
*  PROGRAM.                                                           * A2215500
*                                                                     * A2215510
*  THE SECOND PART IS ENTERED VIA THE I/O INTERRUPTION ROUTINE WHEN-  * A2215520
*  EVER AN  ASSOCIATED  I/O INTERRUPTION  OCCURS.   THE INTERRUPTION  * A2215530
*  CONDITIONS ARE PROCESSED AND CONTROL IS GIVEN TO THE  UNSTAK ROU-  * A2215540
*  TINE WITH  OIOPSW  CONTAINING THE  PSW  WHICH IS TO  BE LOADED IN  * A2215550
*  ORDER TO RETURN CONTROL TO THE CALLING PROGRAM.                    * A2215560
*                                                                     * A2215570
********************************************************************** A2215580
         SPACE                                                         A2215590
IOWAIT   BAL   LINK,GETUCB          GET INDEX PAIR (J,K) OF UCB       A2215600
*        BC    15,IOWERR            ERROR-DEV NOT FOUN(DELETED) V1L1  A2215610
         BC    15,IOWERR            ERROR-CHANNEL NOT FOUND     V1L1  A2215612
         BC    15,IOWERR            ERROR-DEVICE NOT FOUND      V1L1  A2215615
         TM    SENSW,X'01'          IS SENSE SWITCH ON                A2215620
         BC    5,IOWSNS             YES-BRANCH                        A2215630
         BAL   LINK,STRTIO          NO-TRY TO START I/O OPERATION     A2215640
         BC    15,IOWTRM            OPERATION TERMINATED              A2215650
         BC    15,IOWBSY            DEVICE BUSY                       A2215660
         BC    15,IOWBSY            PATH BUSY                         A2215670
         BC    15,IOWEXC            EXCEPTIONAL CONDITION             A2215680
         BC    0,0                  CHANNEL END                       A2215690
         MVC   DEVINT+1(3,J),AIOWIN START-SET DEVINT(J)                 A2215700
IOWBSY   ST    I,OSVPSW+4           I TO OSVPSW(A)                    A2215710
         BC    15,ENARET            RETURN TO CALLER                  A2215720
IOWTRM   AH    I,DEC24              SET RETURN ADDRESS=I+24           A2215730
         BC    15,IOWBSY            RETURN TO CALLER                  A2215740
IOWERR   MVI   STATUS(I),X'12'      SET ERRTYP(I)=2,STRTBT(I)=1       A2215750
         XC    SVCCSW(8,I),SVCCSW(I) SET SVCCSW(I)=0                   A2215760
IOWEXC   AH    I,DEC20              SET RETURN ADDRESS=I+20           A2215770
         BC    15,IOWBSY            RETURN TO CALLER                  A2215780
IOWSNS   CLI   BORCH(J),SNSFLG      SENSE OPN. PERFORMED ON DEVICE    A2215790
         BC    8,IOWBSY             YES-CYCLE ON SVC                  A2215800
         NI    SENSW,X'FE'          NO-SET SENSE SWITCH=0             A2215810
         BC    15,IOWEXC            SENSE FINISHED - RETURN TO        A2215820
*                                     CALLER                          A2215830
         SPACE                                                         A2215840
********************************************************************** A2215850
*      ENTER HERE WHEN ANY INTERRUPTION FOR DEVICE J HAS OCCURRED     * A2215860
********************************************************************** A2215870
         SPACE                                                         A2215880
IOWINT   BC    15,IOWTER            INTERRUPTION TERMINATED           A2215890
         BC    15,IOWCON            OTHER CONDITION-TREAT AS CHANNEL  A2215900
*                                     END                             A2215910
         BC    15,IOWEX             EXCEPTIONAL CONDITION             A2215920
         BC    15,IOWSN             SENSE OPERATION                   A2215930
IOWCON   ST    I,OIOPSW+4           CHANNEL END-I TO OSVPSW(A)        A2215940
         BC    15,UNSTAK            CALL UNSTAK                       A2215950
IOWTER   AH    I,DEC24              SET RETURN ADDRESS=I+24           A2215960
```

```
         BC    15,IOWCON              CONTINUE                          A2215970
IOWEX    AH    I,DEC20                SET RETURN ADDRESS=I+20           A2215980
         BC    15,IOWCON              CONTINUE                          A2215990
IOWSN    OI    SENSW,X'01'            SET SENSE SWITCH=1                A2216000
         BC    15,IOWCON              CONTINUE                          A2216010
         EJECT                                                          A2216020
*********************************************************************** A2216030
*                                                                     * A2216040
*      I/O REQUEST AND CONTINUE ROUTINE                               * A2216050
*                                                                     * A2216060
*  THIS ROUTINE CONSISTS OF TWO DISTINCT PARTS.   IN THE FIRST PART,  * A2216070
*  THE I/O OPERATION IS STARTED (IF POSSIBLE) AND ANY IMMEDIATE CON-  * A2216080
*  DITIONS ARE PROCESSED.   CONTROL IS THEN RETURNED  TO THE CALLING  * A2216090
*  PROGRAM.                                                           * A2216100
*                                                                     * A2216110
*  THE SECOND PART IS ENTERED VIA THE I/O INTERRUPTION ROUTINE WHEN-  * A2216120
*  EVER AN  ASSOCIATED  I/O INTERRUPTION  OCCURS.   THE INTERRUPTION  * A2216130
*  CONDITIONS ARE PROCESSED AND CONTROL IS GIVEN TO THE  UNSTAK ROU-  * A2216140
*  TINE WITH  OIOPSW  CONTAINING THE  PSW  WHICH IS TO  BE LOADED IN  * A2216150
*  ORDER TO RETURN CONTROL TO THE CALLING PROGRAM.                    * A2216160
*                                                                     * A2216170
*********************************************************************** A2216180
         SPACE                                                          A2216190
IOCONT   BAL   LINK,GETUCB            GET INDEX PAIR (J,K) OF UCB       A2216200
         BC    15,IOCFA               ERROR-DEVICE NOT FOUND            A2216210
         BC    15,IOCFA               ERROR CHANNEL NOT FOUND     V1L1  A2216215
         BAL   LINK,STRTIO            TRY TO START I/O OPERATION        A2216220
         BC    15,IOCD                OPERATION TERMINATED              A2216230
         BC    15,IOCB                DEVICE BUSY                       A2216240
         BC    15,IOSTAK              PATH BUSY                         A2216250
         BC    15,IOCG                EXCEPTIONAL CONDITION - SENSE     A2216260
*                                       OPERATION IF UC PRESENT         A2216270
         BC    15,IOCHEN              CHANNEL END                       A2216280
IOCA     AH    I,DEC36                START-I+36 TO I                   A2216290
         MVC   DEVINT+1(3,J),AIOCIN   START-SET DEVINT(J)               A2216300
IOCB     ST    I,OSVPSW+4             I TO OSVPSW(A)                    A2216310
IOCRET   BC    15,ENARET              RETURN TO CALLER                  A2216320
IOCHEN   CLI   1(I),X'02'             IS THIS AN 'SVC 2' CALL           A2216330
         BC    8,IOCA                 YES-GO TO START                   A2216340
         TM    CSW+4,UCORUE           NO-ANY UC OR UE PRESENT           A2216350
         BC    5,IOCA                 YES-GO TO START                   A2216360
         CLI   1(I),X'0D'             IS THIS AN 'SVC 14/13' CALL       A2216370
         BC    4,*+8                  NO- 'SVC 1' CALL- BRANCH          A2216380
         MVI   BORCH(J),FREFLG        YES-BORCH(J)=0 - IGNORE DE        A2216390
         MVC   DEVINT+1(3,J),AIOCIN   SET DEVINT(J)                     A2216400
IOCD     MVC   OSVPSW+5(3),29(I)      NHRET TO OSVPSW(A)                A2216410
IOCE     MVC   SVCPSW(8,I),OSVPSW     OSVPSW TO SVCPSW(I)               A2216420
IOCF     LR    WORK,I                 I+36 TO SVCPSW(I) ADDRESS         A2216430
         AH    WORK,DEC36             *                                 A2216440
         ST    WORK,SVCPSW+4(I)       *                                 A2216450
         NI    OSVPSW,X'00'           DISABLE I/O AND EXTERNAL INT.     A2216460
         BC    15,ENARET              RETURN TO CALLER                  A2216470
IOCFA    MVI   STATUS(I),X'12'        SET STRBT(I)=1-ERRTYP(I)=2        A2216480
         XC    SVCCSW(8,I),SVCCSW(I)  SET SVCCSW(I)=0                   A2216490
IOCG     MVC   OSVPSW+5(3),33(I)      EXCRET TO OSVPSW(A)               A2216500
```

```
         BC    15,IOCE                 RETURN TO CALLER                A2216510
IOSTAK   BAL   LINKA,STACK             ADD J TO CHANNEL CHAIN          A2216520
         BC    15,IOCA                 GO TO START                     A2216530
         EJECT                                                         A2216540
*********************************************************************** A2216550
*      ENTER HERE WHEN ANY INTERRUPTION FOR DEVICE J HAS OCCURRED     * A2216560
*********************************************************************** A2216570
         SPACE                                                         A2216580
IOCINT   BC    15,IOCH                 INTERRUPTION TERMINATED         A2216590
         BC    15,UNSTAK               OTHER CONDITION-GO TO UNSTAK    A2216600
         BC    15,IOCK                 EXCEPTIONAL PROGRAM ERROR OR    A2216610
*                                       TERMINATION WITH UE             A2216620
         BC    15,UNSTAK               SENSE OPERATION CHAINED FOR THE A2216630
*                                       DEVICE-GO TO UNSTAK             A2216640
         CLI   1(I),X'02'              CHANNEL END-IS THIS AN 'SVC 2'  A2216650
*                                       CALL                            A2216660
         BC    8,UNSTAK                YES-GO TO UNSTAK                A2216670
         TM    CSW+4,UCORUE            NO-ANY UC OR UE PRESENT         A2216680
         BC    5,UNSTAK                YES-GO TO UNSTAK                A2216690
         CLI   1(I),X'0D'              NO- IS THIS AN 'SVC 13/14' CALL A2216700
         BC    4,*+8                   NO- 'SVC 1' CALL- BRANCH        A2216710
         MVI   BORCH(J),FREFLG         YES-BORCH(J)=0 - IGNORE DE      A2216720
IOCH     MVC   SVCPSW(8,I),OIOPSW      OIOPSW TO SVCPSW(I)             A2216730
         MVC   OIOPSW+5(3),29(I)       NMRET TO OIOTSW(A)              A2216740
IOCJ     NI    OIOPSW,X'00'            DISABLE I/O AND EXTERNAL INT.   A2216750
         NI    SVCPSW+1(I),X'FD'       0 TO WAIT STATE BIT IN SVCPSW   A2216760
         BC    15,UNSTAK               GO TO UNSTAK                    A2216770
IOCK     MVC   SVCPSW(8,I),OIOPSW      OIOPSW TO SVCPSW(J)             A2216780
         MVC   OIOPSW+5(3),33(I)       EXCRET TO OIOPSW(A)             A2216790
         BC    15,IOCJ                 CONTINUE                        A2216800
         EJECT                                                         A2216810
*********************************************************************** A2216820
*                                                                     * A2216830
*      STRTIO ROUTINE                                                 * A2216840
*                                                                     * A2216850
*  THIS ROUTINE  WILL BE CALLED  IN ORDER TO BEGIN  THE PHYSICAL I/O  * A2216860
*  ACTIVITY RELATED TO A PARTICULAR SVC INDEX  I.   INPUT PARAMETERS  * A2216870
*  TO THIS ROUTINE ARE  THE THREE INDICES  I, J AND K.   THE CALLING  * A2216880
*  SEQUENCE IS OF THE FORM=                                           * A2216890
*                                                                     * A2216900
*              BAL     LINK,STRTIO                                    * A2216910
*       TERM   BC      15,ADDR1     (ADDRESS  LINK)                   * A2216920
*       DEVBSY BC      15,ADDR2     (ADDRESS  LINK+4)                 * A2216930
*       PATHB  BC      15,ADDR3     (ADDRESS  LINK+8)                 * A2216940
*       EXCEPT BC      15,ADDR4     (ADDRESS  LINK+12)                * A2216950
*       CHEND  BC      15,ADDR5     (ADDRESS  LINK+16)                * A2216960
*       START  ANY INSTRUCTION      (ADDRESS  LINK+20)                * A2216970
*                                                                     * A2216980
*  CONTROL WILL BE RETURNED TO ONE OF THE ADDRESSES  LINK, LINK+4,... * A2216990
*  LINK+20, DEPENDING ON THE CONDITIONS DETECTED BY THE ROUTINE. THE  * A2217000
*  ROUTINE WILL THEN BRANCH TO THE APPROPRIATE SEQUENCE IN ORDER  TO  * A2217010
*  PROCESS THESE CONDITIONS.                                          * A2217020
*                                                                     * A2217030
*      TERM       THE OPERATION HAS BEEN STARTED AND TERMINATED WITH- * A2217040
*                 OUT UNIT CHECK,  UNIT EXCEPTION OR OTHER ERROR CON- * A2217050
```

```
*                DITIONS                                                * A2217060
*                                                                       * A2217070
*       DEVBSY   THE DEVICE IS BUSY WITH A  PREVIOUS OPERATION OR IS    * A2217080
*                IN THE CHAINED STATE                                   * A2217090
*                                                                       * A2217100
*       PATHB    THE CHANNEL, SUBCHANNEL, CONTROL UNIT  OR DEVICE IS    * A2217110
*                BUSY,  OR THE OPERATION  MUST BE  DELAYED   BECAUSE    * A2217120
*                SOME OUTSTANDING SENSE REQUESTS HAVE NOT YET TERMI-    * A2217130
*                NATED                                                  * A2217140
*                                                                       * A2217150
*       EXCEPT   ONE OF THE FOLLOWING EXCEPTIONAL CONDITIONS HAS OC-    * A2217160
*                CURRED=                                                * A2217170
*                  -- THE OPERATION  CANNOT BE STARTED DUE TO A UNIT    * A2217180
*                     CHECK OR UNIT EXCEPTION CONDITION ON THE DEVICE   * A2217190
*                  -- THE OPERATION  HAS BEEN STARTED AND TERMINATED    * A2217200
*                     WITH A UNIT CHECK OR UNIT EXCEPTION CONDITION     * A2217210
*                  -- THE DEVICE IS NOT OPERATIONAL                     * A2217220
*                  -- A PROGRAM CHECK CONDITION OCCURRED AT INITIAL-    * A2217230
*                     IZATION                                           * A2217240
*                                                                       * A2217250
*       CHEND    THE OPERATION  HAS STARTED  AND  GAVE AN  IMMEDIATE    * A2217260
*                CHANNEL END CONDITION                                  * A2217270
*                                                                       * A2217280
*       START    THE OPERATION  HAS STARTED  WITHOUT  ANY  IMMEDIATE    * A2217290
*                STATUS CONDITIONS                                      * A2217300
*                                                                       * A2217310
************************************************************************ A2217320
         SPACE                                                          A2217330
STRTIO   TM    BORCH(J),BSYFLG          BORCH(J)=01,10,11               A2217340
         BC    5,4(LINK)                YES-RETURN TO (LINK)+4          A2217350
         XC    SNSADD(11,I),SNSADD(I)   NO-0 TO SNSADD(I),0 TO SVCCSW(I) A2217360
         MVI   STATUS(I),X'10'          0 TO ERRTYP,1 TO STRTBT(I)      A2217370
         ST    I,DEVSVC(J)              I TO DEVSVC(J)                  A2217380
         TM    SNSCNT,ONES              ANY SENSE OPERATION TO BE       A2217390
*                                         PERFORMED ON THE DEVICE       A2217400
         BC    5,SIOBUS                 YES-PATH BUSY,RETURN TO (LINK)+8 A2217410
         MVC   CAW(4),CAWADD(I)         CCWADD(I) TO CAW(A)             A2217420
         LH    DEVICE,DEV360(J)         S/360 DEVICE ADDRESS TO DEVICE  A2217430
SIODEV   SIO   0(DEVICE)                TRY TO START I/O                A2217440
         BC    8,SIOOK                  CC=0-SIO ACCEPTED               A2217450
         BC    4,SIOCSW                 CC=1-CSW STORED                 A2217460
         BC    2,SIOBUS                 CC=2-PATH BUSY                  A2217470
SIOERA   OI    ERRTYP(I),X'01'          CC=3-NOT OPERATIONAL,X'01' TO   A2217480
*                                         ERRTYP                        A2217490
         BC    15,12(LINK)              RETURN TO (LINK)+12             A2217500
SIOBUS   NI    STRTBT(I),X'EF'          0 TO STRTBT(I)                  A2217510
         BC    15,8(LINK)               RETURN TO (LINK)+8              A2217520
SIOOK    OI    BORCH(J),BSYFLG          11 TO BORCH(J)                  A2217530
         BC    15,20(LINK)              RETURN TO (LINK)+20             A2217540
SIOCSW   TM    CSW+5,CHERST             ANY OF PRC,DC,CCC,ICC,CHC       A2217550
         BC    5,SREP0S                 YES-CHANNEL FAILURE,SET UP      A2217560
         MVC   *+7(1),INVST(J)          SEREP INTERFACE                 A2217570
         TM    CSW+4,X'00'              NO-ANY INVALID STATUS BITS      A2217580
         BC    5,SREP1S                 YES-DEVICE FAILURE,SET UP SEREP A2217590
*                                         INTERFACE                     A2217600
```

```
         TM    CSW+5,PC              NO-IS PROGRAM CHECK BIT ON       A2217610
         BC    8,SIODST              NO-EXAMINE STATUS                A2217620
         OI    ERRTYP(I),X'03'       YES-SET PROGRAM CHECK INDICATION A2217630
         BC    15,12(LINK)           RETURN TO (LINK)+12              A2217640
SIODST   TM    CSW+4,B               IS BUSY BIT=1                    A2217650
         BC    8,SIOB                NO-GO TO TEST CHANNEL END        A2217660
         TM    CSW+4,SM              YES-IS STATUS MODIFIER BIT PRES. A2217670
         BC    5,SIOBUS              YES-CU BUSY,RETURN TO (LINK)+8   A2217680
         BC    15,SIODEV             NO-IGNORE OTHER STATUS BITS      A2217690
SIOB     TM    CSW+4,CE              IS CHANNEL BIT PRESENT           A2217700
         BC    1,SIOBB               YES-BRANCH.  NO,                 A2217710
         TM    CSW+4,CUE+DE          NO-IS CUE AND/OR DE BITS PRESENT A2217720
         BC    5,SIODEV              YES-IGNORE                       A2217730
SIOBB    OC    SVCCSW+4(1,I),CSW+4   DEVICE STATUS TO SVCCSW(I)       A2217740
         TM    CSW+4,CE              IS CHANNEL END PRESENT           A2217750
         BC    8,SIOD                NO-OPERATION NOT STARTED         A2217760
         TM    CSW+4,DE              YES-IMMEDIATE OPERATION - IS DE  A2217770
*                                      BIT PRESENT                    A2217780
         BC    5,SIOD                YES-TEST FOR UC OR UE            A2217790
         TM    CSW+4,UC              NO-OPERATION NOT FINISHED - IS   A2217800
*                                      UC BIT PRESENT                 A2217810
         BC    5,SIOC                YES-GO TO TEST RUN CALL          A2217820
         OI    BORCH(J),BSYFLG       NO-CHANNEL END,11 TO BORCH(J)    A2217830
         BC    15,16(LINK)           RETURN TO (LINK)+16              A2217840
SIOC     CLI   1(I),X'0E'            IS THIS A REWIND AND UNLOAD CALL A2217850
         BCR   8,LINK                YES- UC, NORMAL CONDITION        A2217860
         IC    WORK,SNSCNT           NO-INCREMENT SNSCNT BY ONE -     A2217870
         LA    WORK,1(WORK)          (NUMBER OF SENSE OPERATIONS      A2217880
         STC   WORK,SNSCNT           TO BE PERFORMED)                 A2217890
         MVI   BORCH(J),SNSFLG       10 TO BORCH(J)                   A2217900
         BC    15,16(LINK)           RETURN TO (LINK)+16              A2217910
SIOD     TM    CSW+4,UC              IS UC BIT PRESENT                A2217920
         BC    5,SIOSNS              YES-GO TO SENSE                  A2217930
         TM    CSW+4,UE              NO-IS UE BIT PRESENT             A2217940
         BC    5,12(LINK)            YES-RETURN TO (LINK)+12          A2217950
         BCR   15,LINK               NO-TERMINATED,RETURN TO (LINK)   A2217960
SIOSNS   BAL   LINKA,SENSE           PICK UP SENSE BYTES              A2217970
         BC    0,0                   NOP                              A2217980
         BC    15,12(LINK)           SENSE TERMINATED,RETURN TO       A2217990
*                                      (LINK)+12                      A2218000
         EJECT                                                        A2218010
*************************************************************************A2218020
*                                                                     * A2218030
*      IOINT ROUTINE                                                  * A2218040
*                                                                     * A2218050
*  THIS ROUTINE IS ENTERED DIRECTLY VIA THE  I/O NEW PSW WHEN AN I/O  * A2218060
*  INTERRUPTION  OCCURS.   THERE ARE TWO  INPUT PARAMETERS=  THE AD-  * A2218070
*  DRESS OF THE DEVICE IN THE I/O OLD PSW, AND THE CSW.               * A2218080
*                                                                     * A2218090
*  IF THE  INTERRUPTION  IS RELATED  TO SOME  PREVIOUS  I/O REQUEST,  * A2218100
*  DEVINT(J)  WILL CONTAIN THE ADDRESS OF  THE FIRST BYTE OF AN EXIT  * A2218110
*  SEQUENCE.   THIS EXIT SEQUENCE WILL HAVE THE FOLLOWING FORMAT=     * A2218120
*                                                                     * A2218130
*       TERM   BC      15,ADDR1     (ADDRESS  LINK)                  * A2218140
*      OTHER   BC      15,ADDR2     (ADDRESS  LINK+4)                * A2218150
```

```
*       EXCEPT BC      15,ADDR3     (ADDRESS  LINK+8)                 * A2218160
*       SENSE  BC      15,ADDR4     (ADDRESS  LINK+12)                * A2218170
*       CHEND  BC      15,ADDR5     (ADDRESS  LINK+16)                * A2218180
*                                                                     * A2218190
*  AFTER THE EXECUTION OF THIS ROUTINE, THE ABOVE EXIT SEQUENCE WILL  * A2218200
*  GIVE CONTROL  TO ONE OF THE  ADDRESSES  LINK, LINK+4,... LINK+16.  * A2218210
*                                                                     * A2218220
*      TERM       THE OPERATION  HAS TERMINATED  WITHOUT  UNIT CHECK, * A2218230
*                 UNIT EXCEPTION  OR OTHER ERROR CONDITIONS           * A2218240
*                                                                     * A2218250
*      EXCEPT     THE OPERATION HAS TERMINATED AND A  UNIT EXCEPTION, * A2218260
*                 CHAINING CHECK, PROGRAM CHECK  OR  PROTECTION CHECK * A2218270
*                 CONDITION HAS BEEN DETECTED                         * A2218280
*                                                                     * A2218290
*      SENSE      THE OPERATION HAS TERMINATED  AND A UNIT CHECK CON- * A2218300
*                 DITION HAS BEEN DETECTED                            * A2218310
*                                                                     * A2218320
*      CHEND      A CHANNEL END WITHOUT DEVICE END CONDITION HAS BEEN * A2218330
*                 DETECTED                                            * A2218340
*                                                                     * A2218350
*      OTHER      NONE OF THE ABOVE CONDITIONS HAS OCCURRED           * A2218360
*                                                                     * A2218370
*  IF THE INTERRUPTION IS NOT RELATED  TO SOME PREVIOUS  I/O REQUEST  * A2218380
*  (FOR EXAMPLE, A DEVICE END OR ATTENTION INTERRUPTION), THE INTER-  * A2218390
*  RUPTION WILL BE PROCESSED IN THE  IOINT  ROUTINE.                  * A2218400
*                                                                     * A2218410
********************************************************************** A2218420
         SPACE                                                        A2218430
IOINT    STM   1,7,IOGR              GENERAL REGISTERS 1-7 TO IOGR    A2218440
         BAL   LINK,INTUCB           GET INDEX PAIR J,K OF UCB        A2218450
         BC    15,UNSTEX+4           INTERRUPT FOR UNKNOWN CHAN  V1L1 A2218460
*        BC    15,UNSTEX+4           INTERRUPT FOR UNDE(DELETED) V1L1 A2218465
         BC    15,UNSTAK             INTERRUPT FOR UNKNOWN CUNIT V1L1 A2218470
*                                   IGNORE              (DELETED) V1L1 A2218475
         TM    CSW+5,CDICCC          IS ANY DC,ICC OR CCC PRESENT     A2218480
         BC    5,SREP0S              YES-SET UP SEREP INTERFACE       A2218490
         MVC   *+7(1),INVST(J)       *                                A2218500
         TM    CSW+4,X'00'           NO-ANY INVALID DEVICE STATUS BIT A2218510
         BC    5,SREP1I              YES-SET UP SEREP INTERFACE       A2218520
         TM    BORCH(J),SNSFLG       IS BORCH(J)=10 OR 11             A2218530
         BC    12,IOINTK             NO-GO TO TEST FOR A              A2218540
         L     LINK,DEVINT(J)        ADDRESS OF EXIT SEQ. TO (LINK)   A2218550
         L     I,DEVSVC(J)           ADDRESS OF ASSOC. SVC TO I       A2218560
         TM    CSW+4,DESTAT          ANY DEVICE STATUS PRESENT        A2218570
         BC    8,4(LINK)             NO-IGNORE PCI INTERRUPT          A2218580
         OC    SVCCSW(8,I),CSW       YES-CSW OR SVCCSW(I) TO SVCCSW   A2218590
         TM    CSW+5,PORPRC          ANY PC OR PRC PRESENT            A2218600
         BC    8,IOINTA              NO-GO TO TEST FOR UC             A2218610
         OI    ERRTYP(I),X'03'       YES-X'03' TO ERRTYP(I)           A2218620
IOINTA   TM    SVCCSW+4(I),UC        IS UNIT CHECK PRESENT            A2218630
         BC    8,IOINTB              NO-GO TO TEST FOR DE             A2218640
         TM    BORCH(J),BSYFLG       YES-IS BORCH(I)=11               A2218650
         BC    12,IOINTB             NO-BORCH(J)=10,GO TO TEST FOR DE A2218660
         MVI   BORCH(J),SNSFLG       NO-X'10' TO BORCH(J)             A2218670
         IC    WORK,SNSCNT           SNSCNT TO (WORK)                 A2218680
```

```
         LA    WORK,1(WORK)          (WORK)+1 TO (WORK)                A2218690
         STC   WORK,SNSCNT           SNSCNT+1 TO SNSCNT                A2218700
IOINTB   TM    CSW+4,DE              IS DE PRESENT                     A2218710
         BC    1,IOINTD              YES-GO TO TEST FOR UC             A2218720
         TM    CSW+4,CE              NO-IS CE PRESENT                  A2218730
         BC    8,IOINTC              NO-GO TO TEST FOR CUE             A2218740
         BC    1,16(LINK)            YES-RETURN TO (LINK)+16 -         A2218750
*                                     CHANNEL END                       A2218760
IOINTC   TM    CSW+4,CUE             IS CUE PRESENT                    A2218770
         BC    1,4(LINK)             YES-RETURN TO (LINK)+4 - OTHER    A2218780
*                                     CONDITION                         A2218790
IOINTD   TM    SVCCSW+4(I),UC        NO-IS UC PRESENT                  A2218800
         BC    8,IOINTE              NO-TERMINATED,GO TO TEST FOR UE   A2218810
         BAL   LINKA,STACK           YES-ADD J TO CHANNEL CHAIN        A2218820
         BC    15,12(LINK)           RETURN TO (LINK)+12,SENSE -       A2218830
*                                     AFTER STACK                       A2218840
IOINTE   NI    BORCH(J),X'00'        X'00' TO BORCH(J)                 A2218850
         TM    SVCCSW+4(I),UE        IS UE PRESENT                     A2218860
         BC    1,8(LINK)             YES-RETURN TO (LINK)+8 -          A2218870
*                                     EXCEPTIONAL CONDITION             A2218880
         TM    SVCCSW+5(I),CHC       IS CHAINING CHECK                 A2218890
         BC    1,8(LINK)             YES-RETURN TO (LINK)+8 -          A2218900
*                                     EXCEPTIONAL CONDITION             A2218910
         TM    ERRTYP(I),X'03'       NO-IS ERRTYP=X'00'                A2218920
         BC    5,8(LINK)             NO-RETURN TO (LINK)+8 -           A2218930
*                                     EXCEPTIONAL CONDITION             A2218940
         BCR   15,LINK               YES-RETURN TO (LINK)--TERMINATED  A2218950
         SPACE 3                                                       A2218960
IOINTK   TM    CSW+4,A               IS A=1                            A2218970
         BC    8,UNSTAK              NO-IGNORE DE,GO TO UNSTAK         A2218980
         TM    ATTSW,X'01'           YES-IS CONSOLE DISABLED           A2218990
         BC    1,UNSTAK              YES-IGNORE ATTN. INTERRUPTION     A2219000
         L     WORK,DEVATT(J)        NO-ATTPSW(J) TO (WORK)            A2219010
         LTR   WORK,WORK             IS ATTPSW(J)=0                    A2219020
         BC    8,UNSTAK              YES-IGNORE ATTENTION              A2219030
         MVC   0(8,WORK),OIOPSW      NO-OIOPSW TO (ATTPSW(J))          A2219040
         LA    WORK,8(WORK)          ATTRET(J)=ATTPSW(J)+8 TO (WORK)   A2219050
         ST    WORK,TEMP             ATTRET(J) TO OIOPSW(A)            A2219060
         MVC   OIOPSW+5(3),TEMP+1    *                                 A2219070
         NI    OIOPSW,X'00'          DISABLE I/O AND EXTERNAL          A2219080
         BC    15,UNSTAK             INTERRUPTIONS,GO TO UNSTAK        A2219090
         EJECT                                                         A2219100
*************************************************************************A2219110
*                                                                     * A2219120
*  I/O REQUEST CHAIN MANIPULATION                                     * A2219130
*                                                                     * A2219140
*  SUPPOSE THAT FOR SOME CHANNEL  THREE UCB'S  WITH INDICES  J1,  J2  * A2219150
*  AND  J3  ARE CHAINED (IN THAT ORDER) ON THE CHANNEL WITH INDEX K.  * A2219160
*                                                                     * A2219170
*      (IOQBEG(K))  =  J1  =  ADDRESS OF FIRST UCB ON THE CHAIN       * A2219180
*      (DEVCHN(J1)) =  J2  =  ADDRESS OF SECOND UCB ON THE CHAIN      * A2219190
*      (DEVCHN(J2)) =  J3  =  ADDRESS OF THIRD UCB ON THE CHAIN       * A2219200
*      (DEVCHN(J3)) =  0   =  NO FURTHER UCB'S ON THE CHAIN           * A2219210
*                                                                     * A2219220
*      (IOQEND(K)) = DEVCHN(J3)  TO INDICATE THAT THE ADDRESS OF THE  * A2219230
```

```
*                                NEXT  UCB  TO BE ADDED TO THE CHAIN  * A2219240
*                                IS TO BE PLACED IN THE WORD WITH AD- * A2219250
*                                DRESS  DEVCHN(J3)                    * A2219260
*                                                                     * A2219270
*     WHEN THERE ARE NO UCB'S ON THE CHAIN,  (IOQEND(K)) = IOQBEG(K) * A2219280
*                                                                     * A2219290
*  ADDING A UCB TO A CHAIN                                            * A2219300
*                                                                     * A2219310
*  USING THE ABOVE EXAMPLE, TO ADD A UCB WITH INDEX J4 TO THE CHAIN,  * A2219320
*  THE FOLLOWING STEPS ARE EXECUTED=                                  * A2219330
*                                                                     * A2219340
*     PICK UP DEVCHN(J3) FROM IOQEND(K)                               * A2219350
*     PLACE J4 AT ADDRESS DEVCHN(J3)                                  * A2219360
*     PLACE ADDRESS DEVCHN(J4) AT IOQEND(K)                           * A2219370
*                                                                     * A2219380
*  GOING THROUGH A REQUEST CHAIN                                      * A2219390
*                                                                     * A2219400
*  THE FOLLOWING STEPS  ARE PERFORMED WHEN GOING  THROUGH A CHAIN OF  * A2219410
*  I/O REQUESTS=                                                      * A2219420
*                                                                     * A2219430
*  1.  THE FIRST VALUE OF  J  IS PICKED UP FROM IOQBEG(K)             * A2219440
*  2.  DEVCHN(J) IS TESTED, AND IF IT CONTAINS ZERO, THE  CHAINING    * A2219450
*      IS TERMINATED                                                  * A2219460
*  3.  IF DEVCHN(J) CONTAINS A NON-ZERO VALUE, THE NEXT VALUE OF J    * A2219470
*      IS PICKED UP FROM DEVCHN(J) AND STEP 2 IS EXECUTED AGAIN       * A2219480
*                                                                     * A2219490
*  THIS CHAINING IS PERFORMED USING A GENERAL REGISTER  POINTR. THIS  * A2219500
*  REGISTER CONTAINS, SUCCESSIVELY, IOQBEG(K), DEVCHN(J1), DEVCHN(J2) * A2219510
*  ..., DEVCHN(JN).                                                   * A2219520
*                                                                     * A2219530
*  REMOVING A UCB FROM A CHAIN                                        * A2219540
*                                                                     * A2219550
*  TO REMOVE THE UCB WITH INDEX J3  (LAST UCB ON THE CHAIN) FROM THE  * A2219560
*  CHAIN DESCRIBED ABOVE, THE FOLLOWING STEPS ARE EXECUTED=           * A2219570
*                                                                     * A2219580
*     PICK UP ADDRESS DEVCHN(J2) FROM GENERAL REGISTER POINTR         * A2219590
*     PLACE ZERO IN ADDRESS DEVCHN(J2)                                * A2219600
*     PLACE ADDRESS DEVCHN(J2) AT IOQEND(K)                           * A2219610
*                                                          .../...    * A2219620
*                                                                     * A2219630
*********************************************************************** A2219640
        EJECT                                                          A2219650
*********************************************************************** A2219660
*                                                                     * A2219670
*  .../...                                                            * A2219680
*                                                                     * A2219690
*  TO REMOVE  THE UCB WITH INDEX J2  (NOT THE LAST UCB ON THE CHAIN)  * A2219700
*  FROM THE CHAIN DESCRIBED ABOVE, THE FOLLOWING STEPS ARE EXECUTED=  * A2219710
*                                                                     * A2219720
*     PICK UP ADDRESS DEVCHN(J1) FROM GENERAL REGISTER POINTR         * A2219730
*     PLACE (DEVCHN(J2))=J3 AT ADDRESS DEVCHN(J1)                     * A2219740
*     PLACE ZERO IN ADDRESS DEVCHN(J2)                                * A2219750
*                                                                     * A2219760
*********************************************************************** A2219770
        SPACE 3                                                        A2219780
```

```
*********************************************************************** A2219790
*                                                                     * A2219800
*      ROUTINE TO CHAIN AN I/O REQUEST AND CONTINUE                   * A2219810
*                                                                     * A2219820
*********************************************************************** A2219830
         SPACE                                                        A2219840
STACK    OI    BORCH(J),CHNFLG        X'01' TO BORCH(J)               A2219850
         L     POINTR,IOQEND(K)       (IOQEND(K)) TO POINTR           A2219860
         ST    J,0(POINTR)            J TO (POINTR)                   A2219870
         LA    WORKA,DEVCHN(J)        DEVCHN(J) TO (IOQEND(K))        A2219880
         ST    WORKA,IOQEND(K)        *                               A2219890
         BCR   15,LINKA               RETURN TO CALLING ROUTINE       A2219900
         SPACE 2                                                      A2219910
*********************************************************************** A2219920
*                                                                     * A2219930
*      UNSTAK ROUTINE                                                 * A2219940
*                                                                     * A2219950
*  THIS ROUTINE  ATTEMPTS TO  INITIATE AS  MANY I/O OPERATIONS  ON A  * A2219960
*  DESIGNATED  CHANNEL CHAIN  AS POSSIBLE.   THE ROUTINE  IS ENTERED  * A2219970
*  WITH ONE INPUT PARAMETER, THE CHANNEL INDEX  K.                    * A2219980
*  ANY UCB FOR WHICH AN  I/O  OPERATION IS STARTED  (OR IS INHIBITED  * A2219990
*  DUE TO EXCEPTIONAL CONDITIONS)  WILL BE REMOVED  FROM THE CHANNEL  * A2220000
*  CHAIN.                                                             * A2220010
*                                                                     * A2220020
*********************************************************************** A2220030
         SPACE 2                                                      A2220040
UNSTAK   LA    POINTR,IOQBEG(K)       IOQBEG(K) ADDRESS TO (POINTR)   A2220050
         MVI   UNSTSW,ZERO            *                               A2220060
UNSTKA   L     J,0(POINTR)            (POINTR) TO J                   A2220070
         LTR   J,J                    *                               A2220080
         BC    8,UNSTEX               RETURN TO CALLER                A2220090
         L     I,DEVSVC(J)            (DEVSVC(J)) TO I                A2220100
         TM    BORCH(J),SNSFLG        IS SENSE FLAG IN BORCH(J)       A2220110
         BC    1,UNKSNS               YES-GO TO TRY SENSE             A2220120
         NI    BORCH(J),FREFLG        NO-X'00' TO BORCH(J)            A2220130
         ST    POINTR,POINTA          SAVE POINTR                     A2220140
         BAL   LINK,STRTIO            TRY TO START I/O OPERATION      A2220150
         BC    15,UNSTKD              OPERATION TERMINATED            A2220160
         BC    15,*                   DEVICE BUSY-NO INTERRUPTION     A2220170
         BC    15,UNSTKC              PATH BUSY                       A2220180
         BC    15,UNSTKG              ERROR-EXCEPTIONAL CONDITION,    A2220190
*                                      SENSE OPERATION                 A2220200
         BC    15,UNSTKH              CHANNEL END                     A2220210
         L     WORK,DEVCHN(J)         START-(DEVCHN(J)) TO POINTR     A2220220
         L     POINTR,POINTA          RESTORE POINTR                  A2220230
         ST    WORK,0(POINTR)         *                               A2220240
         XC    DEVCHN(4,J),DEVCHN(J)  0 TO DEVCHN(J)                  A2220250
         LTR   WORK,WORK              WAS DEVCHN(J)=0                 A2220260
         BC    7,UNSTEX               YES-RETURN TO CALLER            A2220270
UNSTKB   ST    POINTR,IOQEND(K)       NO-(POINTR) TO IOQEND           A2220280
         BC    15,UNSTEX              RETURN TO CALLER                A2220290
UNSTKC   OI    BORCH(J),CHNFLG        X'01' TO BORCH(J)               A2220300
         LA    POINTR,DEVCHN(J)       DEVCHN(J) TO POINTR             A2220310
         BC    15,UNSTKA              EXAMINE NEXT ELEMENT IN CHAIN   A2220320
UNSTKD   MVC   SVCPSW(8,I),OIOPSW     OIOPSW TO SVCPSW(I)             A2220330
```

```
         MVC   OIOPSW+5(3),29(I)       NRMRET TO OIOPSW(A)                A2220340
UNSTKE   NI    OIOPSW,X'00'            DISABLE I/O AND EXTERNAL INT.      A2220350
         NI    SVCPSW+1(I),X'FD'       0 TO WAIT STATE BIT IN SVCPSW      A2220360
UNSTKF   L     WORK,DEVCHN(J)          (DEVCHN(J)) TO (POINTR)            A2220370
         L     POINTR,POINTA           RESTORE POINTR                     A2220380
         ST    WORK,0(POINTR)          *                                  A2220390
         XC    DEVCHN(4,J),DEVCHN(J)   0 TO (DEVCHN(J))                   A2220400
         LTR   WORK,WORK               WAS (DEVCHN(J))=0                  A2220410
         BC    8,UNSTKB                YES-TERMINATED,RETURN TO CALLER    A2220420
         BC    15,UNSTKA               NO-TRY NEXT ELEMENT IN CHAIN       A2220430
UNSTKG   MVC   SVCPSW(8,I),OIOPSW      OIOPSW TO SVCPSW(I)                A2220440
         MVC   OIOPSW+5(3),33(I)       EXCRET TO OIOPSW(A)                A2220450
         BC    15,UNSTKE               GO TO DISABLE INTERRUPTIONS        A2220460
UNSTKH   CLI   1(I),X'02'              IS THIS AN 'SVC 2' CALL            A2220470
         BC    8,UNSTKF                YES-REMOVE UCB FROM CHAIN          A2220480
         TM    CSW+4,UCORUE            NO-IS ANY UC OR UE PRESENT         A2220490
         BC    5,UNSTKF                YES-REMOVE UCB FROM CHAIN          A2220500
         CLI   1(I),X'0D'              NO- IS THIS AN 'SVC 13/14' CALL    A2220510
         BC    4,*+8                   NO- 'SVC 1' CALL- CONTINUE         A2220520
         MVI   BORCH(J),FREFLG         YES-BORCH(J)=0 - IGNORE DE         A2220530
         BC    15,UNSTKD               NO-EXIT TO TERMINATE               A2220540
UNKSNS   ST    POINTR,POINTA           SAVE POINTR                        A2220550
         BAL   LINKA,SENSE             TRY TO PICK UP SENSE BYTES         A2220560
         BC    15,UNSTKC+4             PATH BUSY                          A2220570
         NI    BORCH(J),X'00'          SENSE OPERATION TERMINATED         A2220580
         L     WORK,SENSW              SENSE COUNT TO (WORK)              A2220590
         BCTR  WORK,0                  *                                  A2220600
         STC   WORK,SNSCNT             SNSCNT-1 IN SNSCNT                 A2220610
         MVI   UNSTSW,ONE              *                                  A2220620
         CLI   1(I),X'0B'              IS THIS AN 'SVC 11' CALL           A2220630
         BC    7,UNSTKG                NO-REMOVE UCB FROM CHAIN           A2220640
         BC    15,UNSTKF               YES-REMOVE UCB FROM CHAIN          A2220650
UNSTEX   NI    OIOPSW+1,X'FD'          0 TO WAIT STATE BIT IN OIOPSW      A2220660
         CLI   UNSTSW,ONE              WAS THE LAST CHAINED OPERATION     A2220670
         BC    8,UNSTAK                * A SENSE- YES,GO TO UNSTAK. NO,   A2220680
         LM    1,7,IOGR                EXIT SEQUENCE, RESTORE GENERAL     A2220690
*                                       REGISTERS 1-7                     A2220700
         LPSW  OIOPSW                  RETURN-OIOPSW IN PSW               A2220710
         EJECT                                                            A2220720
*********************************************************************** A2220730
*                                                                     * A2220740
*      SENSE OPERATION ROUTINE                                        * A2220750
*                                                                     * A2220760
*  THIS ROUTINE TRIES TO EXECUTE A SENSE OPERATION ON A DEVICE.   IT  * A2220770
*  IS ENTERED BY THE CALLING SEQUENCE=                                * A2220780
*                                                                     * A2220790
*              BAL     LINKA,SENSE                                    * A2220800
*                                                                     * A2220810
*  THE INPUT PARAMETERS ARE THE INDEX PAIR  (I,J).                    * A2220820
*                                                                     * A2220830
*  IF THE SENSE OPERATION CANNOT BE STARTED  (CHANNEL, SUBCHANNEL OR  * A2220840
*  CONTROL UNIT BUSY), CONTROL IS RETURNED TO ADDRESS  LINKA.         * A2220850
*                                                                     * A2220860
*  IF THE OPERATION CAN BE STARTED,  THE ROUTINE WAITS UNTIL THE OP-  * A2220870
*  ERATION IS FINISHED, AND THEN RETURNS CONTROL TO ADDRESS LINKA+4.  * A2220880
```

```
*                                                                     * A2220890
*  CERTAIN ERRORS DETECTED DURING THE EXECUTION OF THIS ROUTINE WILL  * A2220900
*  CAUSE THE SEREP INTERFACE TO BE SET UP.                            * A2220910
*                                                                     * A2220920
*  A MAXIMUM OF SIX SENSE BYTES WILL BE READ.  THE FIRST THREE BYTES  * A2220930
*  WILL BE STORED STARTING AT ADDRESS  SNSADD(I)  AND THE LAST THREE  * A2220940
*  BYTES  STARTING AT  ADDRESS  UCBSNS(J).   IF LESS  THAN SIX SENSE  * A2220950
*  BYTES ARE AVAILABLE FROM THE DEVICE,  THE REMAINING BYTES WILL BE  * A2220960
*  SET TO ZERO.                                                       * A2220970
*                                                                     * A2220980
********************************************************************** A2220990
         SPACE                                                        A2221000
SENSE    LA    WORKA,SNSCOM          SET ADDRESS OF SENSE COMMAND     A2221010
         ST    WORKA,CAW             IN (CAW)                         A2221020
         XC    SNSBYT(6),SNSBYT      CLEAR SENSE BYTES BUFFER         A2221030
         LH    DEVICE,DEV360(J)      SYSTEM/360 ADDRESS TO (DEVICE)   A2221040
SENSIO   SIO   0(DEVICE)             START I/O                        A2221050
         BC    8,SENTIO              CC=0 - WAIT UNTIL SENSE FINISHED A2221060
         BC    2,SENSIO              CC=2 - BUSY,TRY AGAIN UNTIL FREE A2221070
         BC    1,SREP3S              CC=3 - NOT OPERATIONAL,SET UP    A2221080
*                                      SEREP INTERFACE                A2221090
         TM    CSW+5,CHSTAT          CC=1 - ANY CHANNEL STATUS BITS   A2221100
         BC    5,SREP0S              YES-ERROR,SET UP SEREP INTERFACE A2221110
         TM    CSW+4,UC              NO-IS UNIT CHECK PRESENT         A2221120
         BC    5,SREP1S              YES-DEVICE FAILURE,SET UP SEREP  A2221130
*                                      INTERFACE                      A2221140
         TM    CSW+4,AORDE           ATTENTION OR DE PRESENT          A2221150
         BC    5,SENSIO              YES-IGNORE                       A2221160
         BCR   15,LINKA              NO-CU BUSY,RETURN TO (LINKA)     A2221170
SENTIO   TIO   0(DEVICE)             TEST I/O                         A2221180
         BC    2,SENTIO              CC=2 - BUSY,WAIT                 A2221190
         BC    1,SREP3S              CC=3 - NOT OPERATIONAL,SET UP    A2221200
*                                      SEREP INTERFACE                A2221210
         TM    CSW+5,CHNRST          CC=1 - ANY CHANNEL STATUS BITS,  A2221220
*                                      EXCEPT IL                      A2221230
         BC    5,SREP0S              YES-CHANNEL FAILURE,SET UP SEREP A2221240
*                                      INTERFACE                      A2221250
         TM    CSW+4,UC              NO-IS UNIT CHECK PRESENT         A2221260
         BC    5,SREP1S              YES-DEV.FAILURE-SET UP SEREP     A2221270
         TM    CSW+4,CE              NO-IS CHANNEL END PRESENT        A2221280
         BC    8,SENTIO              NO-WAIT                          A2221290
         MVC   SNSADD(3,I),SNSBYT    YES-FIRST 3 BYTES OF SENSE TO    A2221300
*                                      SNSADD                         A2221310
         MVC   UCBSNS(3,J),SNSBYT+3  LAST 3 BYTES OF SENSE TO UCB     A2221320
         BC    15,4(LINKA)           RETURN TO (LINKA)+4              A2221330
         EJECT                                                        A2221340
********************************************************************** A2221350
*                                                                     * A2221360
*      SEREP INTERFACE FOR I/O FAILURES                               * A2221370
*                                                                     * A2221380
*  THIS ROUTINE SETS UP IN SYSTEM/360 MAIN STORAGE THE ELEMENTS WHICH * A2221390
*  CONSTITUTE  THE STANDARD  SEREP INTERFACE.   IT CAUSES  A PSW, IN  * A2221400
*  WHICH I/O AND EXTERNAL  INTERRUPTIONS ARE DISABLED,  IN WHICH THE  * A2221410
*  WAIT STATE BIT IS ONE AND FOR WHICH THE LAST BYTE (ADDRESS FIELD)  * A2221420
*  IS SET TO ONE OF THE FOLLOWING VALUES, IS LOADED.                  * A2221430
```

```
*                                                                     * A2221440
*      X'FF'       MACHINE AND I/O FAILURES                         * A2221450
*      X'33'       DEVICE NOT OPERATIONAL                           * A2221460
*                                                                     * A2221470
*  WHEN SEREP IS CALLED  FROM OUTSIDE THE CONTROL PROGRAM,  THE CSW,  * A2221480
*  THE CAW AND THE INTERRUPTION CODE BITS IN THE OIOPSW ARE RESTORED. * A2221490
*  FURTHERMORE,  IN THE CASE OF AN I/O DEVICE FAILURE, THE APPROPRI-  * A2221500
*  ATE SENSE BYTES ARE TRANSFERRED TO LOCATIONS 24-29.                * A2221510
*                                                                     * A2221520
*  WHEN SEREP IS CALLED FROM WITHIN THE CONTROL PROGRAM AFTER EITHER  * A2221530
*  AN SIO OR TIO INSTRUCTION,  THE SYSTEM/360  DEVICE ADDRESS IS SET  * A2221540
*  UP IN  THE OIOPSW.   IN THE CASE OF AN  I/O DEVICE  FAILURE,  THE  * A2221550
*  BYTES AT LOCATIONS 24-29 ARE SET TO ZERO.                          * A2221560
*                                                                     * A2221570
*  IN ALL CASES,  THE BYTE AT LOCATION 115 WILL BE SET TO ONE OF THE  * A2221580
*  FOLLOWING VALUES.                                                  * A2221590
*                                                                     * A2221600
*      X'0F'       I/O CHANNEL FAILURE                              * A2221610
*      X'1F'       I/O DEVICE FAILURE                               * A2221620
*      X'3F'       DEVICE NOT OPERATIONAL                           * A2221630
*                                                                     * A2221640
********************************************************************* A2221650
         SPACE 3                                                      A2221660
********************************************************************* A2221670
*      ENTER HERE FROM SVCINT ROUTINE                                 * A2221680
*              SEREP CALLED FROM OUTSIDE CONTROL PROGRAM              * A2221690
********************************************************************* A2221700
         SPACE                                                        A2221710
SEREP    LR    WORK,I              CALLING SEQUENCE ADDRESS IN WORK  A2221720
         L     I,2(I)              RESTORE I                         A2221730
         TM    2(WORK),X'3F'       IS DEVICE OPERATIONAL             A2221740
         BC    1,SREP3S            NO-GO TO SREP3S                   A2221750
         MVC   CSW(8),SVCCSW(I)    YES-RESTORE CSW                   A2221760
         MVC   CAW(4),CAWADD(I)    AND CAW                           A2221770
         TM    2(WORK),X'1F'       IS I/O DEVICE FAILURE             A2221780
         BC    12,SREP0S           NO-I/O CHANNEL FAILURE            A2221790
         MVC   24(3),SNSADD(I)     YES-SENSE BYTES 0-2 TO BYT 24-26  A2221800
         BAL   LINK,GETUCB         GET INDEX PAIR (J,K) OF UCB       A2221810
         BC    15,*                NOT FOUND                         A2221820
         BC    15,*                NOT FOUND                  V1L1   A2221825
         MVC   27(3),UCBSNS(J)     SENSE BYTES 3-5  TO BYTES 27-29   A2221830
         BC    15,SREP1I           TREAT DEVICE FAILURE              A2221840
SREP1S   XC    24(6),24            DEVICE FAILURE-SIO/TIO,0 TO       A2221850
*                                    BYTES 24-26                     A2221860
SREP1I   MVI   115,X'1F'           DEVICE FAILURE-I/O INTERRUPTION   A2221870
         XC    OIOPSW(8),OIOPSW    CLEAR OLD I/O PSW                 A2221880
         BC    15,WAIT             X'1F' TO BYTE 50,WAIT             A2221890
SREP0S   MVI   115,X'0F'           CHANNEL FAILURE-SIO/TIO           A2221900
         BC    15,WAIT             INTERRUPTION,WAIT                 A2221910
SREP3S   MVI   115,X'3F'           CC=3 - NON-OPERATIONAL DEVICE     A2221920
         NI    119,X'33'           FOUND BY SENSE ROUTINE            A2221930
WAIT     MVC   58(2),2(I)          DEV360(I) TO BYTES 58-59          A2221940
         LPSW  NMCPSW              MACHINE-CHECK NEW PSW TO PSW      A2221950
         EJECT                                                       A2221960
********************************************************************* A2221970
```

```
*                                                                     * A2221980
*  CONSOLE COMMUNICATION                                              * A2221990
*                                                                     * A2222000
*  TWO TYPES OF CONSOLE COMMUNICATION WILL BE HANDLED BY THE CONTROL  * A2222010
*  PROGRAM.   THE FIRST  TYPE WILL ALLOW  A MESSAGE TO  BE SENT FROM  * A2222020
*  THE SIMULATOR TO  THE CONSOLE PRINTER,  AND THE SECOND  TYPE WILL  * A2222030
*  ALLOW TRANSMISSION OF A COMMAND FROM THE CONSOLE KEY BOARD TO THE  * A2222040
*  SIMULATOR IN RESPONSE TO AN ATTENTION INTERRUPTION FROM THE OPER-  * A2222050
*  ATOR.   THERE ARE NO FACILITIES FOR PROCESSING QUEUES OF MESSAGES  * A2222060
*  OR COMMANDS.                                                       * A2222070
*                                                                     * A2222080
*                                  *****                              * A2222090
*                                                                     * A2222100
*  WRITE MESSAGE ROUTINE                                              * A2222110
*                                                                     * A2222120
*  A REQUEST TO WRITE A MESSAGE MAY BE SUBMITTED BY THE SIMULATOR TO  * A2222130
*  THE CONTROL PROGRAM USING AN SVC CALLING SEQUENCE OF THE FORM=     * A2222140
*                                                                     * A2222150
*              CNOP    2,4                                            * A2222160
*       I      SVC     4                                              * A2222170
*       N      DC      X'NN'                                          * A2222180
*              DC      AL3(BUFF)                                      * A2222190
*       I+6    ANY INSTRUCTION                                        * A2222200
*                                                                     * A2222210
*  THE BYTES  TO BE PRINTED  WILL BE TAKEN  FROM LOCATIONS   BUFF+1,  * A2222220
*  BUFF+2,...,BUFF+X'NN'.   THE CONTROL  PROGRAM WILL  SEND THE CON-  * A2222230
*  TENTS OF THESE BYTES TO THE CONSOLE PRINTER USING A WRITE INHIBIT  * A2222240
*  CARRIAGE  RETURN  COMMAND.   CONSEQUENTLY,  IF A NEW  LINE IS RE-  * A2222250
*  QUIRED AT THE END OF THE MESSAGE, THE 'NEW LINE' CHARACTER SHOULD  * A2222260
*  BE SET UP IN LOCATION  BUFF+X'NN'.                                 * A2222270
*                                                                     * A2222280
*  IF THE CONTROL PROGRAM IS BUSY WITH SOME UNFINISHED READ OR WRITE  * A2222290
*  REQUEST FOR THE PRINTER-KEYBOARD WHEN A WRITE MESSAGE CALLING SE-  * A2222300
*  QUENCE IS SUBMITTED,  THE CONTROL PROGRAM  WILL CYCLE ON THE CAL-  * A2222310
*  LING SEQUENCE  UNTIL THE PREVIOUS REQUEST  HAS TERMINATED.   WHEN  * A2222320
*  THE CONTROL PROGRAM ACCEPTS THE CALLING SEQUENCE, IT WILL SET THE  * A2222330
*  CONTENTS OF THE BYTE AT ADDRESS BUFF TO X'00',  WILL INITIATE THE  * A2222340
*  WRITING OF THE MESSAGE  AND WILL RETURN CONTROL TO ADDRESS I+6.    * A2222350
*                                                                     * A2222360
*                                                          .../...    * A2222370
*                                                                     * A2222380
********************************************************************* A2222390
        EJECT                                                          A2222400
********************************************************************* A2222410
*                                                                     * A2222420
*  .../...                                                            * A2222430
*                                                                     * A2222440
*  WHEN THE REQUEST IS TERMINATED,  THE CONTROL PROGRAM WILL SET THE  * A2222450
*  BYTE AT ADDRESS BUFF TO SOME NON-ZERO VALUE.   THUS=               * A2222460
*                                                                     * A2222470
*     -- A PROGRAMMING ERROR  HAS BEEM DETECTED.   THIS PROBABLY IN-  * A2222480
*        DICATES THAT PART OF THE CONTROL PROGRAM HAS BEEN OVERWRIT-  * A2222490
*        TEN.                                                         * A2222500
*             (BUFF)=X'03'                                            * A2222510
*                                                                     * A2222520
```

```
*     -- A DEVICE ERROR HAS BEEN DETECTED DURING THE PRINTING OF THE  * A2222530
*        MESSAGE.   THE CONTROL PROGRAM WILL REPEAT THE MESSAGE.      * A2222540
*        IF A SECOND ERROR OCCURS DURING THE PRINTING OF THE MESSAGE, * A2222550
*        A CONTROL ALARM WILL BE ISSUED.                              * A2222560
*              (BUFF)=X'01'                                           * A2222570
*        IF NO SECOND ERROR OCCURS,                                   * A2222580
*              (BUFF)=X'07'                                           * A2222590
*                                                                     * A2222600
*     -- A DEVICE ERROR  HAS PREVENTED THE  PRINTING OF THE MESSAGE.  * A2222610
*        THE CONTROL PROGRAM WILL TRY TO REPEAT THE OPERATION.        * A2222620
*        IF THE FAILURE OCCURS AGAIN, A CONTROL ALARM WILL BE ISSUED  * A2222630
*        AND THE SEREP INTERFACE WILL BE SET UP.                      * A2222640
*        IF THE FAILURE DOES NOT OCCUR AGAIN,                         * A2222650
*              (BUFF)=X'07'                                           * A2222660
*                                                                     * A2222670
*     -- THE MESSAGE WAS WRITTEN WITHOUT ERROR.                       * A2222680
*              (BUFF)=X'07'                                           * A2222690
*                                                                     * A2222700
*  WHEN THE SIMULATOR IS IN THE DISABLED STATE,  A WRITE MESSAGE RE-  * A2222710
*  QUEST MAY  NOT BE SUBMITTED  UNLESS THE  DISABLED STATE  HAS BEEN  * A2222720
*  CAUSED BY AN  INTERRUPTION RESULTING  FROM AN OPERATOR COMMAND AT  * A2222730
*  THE CONSOLE KEYBOARD.                                              * A2222740
*                                                                     * A2222750
*                                                            .../...  * A2222760
*                                                                     * A2222770
********************************************************************** A2222780
         EJECT                                                         A2222790
********************************************************************** A2222800
*                                                                     * A2222810
*  .../...                                                            * A2222820
*                                                                     * A2222830
*                                  *****                              * A2222840
*                                                                     * A2222850
*  COMMAND INPUT ROUTINE                                              * A2222860
*                                                                     * A2222870
*  WHEN THE ATTENTION KEY ON THE CONSOLE KEYBOARD IS DEPRESSED,  THE  * A2222880
*  SIMULATOR WILL BE INTERRUPTED AND THE CONTROL PROGRAM WILL BE EN-  * A2222890
*  TERED.   IN RESPONSE TO  THIS INTERRUPTION,  THE CONTROL  PROGRAM  * A2222900
*  WILL  SET UP AND  EXECUTE A  READ COMMAND.   INFORMATION  WILL BE  * A2222910
*  READ FROM THE CONSOLE KEYBOARD INTO A COMMAND BUFFER.   WHEN  THE  * A2222920
*  READING OPERATION IS TERMINATED,  CONTROL WILL BE RETURNED TO THE  * A2222930
*  SIMULATOR AT A PRE-DETERMINED ADDRESS.                             * A2222940
*                                                                     * A2222950
*  BEFORE ANY INFORMATION  CAN BE TRANSMITTED  FROM THE CONSOLE KEY-  * A2222960
*  BOARD TO THE SIMULATOR,  AN SVC CALLING SEQUENCE OF THE FOLLOWING  * A2222970
*  FORM MUST BE SUBMITTED.                                            * A2222980
*                                                                     * A2222990
*              CNOP    6,8                                            * A2223000
*       I      SVC     5                                              * A2223010
*       N      DC      X'NN'                                          * A2223020
*              DC      AL3(BUFF)                                      * A2223030
*       COMLEN DC      X'00'                                          * A2223040
*              DC      AL3(COMRET)                                    * A2223050
*       COMPSW DS      D                                              * A2223060
*       I+18   ANY INSTRUCTION                                        * A2223070
```

```
*                                                                     * A2223080
*  THIS CALLING  SEQUENCE NEED BE  PRESENTED TO THE  CONTROL PROGRAM  * A2223090
*  ONLY ONCE, AND THE PARAMETERS WHICH IT CONTAINS WILL BE USED,  AS  * A2223100
*  DESCRIBED BELOW, IN CONJUNCTION WITH ALL COMMANDS FROM THE OPERA-  * A2223110
*  TOR.   (ANY ATTENTION INTERRUPTIONS  WHICH OCCUR PRIOR TO SUBMIT-  * A2223120
*  TING THIS CALLING SEQUENCE WILL BE IGNORED.)                       * A2223130
*                                                                     * A2223140
*  THE BYTE AT ADDRESS  COMLEN WILL CONTAIN THE NUMBER OF CHARACTERS  * A2223150
*  READ.                                                              * A2223160
*                                                                     * A2223170
*  X'NN' DENOTES THE MAXIMUM NUMBER OF CHARACTERS WHICH MAY BE READ.  * A2223180
*  HENCE, THE NUMBER OF CHARACTERS, (COMLEN), MAY NEVER EXCEED X'NN'. * A2223190
*                                                                     * A2223200
*  THE CHARACTERS OF ANY COMMAND WILL BE PLACED IN LOCATIONS BUFF+1,  * A2223210
*  BUFF+2,...,BUFF+(COMLEN).                                          * A2223220
*                                                                     * A2223230
*                                                          .../...    * A2223240
*                                                                     * A2223250
*********************************************************************** A2223260
        EJECT                                                          A2223270
*********************************************************************** A2223280
*                                                                     * A2223290
*  .../...                                                            * A2223300
*                                                                     * A2223310
*  THE FOLLOWING  TERMINATION CONDITIONS  MAY BE ASSOCIATED WITH THE  * A2223320
*  READING OF A COMMAND=                                              * A2223330
*                                                                     * A2223340
*     -- A DEVICE ERROR HAS BEEN DETECTED DURING THE READING OF  THE  * A2223350
*        COMMAND.   THE CONTROL PROGRAM  WILL ISSUE AN ERROR MESSAGE  * A2223360
*        FOR THE OPERATOR  AND WILL RETURN  CONTROL TO THE  POINT OF  * A2223370
*        INTERRUPTION.   THUS, THE COMMAND IS IGNORED.                * A2223380
*                                                                     * A2223390
*     -- A CONTROL ALARM IS ISSUED AND THE SEREP INTERFACE IS SET UP. * A2223400
*        THIS MAY BE THE RESULT OF ONE OF THE FOLLOWING CONDITIONS=   * A2223410
*           - A DEVICE ERROR HAS PREVENTED  THE READING OF THE COM-   * A2223420
*             MAND.   THE CONTROL PROGRAM HAS RETRIED THE OPERATION   * A2223430
*             AND THE FAILURE HAS OCCURRED AGAIN.                     * A2223440
*           - THE ERROR MESSAGE  TO THE OPERATOR  IN THE CASE  OF A   * A2223450
*             DEVICE ERROR  DURING THE EXECUTION  OF A READ COMMAND   * A2223460
*             CANNOT BE WRITTEN.                                      * A2223470
*                                                                     * A2223480
*     -- A PROGRAMMING ERROR HAS OCCURRED.   THIS PROBABLY INDICATES  * A2223490
*        THAT PART OF THE CONTROL PROGRAM HAS BEEN OVERWRITTEN.       * A2223500
*              (BUFF)=X'03'                                           * A2223510
*                                                                     * A2223520
*     -- THE COMMAND HAS BEEN READ WITHOUT ERROR.                     * A2223530
*              (BUFF)=X'07'                                           * A2223540
*                                                                     * A2223550
*  FOR THE LAST TWO OF THESE TERMINATION CONDITIONS, CONTROL WILL BE  * A2223560
*  RETURNED TO THE SIMULATOR AT LOCATION COMRET WITH ALL I/O AND EX-   * A2223570
*  TERNAL INTERRUPTIONS  DISABLED.   THE PSW OF THE SIMULATOR AT THE  * A2223580
*  POINT OF INTERRUPTION WILL BE PLACED IN LOCATION COMPSW.           * A2223590
*                                                                     * A2223600
*  IN ORDER TO AVOID THE POSSIBILITY OF  OVERWRITING THE INFORMATION  * A2223610
*  IN THE COMMAND BUFFER BY  A SUBSEQUENT COMMAND FROM THE OPERATOR,  * A2223620
```

```
*  THE SEQUENCE STARTING AT LOCATION  COMRET  SHOULD HAVE COMPLETELY  * A2223630
*  PROCESSED THIS  INFORMATION BEFORE  RETURNING TO THE POINT OF IN-  * A2223640
*  TERRUPTION.                                                        * A2223650
*                                                                     * A2223660
*  THE CANCEL CONDITION AT THE CONSOLE KEYBOARD WILL BE TREATED NOR-  * A2223670
*  MALLY,  I.E.,  A NEW REQUEST  TO READ  FROM THE  CONSOLE KEYBOARD  * A2223680
*  WILL BE ISSUED.                                                    * A2223690
*                                                                     * A2223700
********************************************************************** A2223710
         EJECT                                                         A2223720
********************************************************************** A2223730
*                                                                     * A2223740
*      ROUTINE TO TRANSMIT MESSAGES TO THE CONSOLE TYPEWRITER         * A2223750
*                                                                     * A2223760
*  THIS ROUTINE IS ENTERED VIA THE  SVC TABLE WHENEVER AN SVC 4 CAL-  * A2223770
*  LING SEQUENCE IS ENCOUNTERED.   IT IS ENTERED WITH ONE PARAMETER,  * A2223780
*  THE SVC INDEX  I.                                                  * A2223790
*                                                                     * A2223800
********************************************************************** A2223810
         SPACE                                                         A2223820
MESAGE   MVC   MESSGR(28),SVCGR       SVCGR TO MESSGR                 A2223830
         MVC   MESPSW(8),OSVPSW       OSVPSW TO MESPSW                A2223840
         ST    I,TEMP                 SAVE  I                         A2223850
         LA    I,CONSOL-2             GET INDEX PAIR (J,K) OF ...     A2223860
         BAL   LINK,GETUCB            ... CONSOLE UCB.                A2223870
         BC    0,0                    UCB NOT FOUND, IGNORE.          A2223880
         BC    0,0                    UCB NOT FOUND            V1L1   A2223885
         L     I,TEMP                 RESTORE  I                      A2223890
         TM    BORCH(J),BSYFLG        IS BORCH(J) = 01,10,11          A2223900
         BC    8,MESGEA               YES-START TO WRITE MESSAGE      A2223910
         ST    I,MESPSW+4             NO-CYCLE ON SVC,I TO MESPSW(A)  A2223920
         BC    15,MESRET              RETURN TO CALLER                A2223930
MESGEA   MVI   OPTYPE,X'01'           OPTYPE=0,CONSBY=01              A2223940
         LA    WORK,MESCOM            WRITE COMMAND ADDRESS TO (WORK) A2223950
         ST    WORK,CONSOL+2          WRITE COMMAND ADDRESS TO CAWADD A2223960
         L     BUFF,2(I)              ((I)+2) TO BUFF                 A2223970
         ST    BUFF,ADBUFF            BUFF TO ADBUFF                  A2223980
         MVI   0(BUFF),X'00'          X'00' TO (BUFF)                 A2223990
         LA    WORK,1(BUFF)           SEQUENCE TO FORM I/O COMMAND     A2224000
         ST    WORK,MESCOM            CCW=X'01',BUFF+1,0,N            A2224010
         OI    MESCOM,X'01'           *                               A2224020
         MVC   MESCOM+7(1),2(I)       *                               A2224030
         AH    I,DEC6                 I+6 TO MESPSW(A)                A2224040
         ST    I,MESPSW+4             *                               A2224050
         EJECT                                                         A2224060
********************************************************************** A2224070
*                                                                     * A2224080
*      EXECUTION ROUTINE FOR CONSOLE OPERATIONS                       * A2224090
*                                                                     * A2224100
*  THIS ROUTINE SUBMITS I/O REQUESTS  FOR THE PROCESSING OF MESSAGES  * A2224110
*  AND OF COMMANDS TO THE CONTROL PROGRAM  AND PROCESSES THE RESULT-  * A2224120
*  ING INTERRUPTIONS.                                                 * A2224130
*                                                                     * A2224140
*  WHEN THIS ROUTINE IS ENTERED, THE CALLING ROUTINE HAS SET PARAME-  * A2224150
*  TERS TO INDICATE THE TYPE OF OPERATION TO BE EXECUTED.  THERE ARE  * A2224160
```

```
*  FOUR TYPES OF OPERATIONS, INDICATED BY THE FOLLOWING PARAMETERS=   * A2224170
*                                                                     * A2224180
*     -- ALRMSW = RDERSW = 0       WRITE MESSAGE                     * A2224190
*        OPTYPE = 0                                                   * A2224200
*                                                                     * A2224210
*     -- ALRMSW = RDERSW = 0       READ COMMAND                      * A2224220
*        OPTYPE = 1                                                   * A2224230
*                                                                     * A2224240
*     -- ALRMSW = 1                CONTROL ALARM                     * A2224250
*        RDERSW = 0                                                   * A2224260
*        OPTYPE = 0,1                                                 * A2224270
*                                                                     * A2224280
*     -- ALRMSW = 0                WRITE SPECIAL  MESSAGE  TO        * A2224290
*        RDERSW = 1                  INDICATE EQUIPMENT CHECK       * A2224300
*        OPTYPE = 1                  ENCOUNTERED DURING  PRE-       * A2224310
*                                    VIOUS READ COMMAND             * A2224320
*                                                                     * A2224330
*********************************************************************** A2224340
         SPACE                                                        A2224350
MESGEB   OI    CPSW,X'80'            CPSW=1                          A2224360
         CNOP  4,8                   SUBMIT 'I/O AND CONTINUE' REQ.  A2224370
         SVC   2                     *                               A2224380
CONSOL   DC    X'0009'               CONSOLE ADDRESS                 A2224390
         DS    F                     CONSOLE CAW                     A2224400
CNSTAT   DS    C                     CONSOLE STATUS                  A2224410
CNSENS   DS    3C                    CONSOLE SENSE BYTES             A2224420
CNSCSW   DS    D                     CONSOLE CSW                     A2224430
CNSPSW   DS    D                     CONSOLE PSW                     A2224440
         DC    A(CNSNRM)             ADDRESS OF ROUTINE TO TREAT     A2224450
*                                     NORMAL RETURN WHEN I/O          A2224460
*                                     INTERRUPTION OCCURS             A2224470
         DC    A(CNSEXC)             ADDRESS OF ROUTINE TO TREAT     A2224480
*                                     EXCEPTIONAL RETURN WHEN I/O     A2224490
*                                     INTERRUPTION OCCURS             A2224500
         XI    CPSW,X'80'            CPSW=0                          A2224510
MESRET   LM    1,7,MESSGR            (MESSGR) TO GENERAL REGISTERS   A2224520
*                                     1-7                             A2224530
         NI    MESPSW+1,X'FD'        0 TO WAIT STATE BIT IN MESPSW   A2224540
         CNOP  2,4                                                    A2224550
         SVC   3                                                      A2224560
         SPACE 3                                                      A2224570
         DC    A(MESPSW)             ADDRESS OF THE DOUBLE WORD IN   A2224580
*                                     WHICH A PSW IS SAVED WHEN A     A2224590
*                                     COMMAND OR MESSAGE ROUTINE IS   A2224600
*                                     ENTERED                         A2224610
         SPACE 3                                                      A2224620
*********************************************************************** A2224630
*                                                                     * A2224640
*      NORMAL RETURN WHEN I/O INTERRUPTION OCCURS                     * A2224650
*                                                                     * A2224660
*********************************************************************** A2224670
         SPACE                                                        A2224680
CNSNRM   TM    CPSW,X'80'            IS CPSW=1                       A2224690
         BC    1,CSNRMA              YES-IGNORE SET-UP SEQUENCE      A2224700
         STM   1,7,MESSGR            NO-GEN. REG. 1-7 TO MESSGR      A2224710
```

```
         MVC   MESPSW(8),CNSPSW        CNSPSW TO MESPSW                        A2224720
CSNRMA   TM    ALRMSW,X'10'            IS ALRMSW=1                             A2224730
         BC    1,ALRMRS                YES-RESTORE READ/WRITE COMMAND          A2224740
         TM    RDERSW,X'20'            NO-IS RDERSW=1                          A2224750
         BC    12,*+12                 NO-BRANCH TO CONTINUE                   A2224760
         MVI   CONBSY,X'00'            YES-X'00' TO CONBSY,GO TO               A2224770
         BC    15,MESRET               EXIT SEQUENCE                           A2224780
         MVI   TEMP,X'07'              X'07' TO TEMP                           A2224790
CSNRMB   XI    CONBSY,X'01'            0 TO CONBSY                             A2224800
         TM    OPTYPE,X'02'            IS THIS A READ OPERATION                A2224810
         BC    1,CSNRMC                YES-GO TO READ SEQUENCE                 A2224820
         L     WORK,ADBUFF             NO-WRITE OPN.,(ADBUFF) TO (WORK)        A2224830
         MVC   0(1,WORK),TEMP          (TEMP) TO (WORK),OPERATION              A2224840
         BC    15,MESRET               TERMINATED,RETURN TO CALLER             A2224850
CSNRMC   LH    WORK,COMCOM+6           READ SEQUENCE                           A2224860
         SH    WORK,CNSCSW+6           COUNT OF BYTES READ TO COMLEN           A2224870
         L     WORKA,ACMLEN            *                                       A2224880
         STC   WORK,0(WORKA)           *                                       A2224890
         L     WORK,ACMBUF             *                                       A2224900
         MVC   0(1,WORK),TEMP          (TEMP) TO COMBUF                        A2224910
         MVC   4(8,WORKA),MESPSW       MESPSW TO COMPSW                        A2224920
         NI    5(WORKA),X'FD'          0 TO WAIT STATE BIT IN SVCPSW           A2224930
         MVC   MESPSW+5(3),ACMRET      COMRET TO MESPSW(A)                     A2224940
         MVI   MESPSW,X'00'            DISABLE I/O AND EXTERNAL INTER-         A2224950
         BC    15,MESRET               RUPTIONS,GO TO INTERRUPT                A2224960
         EJECT                                                                 A2224970
*************************************************************************** A2224980
*                                                                         * A2224990
*      EXCEPTIONAL RETURN WHEN I/O INTERRUPTION OCCURS                      A2225000
*                                                                         * A2225010
*************************************************************************** A2225020
         SPACE 2                                                             A2225030
CNSEXC   TM    CPSW,X'80'              IS CPSW=1                               A2225040
         BC    1,CSEXCA                YES-IGNORE SET-UP SEQUENCE              A2225050
         STM   1,7,MESSGR              NO-GEN. REG. 1-7 TO MESSGR              A2225060
         MVC   MESPSW(8),CNSPSW        CNSPSW TO MESPSW                        A2225070
CSEXCA   TM    CNSTAT,X'02'            IS ERRTYP=2 OR 3                        A2225080
         BC    1,CSEXCB                YES-GO TO TREAT ERRTYP CONDITION        A2225090
         LA    I,CONSOL-2              RESTORE I TO SET UP SEREP INTER-        A2225100
*                                       FACE                                  A2225110
         TM    CNSTAT,X'01'            NO-IS ERRTYP=1                          A2225120
         BC    1,SREP3S                YES-DEVICE NOT OPERATIONAL,SET          A2225130
*                                       UP SEREP INTERFACE                    A2225140
         BC    15,CSEXCC               NO-GO TO TEST FOR UNIT EXCEPTION        A2225150
CSEXCB   MVI   TEMP,X'03'              ERRTYP=2 OR 3-X'03' TO (TEMP)           A2225160
         BC    15,CSNRMB               RETURN TO CALLER OR TO INTER-           A2225170
*                                       RUPTION INTERFACING SEQUENCE          A2225180
CSEXCC   TM    CNSCSW+4,UE             IS UNIT EXCEPTION PRESENT               A2225190
         BC    8,CSEXCD                NO-LOOK FOR UC                          A2225200
         TM    OPTYPE,X'02'            YES-IS THIS A READ OPERATION            A2225210
         BC    8,SREP1I                NO-CALL SEREP,I/O DEVICE ERROR          A2225220
         BC    15,MESGEB               YES-CANCEL,RETRY READ OPERATION         A2225230
CSEXCD   TM    CNSENS,CR               UNIT CHECK-IS THIS A COMMAND            A2225240
*                                       REJECT                                A2225250
         BC    1,CSEXCB                YES-TREAT COMMAND REJECT                A2225260
```

```
         TM    CNSENS,X'0E'          NO-ANY INVALID SENSE BIT PRESENT A2225270
         BC    5,SREP1I              YES-CALL SEREP,I/O DEVICE ERROR  A2225280
         TM    ALRMSW,X'10'          IS ALRMSW=1                      A2225290
         BC    1,ALRMRS              YES-GO TO RESTORE READ/WRITE     A2225300
         TM    CNSENS,EQUCHK         NO-IS EQUIPMENT CHECK            A2225310
         BC    8,CSEXCG              NO-GO TO TEST IF INTERVENTION    A2225320
*                                       REQUIRED                      A2225330
         TM    OPTYPE,X'02'          YES-IS THIS A READ OPERATION     A2225340
         BC    8,CSEXCE              NO-WRITE OPERATION,GO TO PROCESS A2225350
         XI    OPTYPE,X'02'          YES-0 TO OPTYPE                  A2225360
         OI    RDERSW,X'20'          1 TO RDERSW                      A2225370
         LA    WORK,ERDCCW           READ ERROR CCW ADDRESS TO CAWADD A2225380
         ST    WORK,CONSOL+2         *                                A2225390
         BC    15,MESGEB             WRITE OUT MESSAGE                A2225400
CSEXCE   TM    RDERSW,X'20'          IS RDERSW=1                      A2225410
         BC    1,MESRET              YES-RETURN TO POINT OF INT.      A2225420
         TM    RETRSW,X'08'          NO-IS RETRSW=1                   A2225430
         BC    8,CSEXCF              NO-GO TO RETRY MESSAGE           A2225440
         BAL   LINK,ALARM            YES-ISSUE CONTROL ALARM          A2225450
         MVI   TEMP,X'01'            X'01' TO (TEMP)                  A2225460
         BC    15,CSNRMB             RETURN TO POINT OF INTERRUPTION  A2225470
CSEXCF   OI    RETRSW,X'08'          1 TO RETRSW                      A2225480
         BC    15,MESGEB             RETRY MESSAGE                    A2225490
CSEXCG   TM    CNSENS,INTREQ         IS INTERVENTION REQUIRED         A2225500
         BC    8,CSEXCH              NO-GO TO TEST FOR BUS OUT CHECK  A2225510
         TM    INTSW,X'04'           YES-IS INTERVENTION REQUIRED     A2225520
*                                       SIGNALED                      A2225530
         BC    1,MESGEB              YES-RETRY                        A2225540
         OI    INTSW,X'04'           NO-1 TO INTSW                    A2225550
         BAL   LINK,ALARM            ISSUE CONTROL ALARM              A2225560
         BC    15,MESGEB             RETRY                            A2225570
CSEXCH   TM    RETRSW,X'08'          BUS OUT CHECK-IS RETRSW=1        A2225580
         BC    8,CSEXCJ              NO-GO TO RETRY                   A2225590
         IC    WORK,CNSCSW+4         YES-SAVE DEVICE STATUS           A2225600
         BAL   LINK,ALARM            ISSUE CONTROL ALARM              A2225610
         STC   WORK,CNSCSW+4         RESTORE DEVICE STATUS            A2225620
         TM    CNSCSW+4,DE           IS DEVICE END PRESENT            A2225630
         BC    8,SREP1I              NO-CALL SEREP,I/O DEVICE ERROR   A2225640
         TM    RDERSW,X'20'          YES-IS RDERSW=1                  A2225650
         BC    1,MESRET              YES-RETURN TO POINT OF INT.      A2225660
         MVI   TEMP,X'01'            NO-X'01' TO (TEMP)               A2225670
         BC    15,CSNRMB             RETURN TO POINT OF INTERRUPTION  A2225680
CSEXCJ   OI    RETRSW,X'08'          1 TO RETRSW                      A2225690
         BC    15,MESGEB             RETRY                            A2225700
         EJECT                                                        A2225710
*************************************************************************A2225720
*                                                                      * A2225730
*      ROUTINE TO SET PARAMETERS FOR THE READ COMMAND ROUTINE          * A2225740
*                                                                      * A2225750
*  THIS ROUTINE IS ENTERED VIA THE SVC TABLE WHENEVER AN SVC 5 CALL-   * A2225760
*  ING SEQUENCE IS ENCOUNTERED.                                        * A2225770
*                                                                      * A2225780
*************************************************************************A2225790
         SPACE                                                        A2225800
SETCOM   MVC   COMCOM+7(1),2(I)       (I+2) TO N                       A2225810
```

```
         L     WORK,2(I)              COMBUF TO (ACMBUF)                A2225820
         ST    WORK,ACMBUF            *                                 A2225830
         LA    WORK,1(WORK)           COMBUF+1 TO COMCOM(A)             A2225840
         ST    WORK,COMCOM            *                                 A2225850
         OI    COMCOM,X'0A'           READ OPERATION TO COMCOM          A2225860
         AH    I,DEC6                 COMLEN TO (ACMLEN)                A2225870
         ST    I,ACMLEN               *                                 A2225880
         MVC   ACMRET(3),1(I)         COMRET TO (ACMRET)                A2225890
         AH    I,DEC12                I+18 TO OSVPSW(A)                 A2225900
         ST    I,OSVPSW+4             *                                 A2225910
         L     J,JCONS                COMPSW TO ATTSW FOR CONSOLE       A2225920
         MVC   DEVATT(4,J),ACMPSW     *                                 A2225930
         BC    15,ENARET              OSVPSW TO PSW,RETURN TO CALLER    A2225940
         SPACE 2                                                        A2225950
*****************************************************************      A2225960
*                                                               *      A2225970
*      ROUTINE TO READ COMMANDS FROM CONSOLE KEYBOARD           *      A2225980
*                                                               *      A2225990
*  THIS ROUTINE  IS ENTERED WHENEVER  AN ATTENTION  INTERRUPTION FOR *  A2226000
*  THE 1052 PRINTER-KEYBOARD IS DETECTED BY THE CONTROL PROGRAM.  *     A2226010
*                                                               *      A2226020
*  THE  PSW  OF THE SIMULATOR  AT THE POINT OF INTERRUPTION  WILL BE *  A2226030
*  PLACED IN THE DOUBLE WORD WITH ADDRESS  COMPSW  AND CONTROL  WILL *  A2226040
*  BE TRANSFERRED  TO ADDRESS  COMPSW+8  (THE ROUTINE  COMAND STARTS *  A2226050
*  AT THIS  ADDRESS).   THE ADDRESS  DEVATT  OF THE  UCB  ASSOCIATED *  A2226060
*  WITH THE 1052 CONTAINS THE ADDRESS  COMPSW.                   *      A2226070
*                                                               *      A2226080
*****************************************************************      A2226090
         SPACE                                                          A2226100
COMPSW   DS    D                      OLD I/O PSW FOR ATTENTION         A2226110
*                                       INTERRUPTION                    A2226120
COMAND   STM   1,7,MESSGR             GENERAL REGISTERS 1-7 TO MESSGR   A2226130
         MVC   MESPSW(8),COMPSW         COMPSW TO MESPSW                A2226140
         LA    I,CONSOL-2             GET INDEX PAIR (J,K) OF ...       A2226150
         BAL   LINK,GETUCB            ... CONSOLE UCB.                  A2226160
         BC    0,0                    UCB NOT FOUND, IGNORE.            A2226170
         BC    0,0                    UCB NOT FOUND              V1L1   A2226175
         TM    BORCH(J),BSYFLG        IS BORCH(J) = FREE FLAG           A2226180
         BC    5,MESRET               RETURN TO POINT OF INTERRUPT      A2226190
CMNDB    MVI   OPTYPE,X'03'           OPTYPE=1,CONBSY=01                A2226200
         LA    WORK,COMCOM            READ CCW ADDRESS IN CAWADD        A2226210
         ST    WORK,CONSOL+2          *                                 A2226220
         BC    15,MESGEB              TRY TO EXECUTE READ OPERATION     A2226230
         EJECT                                                          A2226240
*****************************************************************      A2226250
*                                                               *      A2226260
*      CONTROL ALARM ROUTINE                                    *      A2226270
*                                                               *      A2226280
*  THIS ROUTINE  IS CALLED IN ORDER  TO EXECUTE A CONTROL  ALARM RE- *  A2226290
*  QUEST.   IT IS ENTERED WITH THE CALLING SEQUENCE,             *      A2226300
*                                                               *      A2226310
*              BAL     LINK,ALARM                               *      A2226320
*                                                               *      A2226330
*****************************************************************      A2226340
         SPACE                                                          A2226350
```

```
ALARM    STM   WORK,LINK,ALMLNK        SAVE (LINK) IN ALMLNK              A2226360
         MVC   ALMCMD(4),CONSOL+2      SAVE READ/WRITE COMMAND            A2226370
         LA    WORK,ALMCCW             ALARM CCW ADDRESS IN CAWADD        A2226380
         ST    WORK,CONSOL+2           *                                  A2226390
         OI    ALRMSW,X'10'            1 TO ALRMSW                        A2226400
         BC    15,MESGEB               TRY TO EXECUTE ALARM COMMAND       A2226410
         SPACE 2                                                          A2226420
ALRMRS   XI    ALRMSW,X'10'            0 TO ALRMSW                        A2226430
         MVC   CONSOL+2(4),ALMCMD      RESTORE READ/WRITE COMMAND         A2226440
         LM    WORK,LINK,ALMLNK        RESTORE LINK                       A2226450
         BCR   15,LINK                 RETURN TO READ/WRITE E.R.P.        A2226460
         EJECT                                                            A2226470
*********************************************************************** A2226480
*                                                                     * A2226490
*      DEFINITION OF PARAMETERS USED IN CONSOLE                       * A2226500
*                                  COMMUNICATION ROUTINES             * A2226510
*                                                                     * A2226520
*********************************************************************** A2226530
         SPACE                                                            A2226540
ALMLNK   DS    2F                      USED BY ALARM ROUTINE TO SAVE      A2226550
*                                        SUBROUTINE RETURN ADDRESS        A2226560
ALMCMD   DS    4C                      USED BY ALARM ROUTINE TO SAVE      A2226570
*                                        READ/WRITE COMMAND               A2226580
CONBSY   DC    X'00'                   A ONE-BIT QUANTITY DEFINED THUS=   A2226590
         SPACE                                                            A2226600
*                                      WHEN THE CONTROL                   A2226610
*                                      PROGRAM IS BUSY WITH A             A2226620
*                                      READ/WRITE SEQUENCE,    CONBSY=1   A2226630
*                                      OTHERWISE,              CONBSY=0   A2226640
         SPACE                                                            A2226650
OPTYPE   EQU   CONBSY                  A ONE-BIT QUANTITY DEFINED THUS=   A2226660
         SPACE                                                            A2226670
*                                      OPTYPE=0  A MESSAGE IS BEING       A2226680
*                                                WRITTEN                  A2226690
*                                      OPTYPE=1  A COMMAND IS BEING       A2226700
*                                                READ                     A2226710
         SPACE                                                            A2226720
INTSW    EQU   CONBSY                  FOUR ONE BIT QUANTITIES USED       A2226730
RETRSW   EQU   CONBSY                    DURING ERROR RECOVERY PROCE-     A2226740
ALRMSW   EQU   CONBSY                    DURES ON THE 1052 PRINTER-       A2226750
RDERSW   EQU   CONBSY                    KEYBOARD, AND DEFINED THUS=      A2226760
         SPACE                                                            A2226770
*                                      WHEN AN INTERVENTION               A2226780
*                                      REQUIRED CONDITION HAS             A2226790
*                                      BEEN DETECTED AND A                A2226800
*                                      CONTROL ALARM HAS BEEN             A2226810
*                                      ISSUED,                 INTSW=1    A2226820
*                                      OTHERWISE,              INTSW=0    A2226830
         SPACE                                                            A2226840
*                                      WHEN A BUS OUT OR AN               A2226850
*                                      EQUIPMENT CHECK CONDI-             A2226860
*                                      TION HAS BEEN DETECTED             A2226870
*                                      AND ONE RETRY HAS BEEN             A2226880
*                                      EFFECTED,               RETRSW=1   A2226890
*                                      OTHERWISE,              RETRSW=0   A2226900
```

```
         SPACE                                                          A2226910
*                                WHEN A CONTROL ALARM                   A2226920
*                                REQUEST HAS BEEN SUB-                  A2226930
*                                MITTED,                 ALRMSW=1       A2226940
*                                OTHERWISE,              ALRMSW=0       A2226950
         SPACE                                                          A2226960
*                                WHEN AN EQUIPMENT                      A2226970
*                                CHECK CONDITION HAS                    A2226980
*                                BEEN DETECTED DURING                   A2226990
*                                THE READING OF A                       A2227000
*                                COMMAND,                RDERSW=1       A2227010
*                                OTHERWISE,              RDERSW=0       A2227020
         SPACE                                                          A2227030
CPSW     EQU   CONBSY             A ONE-BIT QUANTITY DEFINED THUS=       A2227040
         SPACE                                                          A2227050
*                                CPSW=1   JUST PRIOR TO SUBMIT-         A2227060
*                                         TING AN I/O REQUEST AND       A2227070
*                                         CONTINUE                      A2227080
*                                CPSW=0   IF CONTROL IS RETURNED        A2227090
*                                         TO THE INSTRUCTION FOL-       A2227100
*                                         LOWING THIS REQUEST           A2227110
*                                         BEFORE A NORMAL RETURN        A2227120
*                                         OR AN EXCEPTIONAL RE-         A2227130
*                                         TURN INTERRUPTION             A2227140
*                                         OCCURS                        A2227150
         SPACE                                                          A2227160
ERDMES   DC    C'A2'              READ ERROR MESSAGE                     A2227170
         DC    C'14A EQUIPMENT '  *                                      A2227180
         DC    C'CHECK-TRY '      *                                      A2227190
         DC    C'AGAIN'           *                                      A2227200
         DC    X'15'              *                                      A2227210
         EJECT                                                          A2227220
ADBUFF   DS    F                  USED TO SAVE THE CONTENTS OF           A2227230
*                                   GENERAL REGISTER BUFF IN THE        A2227240
*                                   MESSAGE TRANSMISSION ROUTINE        A2227250
         SPACE                                                          A2227260
TEMP     DS    D                  TEMPORARY STORAGE                      A2227270
SNSBYT   EQU   TEMP               SENSE BYTES BUFFER                     A2227280
SNSCOM   CCW   X'04',TEMP,0,6     SENSE CCW                              A2227290
MESCOM   CCW   X'01',*,0,0        WRITE MESSAGE CCW                      A2227300
ALMCCW   CCW   X'0B',*,0,1        CONTROL ALARM CCW                      A2227310
ERDCCW   CCW   X'01',ERDMES,X'20',32 READ ERROR COMMAND CCW              A2227320
COMCOM   CCW   X'0A',*,0,0        READ COMMAND CCW                       A2227330
ACMBUF   DC    A(0)               ADDRESS OF COMMAND BUFFER              A2227340
ACMLEN   DC    A(0)               ADDRESS OF COMMAND LENGTH              A2227350
ACMRET   DC    AL3(0)             ADDRESS OF RETURN FOR MESAGE/          A2227360
*                                   COMAND ROUTINES                     A2227370
         SPACE 2                                                        A2227380
PRBASE   DS    F                  USED TO SAVE THE CONTENTS OF           A2227390
*                                   GENERAL REGISTER BASE DURING        A2227400
*                                   THE PROCESSING OF PROGRAM           A2227410
*                                   INTERRUPTIONS                       A2227420
         SPACE 2                                                        A2227430
EXBASE   DS    2F                 USED TO SAVE THE CONTENTS OF           A2227440
*                                   GENERAL REGISTERS BASE AND          A2227450
```

```
*                                       WORK DURING THE PROCESSING OF  A2227460
*                                       EXTERNAL INTERRUPTIONS         A2227470
         SPACE                                                         A2227480
POINTA   DS    F                      USED TO SAVE THE CONTENTS OF   A2227490
*                                       GENERAL REGISTER POINTR IN THE A2227500
*                                       UNSTAK ROUTINE                A2227510
         SPACE                                                         A2227520
PRPSWA   DC    A(0)                   ADDRESS OF PROGRAM PSW         A2227530
EXTPSW   DC    A(0)                   ADDRESS OF EXTERNAL PSW        A2227540
ACMPSW   DC    A(COMPSW)              ADDRESS OF COMMAND INTERRUPTION A2227550
*                                       PSW                           A2227560
JCONS    DC    A(0)                   ADDRESS OF UCB FOR CONSOLE     A2227570
TIMINA   DC    A(0)                   ADDRESS OF TIMER INTERRUPTION  A2227580
*                                       RETURN                        A2227590
KEYINA   DC    A(0)                   ADDRESS OF INTERRUPT KEY INTER- A2227600
*                                       RUPTION RETURN                A2227610
SENSW    DC    X'00'                  SENSE SWITCH                   A2227620
ZEROS    DC    X'0000'                                               A2227630
SNSCNT   DC    X'00'                  NUMBER OF SENSE OPERATIONS TO  A2227640
*                                       BE PERFORMED                  A2227650
PRRETA   DC    AL3(0)                 ADDRESS OF PROGRAM INTERRUPTION A2227660
*                                       RETURN                        A2227670
EXTSWA   DC    AL3(EXTINT)            ADDRESS OF ROUTINE TO PROCESS  A2227680
*                                       EXTERNAL INTERRUPTIONS        A2227690
         SPACE                                                         A2227700
AIOWIN   DC    AL3(IOWINT)            ADDRESS OF THE INTERRUPTION    A2227710
*                                       PROCESSING SEQUENCE OF THE    A2227720
*                                       I/O REQUEST AND WAIT ROUTINE  A2227730
         SPACE 2                                                       A2227740
AIOCIN   DC    AL3(IOCINT)            ADDRESS OF THE INTERRUPTION    A2227750
*                                       PROCESSING SEQUENCE OF THE I/O A2227760
*                                       REQUEST AND CONTINUE ROUTINE  A2227770
         SPACE                                                         A2227780
INTCD1   DC    X'0001'                INTERRUPTION CODE = 1          A2227790
ATTSW    DC    X'00'                  USED TO INDICATE WHETHER OR NOT A2227800
*                                       ATTENTION INTERRUPTIONS ARE TO A2227810
*                                       BE IGNORED                    A2227820
UNSTSW   DC    X'00'                  USED IN UNSTAK ROUTINE TO INDI- A2227830
*                                       CATE WHETER OR NOT THE LAST   A2227840
*                                       UNSTACKED OPERAT. WAS A SENSE A2227850
ONE      EQU   X'01'                  *                              A2227860
ZERO     EQU   X'00'                  *                              A2227870
ONES     EQU   X'FF'                                                 A2227880
         SPACE 2                                                       A2227890
DEC2     DC    H'2'                   DECIMAL CONSTANTS              A2227900
DEC6     DC    H'6'                   *                              A2227910
DEC12    DC    H'12'                  *                              A2227920
DEC20    DC    H'20'                  *                              A2227930
DEC24    DC    H'24'                  *                              A2227940
DEC36    DC    H'36'                  *                              A2227950
DEC40    DC    H'40'                  *                              A2227960
         EJECT                                                         A2227970
*********************************************************************** A2227980
*                                                                     * A2227990
*  CHANNEL TABLE (CHTAB)                                              * A2228000
```

```
*                                                                     * A2228010
*  THIS TABLE CONSISTS OF 8 CONSECUTIVE WORDS, EACH WORD CORRESPOND-  * A2228020
*  ING, IN ORDER, TO THE EIGHT POSSIBLE CHANNEL ADDRESSES 0 TO 7.     * A2228030
*                                                                     * A2228040
*  EACH WORD CONTAINS  THE ADDRESS OF THE FIRST ELEMENT OF THE CHAN-  * A2228050
*  NEL CONTROL BLOCK FOR THE CORRESPONDING CHANNEL.                   * A2228060
*                                                                     * A2228070
*  IF THE CONTENTS  OF A PARTICULAR WORD IS ZERO,  THE CORRESPONDING  * A2228080
*  CHANNEL IS NOT AVAILABLE.   SINCE CHANNEL  7  CANNOT EXIST ON THE  * A2228090
*  SYSTEM/360, THE LAST WORD OF  CHTAB  ALWAYS CONTAINS ZERO.         * A2228100
*                                                                     * A2228110
*  CHANNEL CONTROL BLOCKS (AND THE CORRESPONDING UNIT CONTROL BLOCKS) * A2228120
*  WILL BE SET UP BY THE INITIALIZER  PROGRAM , BASED ON INFORMATION  * A2228130
*  ENTERED ON CONTROL CARDS.                                          * A2228140
*                                                                     * A2228150
********************************************************************* A2228160
         SPACE                                                        A2228170
CHTAB    DC    8F'0'                                                 A2228180
         DC    A(COMCOM)              ENTRY POINTS - FROM THE SIMULA- A2228190
         DC    A(JCONS)                 TOR, THE I/O SUPPORT PACKAGE  A2228200
         DC    A(CONSOL)                AND THE CONTROL PROGRAM       A2228210
         DC    A(CHTAB)                 INITIALIZATION ROUTINE        A2228220
         DC    A(DMP360)              SYSTEM/360 DUMP ADDRESS         A2228230
         DC    X'FF'                  *                               A2228240
         END                                                          A2228250
         AOPIN CROSSREF                                               A2300010
A23B     TITLE 'I/O SUPPORT PACKAGE FOR CURRENT SYSTEMS SIMULATORS'   A2300020
IOPACK   START 3488                   THIS ASSEMBLY LOCATION MUST ALSO A2300030
         USING *,IOBASE               * BE SPECIFIED BY THE OPERANDS  A2300040
*                                     * OF DC STATEMENTS AT ADDRESSES A2300050
*                                     * SVCTAB+34,SVCTAB+36 IN CONTROL A2300060
*                                     * PROGRAM.                      A2300070
         SPACE                                                        A2300080
********************************************************************* A2300090
*                                                                     * A2300100
*                    I/O SUPPORT PACKAGE PROGRAM                      * A2300110
*                                                                     * A2300120
*                              FOR                                    * A2300130
*                                                                     * A2300140
*              IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620              * A2300150
*                                                                     * A2300160
*              ----------------------------------------               * A2300170
*                                                                     * A2300180
*  THE I/O SUPPORT PACKAGE IS A PROGRAM  CONSISTING OF A SET OF SUB-  * A2300190
*  ROUTINES WHICH PERFORM VARIOUS  OPERATIONS  ON SYSTEM/360 I/O DE-  * A2300200
*  VICES.                                                             * A2300210
*                                                                     * A2300220
*  THE I/O OPERATIONS  WHICH THE I/O SUPPORT  PACKAGE IS DESIGNED TO  * A2300230
*  PERFORM ARE=                                                       * A2300240
*                                                                     * A2300250
*    READ A CARD              IBM 2540, 2501, 2520, 1442              * A2300260
*    PUNCH A CARD (OPTIONAL)  IBM 2540, 2520, 1442                    * A2300270
*    WRITE A MESSAGE          IBM 1052                                * A2300280
*    READ A COMMAND           IBM 1052                                * A2300290
*    PRINT A LINE             IBM 1403, 1443                          * A2300300
```

```
*    PRINT A LINE AND SKIP                                              * A2300310
*        TO FIRST LINE ON                                               * A2300320
*        NEXT PAGE            IBM 1403, 1443                            * A2300330
*    READ A TAPE RECORD       IBM 2400 (7 AND 9-TRACK)                  * A2300340
*    WRITE A TAPE RECORD      IBM 2400 (7 AND 9-TRACK)                  * A2300350
*    WRITE A TAPE MARK        IBM 2400                                  * A2300360
*                                                                       * A2300370
*  THESE  ROUTINES ARE ALL  DESIGNED  FOR NON-OVERLAPPED  OPERATION.    * A2300380
*  THUS,  PROGRAM EXECUTION  WILL BE SUSPENDED  UNTIL THE I/O OPERA-    * A2300390
*  TION IS TERMINATED.                                                  * A2300400
*                                                                       * A2300410
*  THE I/O SUPPORT PACKAGE EXAMINES THE ERROR  CONDITIONS  WHICH CAN    * A2300420
*  OCCUR WHEN OPERATING  THE ABOVE  MENTIONED  DEVICES AND TAKES THE    * A2300430
*  ACTION PRESCRIBED BY SYSTEM/360 STANDARDS.  OPERATOR  MESSAGE FA-    * A2300440
*  CILITIES ARE PROVIDED VIA THE 1052 PRINTER-KEYBOARD.                 * A2300450
*                                                                       * A2300460
************************************************************************ A2300470
         EJECT                                                           A2300480
************************************************************************ A2300490
*                                                                       * A2300500
*  LOGICAL I/O OPERATION REQUESTS                                       * A2300510
*                                                                       * A2300520
*  BEFORE A LOGICAL  I/O  OPERATION REQUEST CAN BE  SUBMITTED TO THE    * A2300530
*  I/O  SUPPORT PACKAGE,  AN  SVC  CALLING SEQUENCE OF THE FOLLOWING    * A2300540
*  FORM MUST BE SUBMITTED=                                              * A2300550
*                                                                       * A2300560
*              CNOP    0,4                                              * A2300570
*       I      SVC     17                                               * A2300580
*       SYMBOL DS      6C                                               * A2300590
*       DEV360 DC      X'0DDD'                                          * A2300600
*       TYPE   DC      C'TTTT'                                          * A2300610
*       IOTYPE DS      C                                                * A2300620
*              DC      AL3(ERROR)                                       * A2300630
*       I+20   ANY INSTRUCTION                                          * A2300640
*                                                                       * A2300650
*  THIS CALLING SEQUENCE ASSIGNS A SYSTEM/360  DEVICE ADDRESS TO THE    * A2300660
*  SYMBOLIC NAME SYMBOL.                                                * A2300670
*                                                                       * A2300680
*  SYMBOL  IS A  SYMBOLIC  NAME  ASSIGNED  BY  THE  SIMULATOR  TO A     * A2300690
*  SYSTEM/360 DEVICE.  THIS NAME MAY CONTAIN FROM ONE TO EIGHT CHAR-    * A2300700
*  ACTERS,  BEING ANY COMBINATION OF ALPHABETIC AND  NUMERIC CHARAC-    * A2300710
*  TERS.  THE FIRST CHARACTER MUST BE ALPHABETIC, THE SYMBOL IS LEFT     * A2300720
*  ADJUSTED,  AND ALL REMAINING CHARACTERS IN THE  EIGHT-BYTE  FIELD     * A2300730
*  MUST BE BLANK.                                                       * A2300740
*                                                                       * A2300750
*  IOTYPE IS ONE CHARACTER, I OR O,  WHICH SPECIFIES THE TYPE OF OP-    * A2300760
*  ERATION  (INPUT OR OUTPUT)  TO BE PERFORMED ON THE  DEVICE  NAMED    * A2300770
*  SYMBOL.  DDD DENOTES THE SYSTEM/360  ADDRESS AND TTTT THE TYPE OF    * A2300780
*  THIS DEVICE.                                                         * A2300790
*                                                                       * A2300800
*  THE TYPES OF DEVICE AND THE CORRESPONDING  OPERATIONS ACCEPTED ON    * A2300810
*  THESE DEVICES ARE AS FOLLOWS=                                        * A2300820
*                                                                       * A2300830
*     2540, 2520, 1442    I(O OPTIONAL)                                 * A2300840
*     2501                I                                             * A2300850
```

```
*      1403, 1443          O                                          * A2300860
*      2400                I AND O (*)                                * A2300870
*      1052                I AND O                                    * A2300880
*                                                                     * A2300890
*  (*) FOR AN I/O OPERATION ON THE 2400 MAGNETIC TAPE, 7T, BCD MODE,  * A2300900
*      IOTYPE CONSISTS OF TWO FOUR-BIT DIGITS DENOTED, FROM LEFT  TO  * A2300910
*      RIGHT, AS  D0,D1.                                              * A2300920
*      D0 SPECIFIES THE TYPE OF OPERATION   '1' INPUT  OPERATION      * A2300930
*                                           '0' OUTPUT OPERATION      * A2300940
*      D1 SPECIFIES THE DENSITY             '2' 200  BPI              * A2300950
*                                           '5' 556  BPI              * A2300960
*                                           '8' 800  BPI              * A2300970
*                                                                     * A2300980
*                                                          .../...    * A2300990
*                                                                     * A2301000
*********************************************************************** A2301010
        EJECT                                                          A2301020
*********************************************************************** A2301030
*                                                                     * A2301040
*  .../...                                                            * A2301050
*                                                                     * A2301060
*  WITH EACH GROUP 'SYMBOL,DEV360'  IS ASSOCIATED A BLOCK OF CONTROL  * A2301070
*  INFORMATION IN A TABLE CALLED SYMBOL TABLE.                        * A2301080
*                                                                     * A2301090
*  THE I/O SUPPORT PACKAGE VERIFIES THE FOLLOWING CONDITIONS=         * A2301100
*                                                                     * A2301110
*    --THE SYMBOL TABLE IS NOT FULL                                   * A2301120
*        IF THE TABLE IS FULL, IOTYPE IS SET TO X'01'                 * A2301130
*                                                                     * A2301140
*    --A ROUTINE EXISTS FOR THE OPERATION  TO  BE  PERFORMED AND FOR  * A2301150
*      THE DEVICE TO BE USED                                          * A2301160
*        IF NOT, IOTYPE IS SET TO X'02'                               * A2301170
*                                                                     * A2301180
*    --A UNIT CONTROL  BLOCK IN THE CONTROL  PROGRAM EXISTS FOR THIS  * A2301190
*      DEVICE                                                         * A2301200
*        IF NOT, IOTYPE IS SET TO X'03'                               * A2301210
*                                                                     * A2301220
*  IN THE ABOVE CASES, WHEN IOTYPE IS SET TO X'01', X'02', OR X'03',  * A2301230
*  CONTROL IS RETURNED TO  THE SIMULATOR AT LOCATION  ERROR.  OTHER-  * A2301240
*  WISE, THE GROUP  'SYMBOL,DEV360'   IS PLACED IN THE SYMBOL  TABLE  * A2301250
*  AND CONTROL IS RETURNED TO THE SIMULATOR AT LOCATION I+20.          * A2301260
*                                                                     * A2301270
*  THE  SYMBOL  TABLE  MAY CONTAIN A MAXIMUM OF TEN GROUPS  'SYMBOL,  * A2301280
*  DEV360'.  ONCE AN ENTRY IS PLACED IN THE TABLE,  IT CANNOT BE RE-  * A2301290
*  MOVED.  THEREFORE, THE SVC 17 CALLING SEQUENCE EITHER ADDS A  NEW  * A2301300
*  GROUP  'SYMBOL,DEV360'  TO THE TABLE  (IF THE TABLE IS NOT FULL),  * A2301310
*  OR ASSIGNS A DIFFERENT  SYSTEM/360  DEVICE TO A SYMBOL ALREADY IN  * A2301320
*  THE TABLE.                                                         * A2301330
*                                                                     * A2301340
*                                *****                                * A2301350
*                                                                     * A2301360
*  THE FUNCTIONS OF THE SVC 17  CALLING SEQUENCE MAY BE PERFORMED BY  * A2301370
*  ENTERING A  CONTROL  CARD AT THE TIME OF PROGRAM  INITIALIZATION.  * A2301380
*  THIS CONTROL CARD HAS THE FOLLOWING FORMAT=                        * A2301390
*                                                                     * A2301400
```

```
*       / DEVSUP SYMBOL=X'DDD',TTTT,IOTYPE,DENSITY                      * A2301410
*                                                                       * A2301420
*  WHERE TTTT, DDD,  SYMBOL, AND IOTYPE  DENOTE THE  SAME QUANTITIES    * A2301430
*  AS IN THE SVC 17 CALLING SEQUENCE, BUT THAT IOTYPE IS ALWAYS I OR    * A2301440
*  O AND DENSITY IS USED WITH TTTT= 2400 FOR A 7-TRACK, BCD MODE, O-    * A2301450
*  PERATION. (DENSITY= 200, 556 OR 800).                                * A2301460
*                                                                       * A2301470
*  THE BLANKS BEFORE AND AFTER 'DEVSUP' MUST BE RESPECTED.              * A2301480
*                                                                       * A2301490
*  THE SYMBOL TABLE IS CREATED AT CONTROL PROGRAM INITIALIZATION BE-    * A2301500
*  FORE  THE SIMULATOR IS LOADED.  THE CONTENTS OF THE TABLE  REMAIN    * A2301510
*  UNCHANGED WHEN  CONTROL IS TRANSFERRED FROM THE RELOCATING LOADER    * A2301520
*  TO THE SIMULATOR.                                                    * A2301530
*                                *****                                  * A2301540
*                                                              .../...  * A2301550
*                                                                       * A2301560
************************************************************************ A2301570
         EJECT                                                           A2301580
************************************************************************ A2301590
*                                                                       * A2301600
*  .../...                                                              * A2301610
*                                                                       * A2301620
*  THE SVC CALLING SEQUENCE FOR A LOGICAL  I/O  OPERATION  IS OF THE    * A2301630
*  FOLLOWING FORM=                                                      * A2301640
*                                                                       * A2301650
*              CNOP    0,4                                              * A2301660
*      I       SVC     18                                               * A2301670
*      SYMBOL  DS      8C                                               * A2301680
*      COUNT   DC      FL2'NN'                                          * A2301690
*      BUFFER  DC      A(BUFF)                                          * A2301700
*      I+16    ANY INSTRUCTION                                          * A2301710
*                                                                       * A2301720
*  SYMBOL DENOTES  THE SAME  QUANTITY AS IN THE  SVC 17  CALLING SE-    * A2301730
*  QUENCE.   COUNT CONTAINS  THE NUMBER OF BYTES  OF DATA TO BE PRO-    * A2301740
*  CESSED AND BUFFER CONTAINS THE ADDRESS OF THE INPUT/OUTPUT BUFFER    * A2301750
*  FOR THE DEVICE BEING USED.                                           * A2301760
*                                                                       * A2301770
*  THE DATA WILL BE FETCHED FROM OR PLACED IN LOCATIONS                 * A2301780
*                                                                       * A2301790
*      BUFF+1, BUFF+2, ...BUFF+X'NN'                                    * A2301800
*                                                                       * A2301810
*  FOR AN OUTPUT OPERATION ON THE  1403  OR  1443,  THE CHARACTER IN    * A2301820
*  LOCATION BUFF+1 SPECIFIES THE TYPE OF PRINT COMMAND, AS FOLLOWS=     * A2301830
*                                                                       * A2301840
*    ANY CHARACTER BUT '1'   WRITE AND SPACE ONE LINE AFTER PRINTING    * A2301850
*    THE CHARACTER '1'       WRITE AND SKIP TO CHANNEL 1 AFTER          * A2301860
*                            PRINTING                                   * A2301870
*                                                                       * A2301880
*  THUS, THE DATA WILL BE FETCHED FROM LOCATIONS                        * A2301890
*                                                                       * A2301900
*      BUFF+2, BUFF+3, ...BUFF+X'NN'                                    * A2301910
*                                                                       * A2301920
*  FOR AN OUTPUT OPERATION ON THE 2400, IT MAY BE NECESSARY TO WRITE    * A2301930
*  A TAPE  MARK  (PARTICULARLY AFTER A UNIT EXCEPTION  HAS OCCURRED,    * A2301940
*  DENOTING THE END OF TAPE). TO WRITE A TAPE MARK,  COUNT MUST CON-    * A2301950
```

```
*  TAIN ONE  (NN=1) AND BUFF+1 MUST CONTAIN A 7/8 PUNCH (7F).          * A2301960
*                                                                      * A2301970
*  THE INPUT/OUTPUT OPERATION WILL BE PERFORMED USING AN SVC 11 (I/O   * A2301980
*  REQUEST AND WAIT)  CALLING  SEQUENCE.  THE CONTROL  PROGRAM  WILL   * A2301990
*  CYCLE ON THE SVC 11 CALLING SEQUENCE UNTIL THE REQUEST HAS TERMI-   * A2302000
*  NATED.   THE REQUEST  MAY BE TERMINATED  IN ANY OF THE  FOLLOWING   * A2302010
*  WAYS=                                                               * A2302020
*                                                                      * A2302030
*                                                              .../... * A2302040
*                                                                      * A2302050
************************************************************************ A2302060
        EJECT                                                           A2302070
************************************************************************ A2302080
*                                                                      * A2302090
*  .../...                                                             * A2302100
*                                                                      * A2302110
*      --A CATASTROPHIC  ERROR HAS OCCURRED.    CONTROL  IS  RETURNED * A2302120
*        EITHER  FROM THE CONTROL PROGRAM TO THE I/O SUPPORT PACKAGE   * A2302130
*        OR FROM THE  I/O  SUPPORT PACKAGE TO THE SIMULATOR.  IN THE   * A2302140
*        FIRST CASE, THE STANDARD SEREP INTERFACE IS SET UP.  IN THE   * A2302150
*        SECOND CASE, A MESSAGE IS ISSUED REQUESTING THAT THE SYSTEM/  * A2302160
*        360 DUMP PROGRAM BE LOADED (A PART OF THE SYSTEM HAS PROBAB-  * A2302170
*        LY BEEN OVERWRITTEN) OR THAT THE STANDARD  SEREP PROGRAM BE   * A2302180
*        LOADED (A MACHINE MALFUNCTION HAS BEEN DETECTED).             * A2302190
*                                                                      * A2302200
*      --THE DEVICE 'SYMBOL' IS UNKNOWN TO THE I/O SUPPORT  PACKAGE.   * A2302210
*        IT HAS NOT BEEN DEFINED BY A CONTROL CARD  NOR BY AN SVC 17   * A2302220
*        CALLING SEQUENCE.                                             * A2302230
*        THE BYTE AT ADDRESS BUFF IS SET TO THE VALUE X'01'.           * A2302240
*                                                                      * A2302250
*      --A DEVICE MALFUNCTION HAS BEEN DETECTED DURING THE EXECUTION   * A2302260
*        OF THE I/O REQUEST  AND A MESSAGE HAS BEEN ISSUED TO ADVISE   * A2302270
*        THE OPERATOR OF THE  MALFUNCTION.  THE I/O SUPPORT  PACKAGE   * A2302280
*        HAS RECEIVED A COMMAND TO TERMINATE THE I/O OPERATION.        * A2302290
*        THE BYTE AT ADDRESS BUFF IS SET TO THE VALUE X'02'.           * A2302300
*                                                                      * A2302310
*      --A UNIT  EXCEPTION  CONDITION HAS OCCURRED  DURING A READ OR   * A2302320
*        WRITE OPERATION ON A MAGNETIC TAPE. A MESSAGE IS ISSUED AND   * A2302330
*        CONTROL IS RETURNED TO THE SIMULATOR  WITH  THE BYTE AT AD-   * A2302340
*        DRESS BUFF SET TO THE VALUE X'03'.                            * A2302350
*                                                                      * A2302360
*      --NONE OF THE ABOVE CONDITIONS HAS OCCURRED, THAT IS, THE I/O   * A2302370
*        OPERATION  HAS TERMINATED  WITH NO EXCEPTIONAL  CONDITIONS.   * A2302380
*        THE BYTE AT ADDRESS BUFF IS SET TO THE VALUE X'07'.           * A2302390
*                                                                      * A2302400
*  IN THE LAST FOUR OF THESE CASES,  THE I/O SUPPORT PACKAGE RETURNS   * A2302410
*  CONTROL TO THE SIMULATOR AT LOCATION I+16.                          * A2302420
*                                                                      * A2302430
************************************************************************ A2302440
        EJECT                                                           A2302450
************************************************************************ A2302460
*                                                                      * A2302470
*                     GENERAL REGISTER ASSIGNMENT                      * A2302480
*                                                                      * A2302490
************************************************************************ A2302500
```

```
         SPACE                                                          A2302510
I        EQU   1                    ADDRESS OF 1ST BYTE OF SVC          A2302520
*                                   * CALLING SEQUENCE                  A2302530
X        EQU   2                    *                                   A2302540
WORK2    EQU   2                    WORKING REGISTER                    A2302550
INTCDE   EQU   3                    INTERRUPTION CODE IN SVC INSTR.     A2302560
BYTCNT   EQU   3                                                       A2302570
BYTE     EQU   4                                                       A2302580
WORK4    EQU   4                    WORKING REGISTER                    A2302590
WORK5    EQU   5                      -       -                         A2302600
NXTBYT   EQU   5                                                       A2302610
WORK6    EQU   6                    WORKING REGISTER                    A2302620
IOBASE   EQU   7                    BASE REGISTER (LOADED IN CONTROL    A2302630
*                                     PROGRAM)                         A2302640
WORK7    EQU   8                    WORKING REGISTER                    A2302650
LINK     EQU   9                    RETURN ADDRESS FOR SUBROUTINE       A2302660
*                                   * CALL                              A2302670
LNKMES   EQU   10                   RETURN ADDRESS FOR SUBROUTINE       A2302680
*                                   * CALL (WRITE MESSAGE)              A2302690
         SPACE                                                          A2302700
*********************************************************************** A2302710
*                                                                     * A2302720
*      DEFINITION OF DEVICE AND CHANNEL SYMBOLIC STATUS BITS          * A2302730
*                                                                     * A2302740
*********************************************************************** A2302750
         SPACE                                                          A2302760
BUSY     EQU   X'10'                BUSY                                A2302770
DE       EQU   X'04'                DEVICE END                          A2302780
UC       EQU   X'02'                UNIT CHECK                          A2302790
UE       EQU   X'01'                UNIT EXCEPTION                      A2302800
CHC      EQU   X'01'                CHAINING CHECK                      A2302810
CHN      EQU   X'01'                  -       -                         A2302820
         SPACE                                                          A2302830
*********************************************************************** A2302840
*                                                                     * A2302850
*              DEFINITION OF SYMBOLIC SENSE BITS                      * A2302860
*                                                                     * A2302870
*********************************************************************** A2302880
         SPACE                                                          A2302890
CR       EQU   X'80'                COMMAND REJECT                      A2302900
CMDRJT   EQU   X'80'                  -       -                         A2302910
INTREQ   EQU   X'40'                INTERVENTION REQUIRED               A2302920
BUSOUT   EQU   X'20'                BUS-OUT CHECK                       A2302930
EQUCHK   EQU   X'10'                EQUIPMENT CHECK                     A2302940
DATCH    EQU   X'08'                DATA CHECK                          A2302950
DATCHK   EQU   X'08'                DATA CHECK                          A2302960
OVERRN   EQU   X'04'                OVERRUN                             A2302970
PRTCHK   EQU   BUSOUT+EQUCHK        *                                   A2302980
ABSTAT   EQU   X'60'                2400 TU - STATUS A AND B            A2302990
NOISE    EQU   X'80'                NOISE                               A2303000
LOADPT   EQU   X'08'                LOAD POINT                          A2303010
FILEPR   EQU   X'02'                FILE PROTECT                        A2303020
         SPACE 2                                                        A2303030
OSVPSW   EQU   X'20'                SUPERVISOR CALL OLD PSW             A2303040
SVCGR    EQU   X'188'               SAVE AREA FOR GENERAL REGISTERS     A2303050
```

```
*                                       * 1-7 AFTER SVC INTERRUPTION     A2303060
*                                       * (AREA LOCATED IN CONTR. PROG.) A2303070
DEVATT   EQU   24                       DISPLACEMENT RELATIVE TO THE     A2303080
*                                       * ADDRESS OF THE 1052 UCB        A2303090
*                                       * LOCATED IN CONTROL  PROGRAM    A2303100
*                                       * (USED ONLY WITH THIS DEVICE)   A2303110
         SPACE 2                                                         A2303120
*********************************************************************** A2303130
*                                                                     * A2303140
*                          ENTRY  ROUTINE                             * A2303150
*                                                                     * A2303160
*  THIS ROUTINE  IS ENTERED FROM THE CONTROL PROGRAM  EVERY  TIME  A  * A2303170
*  LOGICAL I/O REQUEST IS SUBMITTED. IT IS THE FIRST ROUTINE  EXECU-  * A2303180
*  TED. THE ENTRY POINT IS THE ASSEMBLY LOCATION, IOPACK,  SPECIFIED  * A2303190
*  BY THE OPERAND OF THE START STATEMENT.                             * A2303200
*                                                                     * A2303210
*********************************************************************** A2303220
         SPACE                                                           A2303230
         MVC   CALLGR(28),SVCGR        SAVE G.R. 1-10                    A2303240
         STM   8,10,CALLGR+28          *                                 A2303250
         MVC   CALPSW(8),OSVPSW        SAVE SVC OLD PSW                  A2303260
         ST    1,CALLSQ                SAVE ADDR. OF SVC CALLING SEQ.    A2303270
         CH    INTCDE,DEC34            SVC 17 CALL (DEVICE ASSIGNMENT)   A2303280
         BC    8,SETSYM                YES,BRANCH                        A2303290
         OI    IOSW,X'01'              NO,REQUEST FOR I/O OPERATION-SET  A2303300
         MVC   MESLNK(4),12(1)         I/O OP. SWITCH ON AND SAVE ADDR.  A2303310
*                                         OF I/O BUFFER.                A2303320
*                                                                       A2303330
*  LOOK-UP SYMBOL IN SYMBOL TABLE (SYMTAB).                             A2303340
*                                       INITIALIZE TABLE LOOK-UP        A2303350
SETSYM   LA    WORK6,10                MAX. NUMBER OF ELEMENTS IN TABLE  A2303360
         L     WORK7,TABBEG            START AND CURRENT ADDRESSES OF    A2303370
         L     WORK2,TABEND            * SYMTAB.                         A2303380
TABTRM   CR    WORK7,WORK2             IS TABLE EXHAUSTED                A2303390
         BC    8,FILL                  YES,BRANCH- NO,                   A2303400
         CLC   0(8,WORK7),2(1)         ARE SYMBOLS IDENTICAL             A2303410
         BC    8,REPTAB                YES,BRANCH- NO, INCREMENT START   A2303420
         LA    WORK7,20(WORK7)         * ADDRESS OF SYMTAB AND EXAMINE   A2303430
         BCT   WORK6,TABTRM            * NEXT ELEMENT.                   A2303440
TABERR   MVI   TABSW,X'01'             SYMBOL NOT IN TABLE AND TABLE     A2303450
*                                       EXHAUSTED OR SVC 18 WITH UNKNOWN  A2303460
         BC    15,ERROR                SYMBOL.                           A2303470
*                                                                       A2303480
*  MAKE  A NEW ENTRY IN THE TABLE.                                      A2303490
*                                                                       A2303500
FILL     TM    IOSW,X'01'              REQUEST FOR I/O OPERATION- YES,   A2303510
         BC    1,TABERR                * ERROR (SYMBOL UNKNOWN), BRANCH  A2303520
FILLC    LA    WORK2,20(WORK2)         UPDATE CURRENT ADDR. OF SYMTAB    A2303530
         BC    15,FILLA                GO TO EXAMINE TYPES OF DEVICE     A2303540
*                                       * AND I/O OPERATION.              A2303550
REPTAB   TM    IOSW,X'01'              REQUEST FOR I/O OPERATION         A2303560
         BC    1,STRTIO                YES, BRANCH TO START OPERATION    A2303570
         CLC   10(7,1),ZEROS           IS REQUEST FOR DEVICE ...         A2303580
         BC    7,FILLD                 ...CHARACTERISTICS. NO,BRANCH     A2303590
         MVC   10(7,1),8(WORK7)        YES SEND ADDRESS AND TYPE OF...   A2303600
```

```
         BC    15,RETURN              ...DEVICE TO CALLER                A2303610
FILLD    L     WORK2,TABEND           RELOAD  CURRENT ADDR. OF SYMTAB    A2303620
*                                                                        A2303630
*  LOOK-UP  DEVICE IN DEVICE  LIST (DEVID).                              A2303640
*                                                                        A2303650
FILLA    LA    WORK4,DEVID            START ADDR. OF DEVID LIST          A2303660
FILLB    CLC   0(5,WORK4),12(I)       IS TYPE IN THIS LIST               A2303670
         BC    8,REPTB1               YES, BRANCH (FOR A LAST VERIF.)    A2303680
         LA    WORK4,8(WORK4)         NO- ADDR. OF NEXT LIST ELEMENT     A2303690
         CLI   0(WORK4),X'00'         IS TABLE EXHAUSTED                 A2303700
         BC    7,FILLB                NO, GO TO EXAMINE NEXT ELEMENT     A2303710
*                                      YES, EXAMINE DEVICE TYPE  AND      A2303720
*                                      BYTE IN LOCATION IOTYPE(I) IN      A2303730
*                                      SVC 17 CALLING SEQUENCE.           A2303740
         CLC   TP2400(4),12(I)        2400 MAGNETIC TAPE                 A2303750
         BC    7,ERRPRG               NO, INCORRECT DEV. OR OPERATION    A2303760
         TM    16(I),X'E0'            YES, IS IT A 7TRACK MODE OPERAT.   A2303770
         BC    5,ERRPRG               NO, ERROR.                         A2303780
         TM    16(I),X'0F'            YES, IS DENSITY SPECIFIED          A2303790
         BC    8,ERRPRG               NO, ERROR.                         A2303800
         LA    WORK4,TP2400           YES, SVC 17 OK.  LOAD ADDRESS      A2303810
*                                      * OF ACCORDING LIST ELEMENT.       A2303820
REPTB1   MVC   VERIFY(6),10(I)        SET-UP VERIFY SEQUENCE             A2303830
         CNOP  0,4                    DOES AN UCB EXIST FOR THE          A2303840
         SVC   0                      * DEVICE  IN THE CONTROL PROG.     A2303850
VERIFY   DC    X'0000'                * (DEVICE ADDRESS)                 A2303860
         DC    C'    '                * (DEVICE TYPE)                    A2303870
         DC    X'00'                  * (SPECIAL FEATURES- UNUSED)       A2303880
         DC    AL3(ERRCP)             NO, RETURN TO ERROR                A2303890
         MVC   0(15,WORK7),2(I)       YES, PLACE NEW ELEMENT IN SYMTAB   A2303900
         MVC   17(3,WORK7),5(WORK4)   *                                  A2303910
         ST    WORK2,TABEND           UPDATE CURRENT ADDR. OF SYMTAB     A2303920
*                                                                        A2303930
RETURN   LA    I,20(I)                I+20 - RETURN TO SVC 17 CALL       A2303940
         ST    I,CALPSW+4             RETURN ADDRESS TO SVC OLD PSW      A2303950
         LM    1,10,CALLGR            RESTORE G.R. 1-10                  A2303960
         CNOP  2,4                    *                                  A2303970
         SVC   3                      RETURN TO CALLER                   A2303980
         DC    A(CALPSW)              *                                  A2303990
*                                                                        A2304000
ERRPRG   MVI   TABSW,X'02'            PROGRAMMING ERROR- INVALID PARA-   A2304010
         BC    15,ERROR               * METERS IN SVC 17 SEQUENCE.       A2304020
ERRCP    MVI   TABSW,X'03'            NO UCB IN CONTR.PROG. FOR THE      A2304030
ERROR    TM    IOSW,X'01'             * DEVICE. IS SVC 18 CALL           A2304040
         BC    1,EXCRET               YES,BRANCH.                        A2304050
         MVC   16(1,I),TABSW          NO, SET ERROR INDICATION IN BYTE   A2304060
         L     I,16(I)                * AT LOCATION BUFF AND RETURN      A2304070
         BC    15,RETURN+4            * TO CALLER, (I)+16.               A2304080
         EJECT                                                          A2304090
*************************************************************************A2304100
*                                                                       *A2304110
*            I/O REQUEST SEQUENCE INITIALIZATION ROUTINE                *A2304120
*                                                                       *A2304130
*                 NAME= STRTIO                                          *A2304140
*                                                                       *A2304150
```

```
*  THIS ROUTINE PERFORMS CERTAIN INITIALIZATION STEPS  FOR  ALL  I/O  * A2304160
*  OPERATIONS.                                                        * A2304170
*  THIS ROUTINE HAS TWO ENTRY POINTS                                  * A2304180
*        STRTIO  ENTERED FROM THE ENTRY ROUTINE WHEN AN I/O           * A2304190
*                OPERATION IS BEING INITIALIZED.                      * A2304200
*        RETSIO  ENTERED FROM THE SPECIFIC I/O ROUTINES FOR           * A2304210
*                RETRY OPERATIONS.                                    * A2304220
*  THIS ROUTINE EXITS TO THE SPECIFIC I/O ROUTINES.                   * A2304230
*                                                                     * A2304240
********************************************************************** A2304250
         SPACE                                                         A2304260
STRTIO   XC    RETRSW(3),RETRSW       RESET ALL SWITCHES FOR I/O OPE.  A2304270
         L     LINK,MESLNK            SET UP IN CHANNEL COMMAND WORD.. A2304280
         LA    LINK,1(LINK)           * ADDRESS OF DATA (=BUFF+1)  AND A2304290
         ST    LINK,CCWPR             *                                A2304300
         MVC   CCWPR+6(2),10(I)       * BYTE COUNT                     A2304310
         MVC   CALL+2(2),8(WORK7)     PLACE IN SVC 11 SEQ. ADDRESSES   A2304320
         MVC   CALL+5(3),ACCWPR       * OF DEVICE AND CCW.             A2304330
RETSIO   L     WORK2,16(WORK7)        BRANCH TO SPECIFIC  I/O ROUTINE  A2304340
         BALR  LINK,WORK2             *                                A2304350
         SPACE                                                         A2304360
********************************************************************** A2304370
*                                                                    * A2304380
*                      I/O REQUEST EXITS ROUTINE                     * A2304390
*                                                                    * A2304400
*                    NAME= NRMRET                                    * A2304410
*                                                                    * A2304420
*  THIS ROUTINE IS ENTERED FROM THE SPECIFIC I/O ROUTINES AT THE LO- * A2304430
*  CATIONS SPECIFIED BY SYMBOLIC NAMES (NRMRET OR EXCRET) AND  EXITS * A2304440
*  TO LOCATION  RETURN+4  IN ENTRY ROUTINE.                          * A2304450
*  THIS ROUTINE SETS THE BYTE AT ADDRESS  BUFF, IN CALLING PROGRAM ,  * A2304460
*  TO SOME NON-ZERO VALUE. SEE ABOVE, PROGRAM FUNCTIONS.             * A2304470
*                                                                    * A2304480
********************************************************************** A2304490
         SPACE                                                         A2304500
NRMRET   MVI   TABSW,X'07'           NORMAL RETURN- I/O OPERAT. OK    A2304510
EXCRET   L     LINK,MESLNK           EXCEPTIONAL RETURN- SET BYTE IN  A2304520
         MVC   0(1,LINK),TABSW        * LOCATION BUFF  ACCORDINGLY.   A2304530
         LA    I,16(I)                RETURN ADDRESS                   A2304540
         MVI   IOSW,X'00'             RESET INPUT/OUTPUT SWITCH        A2304550
         BC    15,RETURN+4            *                                A2304560
         EJECT                                                         A2304570
********************************************************************** A2304580
*                                                                    * A2304590
*                       I/O REQUEST ROUTINE                          * A2304600
*                                                                    * A2304610
*                    NAME= CALLA                                     * A2304620
*                                                                    * A2304630
*  THIS  ROUTINE INITIATES THE I/O OPERATIONS BY ISSUYING AN I/O RE- * A2304640
*  QUEST AND WAIT CALL (SVC 11). THE ROUTINE THEN ANALYZES THE  STA- * A2304650
*  TUS SET BY THE CONTROL PROGRAM AND PERFORMS  A  SERIES OF CHANNEL * A2304660
*  STATUS AND SENSE BYTES TESTS.                                     * A2304670
*  THE ROUTINE IS ENTERED FROM THE SPECIFIC  I/O ROUTINES  AND EXITS * A2304680
*  TO VARIOUS LOCATIONS  ACCORDING TO THE CONDITIONS FOUND.          * A2304690
*                                                                    * A2304700
```

```
*********************************************************************** A2304710
        SPACE                                                           A2304720
CALLA   STM   1,10,TEMPGR           SAVE I/O PACKAGE REGISTERS          A2304730
        LM    1,10,CALLGR           RESTORE CALLING REGISTERS           A2304740
        SVC   9                     ENABLE I/O AND EXTERN.INTERRUPTS    A2304750
        CNOP  4,8                   *                                   A2304760
CALL    SVC   11                    I/O REQUEST AND WAIT                A2304770
        DC    X'0000'               * DEVICE ADDRESS                    A2304780
        DC    A(0)                  * CCW  ADDRESS                      A2304790
ERRPR   DC    A(0)                  * ERROR TYPE AND SENSE BYTES (3)    A2304800
CSWPR   DS    D                     * CSW                               A2304810
        BC    0,0                   NOP.                                A2304820
        SVC   8                     DISABLE I/O AND EXT. INTERRUPTS     A2304830
        ST    IOBASE,0              SAVE CONTENTS OF G.R. IOBASE        A2304840
        BALR  IOBASE,0              RESTORE I/O SUPPORT PACKAGE..       A2304850
SQCALL  STM   1,10,CALLGR-SQCALL(IOBASE) ..REGISTERS AND SAVE...        A2304860
        LM    1,10,TEMPGR-SQCALL(IOBASE) ..CALLING REGISTERS            A2304870
        MVC   CALLGR+24(4),0        REPLACE IOBASE                      A2304880
        TM    CSWPR+4,BUSY          IS BUSY CONDITION PRESENT           A2304890
        BC    1,CALLA               YES- CYCLE ON SVC 11 CALL           A2304900
        TM    ERRPR,X'02'           NO, IS ERRTYP= 02 OR 03             A2304910
        BC    1,PGRACT              YES-CALL DUMP                       A2304920
        CLI   ERRPR,X'11'           NO, IS DEVICE NOT OPERATIONAL       A2304930
        BC    8,SRPACT              YES- CALL SEREP                     A2304940
        NC    INVDEV(1),ERRPR+1     NO, IS ANY INVALID SENSE BITS       A2304950
        BC    4,SRPACT              YES- CALL SEREP                     A2304960
        NC    INVCHN(1),CSWPR+5     NO, IS ANY INVALID CHANNEL BITS     A2304970
        BC    4,SRPACT              YES-CALL SEREP                      A2304980
        BCR   15,LINK               NO, NORMAL RETURN TO CALLER         A2304990
*                                                                       A2305000
CCWPR   CCW   X'00',*,X'00',1       CCW USED TO PERFORM I/O OPERAT.     A2305010
*                                   * (NO CHAINING, SLI OFF)            A2305020
ACCWPR  DC    AL3(CCWPR)            *                                   A2305030
        EJECT                                                           A2305040
*********************************************************************** A2305050
*                                                                     * A2305060
*           CALL WRITE ROUTINE FOR I/O PACKAGE MESSAGES               * A2305070
*                                                                     * A2305080
*  THIS ROUTINE IS USED TO SEND  THE  MESSAGES PECULIAR  TO THE  I/O  * A2305090
*  SUPPORT PACKAGE PROGRAM.                                           * A2305100
*  THE ROUTINE HAS FOUR ENTRY POINTS=                                 * A2305110
*        INFACT   USED TO SEND MESSAGE 'END OF FILE'.  CONTROL  WILL  * A2305120
*                 BE RETURNED TO THE CALLING PROGRAM.                 * A2305130
*        PGRACT   USED TO SEND MESSAGE 'PROGRAM ERROR'. AFTER  PRIN-  * A2305140
*                 THE WAIT STATE IS ENTERED.                          * A2305150
*        SRPACT   USED TO SEND MESSAGE 'LOAD SEREP PROGRAM'.   AFTER  * A2305160
*                 PRINTING  A 'SET-UP SEREP INTERFACE' REQUEST    IS  * A2305170
*                 SUBMITTED TO THE CONTROL PROGRAM.                   * A2305180
*        HEADMS   USED TO SEND ANY I/O PACKAGE MESSAGE.               * A2305190
*  THE MESSAGES ARE PRINTED IN TWO OR THREE TIMES=                    * A2305200
*  1. MESSAGE CODE, SEQUENTIAL NUMBER, DEVICE ADDRESS.                * A2305210
*  2. TEXT OF MESSAGE.                                                * A2305220
*  3. COMMENT, IF NEED BE.                                            * A2305230
*                                                                     * A2305240
*********************************************************************** A2305250
```

```
INFACT   LA    WORK2,UEMES          PREPARE MESSAGE 'END OF FILE',   A2305260
         MVI   MESACT,X'00'         * NO OPERATOR ANSWER REQUESTED.  A2305270
         BAL   LNKMES,HEADMS        GO TO CALL WRITE SEQUENCE.       A2305280
INFCT1   MVI   TABSW,X'03'          WRITING COMPLETED. SET 'EOF-EOT' A2305290
         BC    15,EXCRET            * INDICATION IN BYTE AT LOCATION A2305300
*                                   * BUFF AND RETURN TO CALLER.     A2305310
PGRACT   LA    WORK2,PGRMES         PREPARE MESSAGE 'PROGRAM ERROR', A2305320
         MVI   MESACT,X'00'         * NO OPERATOR ANSWER  REQUESTED. A2305330
         BAL   LNKMES,HEADMS        GO TO CALL WRITE SEQUENCE.       A2305340
         BC    15,WTCMDR            WRITING COMPLETED. GO TO ENTER   A2305350
*                                   * THE WAIT STATE.                A2305360
SRPACT   LA    WORK2,CATMES         PREPARE MESSAGE 'LOAD SEREP',    A2305370
         MVI   MESACT,X'00'         * NO OPERATOR ANSWER REQUESTED.  A2305380
         BAL   LNKMES,HEADMS        GO TO CALL WRITE SEQUENCE        A2305390
         CNOP  2,4                  *                                A2305400
         SVC   7                    SET-UP SEREP INTERFACE           A2305410
SEREP    DC    X'3F'                * TYPE OF I/O FAILURE            A2305420
         DC    AL3(CALL)            * ADDR. OF SVC 11.               A2305430
*                                                                     A2305440
HEADMS   MVC   MESID(4),0(WORK2)     SET UP SEQUENCE TO WRITE  TEXT   A2305450
*                                   * PART OF MESSAGE.               A2305460
         MVC   NUMBER(3),4(WORK2)   SET SEQUENCE NUMBER OF MESSAGE   A2305470
         CNOP  0,4                  * IN CODE PART.                  A2305480
         BAL   LINK,CONSLE          CALL ROUTINE TO WRITE MESSAGE    A2305490
         DC    FL1'10'              * CODE AND DEVICE ADDRESS.       A2305500
         DC    AL3(HEADER)          *                                A2305510
         BAL   LINK,CNSL            CALL ROUTINE TO WRITE TEXT OF    A2305520
MESID    DC    FL1'0'               * MESSAGE.                       A2305530
         DC    AL3(0)               *                                A2305540
         BAL   LINK,CNSL            CALL ROUTINE TO WRITE TEXT SPE-  A2305550
MESACT   DC    FL1'0'               * CIFYING ACTION TO BE TAKEN, IF A2305560
         DC    AL3(0)               * NEED BE.                       A2305570
         BAL   LINK,CNSL            END OF WRITING. RETURN TO CALLER A2305580
         DC    X'00'                *                                A2305590
         EJECT                                                       A2305600
*********************************************************************A2305610
*                                                                   * A2305620
*           WRITE I/O SUPPORT PACKAGE MESSAGES ROUTINE              * A2305630
*                                                                   * A2305640
*                    NAME= CONSLE                                   * A2305650
*                                                                   * A2305660
*  THIS ROUTINE WRITES MESSAGES -FROM I/O PACKAGE PROGRAM OR CALLING * A2305670
*  PROGRAM- ON THE 1052 PRINTER KEYBOARD. IT INITIATES THE OPERATION * A2305680
*  BY ISSUYING AN 'SVC 4' CALL. IT, THEN, ANALYZES THE STATUS SET BY * A2305690
*  THE CONTROL PROGRAM  AND RETURNS CONTROL TO THE I/O PACKAGE  CAL- * A2305700
*  LING ROUTINE.                                                    * A2305710
*  THIS ROUTINE IS ENTERED AT LOCATION.                             * A2305720
*        CONSLE    TO WRITE MESSAGE CODE AND SEQUENTIAL NUMBER,  AND * A2305730
*                  DEVICE ADDRESS.                                  * A2305740
*        CNSL      TO WRITE MESSAGE TEXT AND COMMENT (IF NEED BE).  * A2305750
*                  THESE TWO ENTRIES ARE ONLY USED  FOR  I/O PACKAGE * A2305760
*                  MESSAGES.                                        * A2305770
*        CNSLB     TO WRITE THE CALLING PROGRAM   MESSAGES.         * A2305780
*                                                                   * A2305790
*********************************************************************A2305800
```

```
         SPACE                                                          A2305810
CONSLE   LH    X,8(WORK7)              CONVERT DEVICE ADDRESS TO HEXA-  A2305820
         LA    BYTE,DVADDR             * DECIMAL AND STORE RESULT IN    A2305830
         BAL   WORK6,CVRTH             * HEADER MESSAGE.                A2305840
         DC    H'3'                    * (NUMBER OF BYTES).             A2305850
CNSL     CLI   0(LINK),X'00'           IS MESSAGE FULLY WRITTEN         A2305860
         BCR   8,LNKMES                YES-RETURN TO CALLER             A2305870
CNSLB    MVC   MESSCT(4),0(LINK)       NO, WRITE NEXT PART              A2305880
         CNOP  2,4                     *                                A2305890
         SVC   4                       WRITE MESSAGE                    A2305900
MESSCT   DC    X'00'                   * BYTE COUNT                     A2305910
         DC    AL3(0)                  * MESSAGE ADDRESS                A2305920
         L     BYTE,MESSCT             *                                A2305930
         TM    0(BYTE),X'07'           IS MESSAGE WRITTEN               A2305940
         BC    5,4(LINK)               YES-RETURN TO CALLER             A2305950
*                                       NO, WAIT END OF WRITING          A2305960
CONSLA   STM   1,10,TEMPGR             SAVE G.R. 1-10                   A2305970
         LM    1,10,CALLGR             RESTORE CALLING REGISTERS        A2305980
         SVC   19                      WAIT FOR END OF WRITE OPERATION  A2305990
CSLRET   SVC   8                       DISABLE I/O,EXT. INTERRUPTS      A2306000
         ST    IOBASE,0                SAVE CONTENTS OF G.R. IOBASE     A2306010
         BALR  IOBASE,0                * SAVE CALLING G.R. 1-10  AND    A2306020
CONSLB   STM   1,10,CALLGR-CONSLB(IOBASE) * RESTORE I/O PACKAGE         A2306030
         LM    1,10,TEMPGR-CONSLB(IOBASE) * G.R. 1-10                   A2306040
         MVC   CALLGR+24(4),0          *                                A2306050
         BC    15,MESSCT+4             GO AND SEE IF MESSAGE IS WRITTEN A2306060
         EJECT                                                          A2306070
*********************************************************************** A2306080
*                                                                     * A2306090
*                  INPUT COMMANDS PROCESSING ROUTINE                  * A2306100
*                                                                     * A2306110
*                    NAME= SWEXT                                      * A2306120
*                                                                     * A2306130
*  THIS ROUTINE IS ENTERED, FROM THE SPECIFIC I/O ROUTINES, WHEN THE  * A2306140
*  OPERATOR MUST TYPE ONE COMMAND ON THE  1052 PRINTER-KEYBOARD   TO  * A2306150
*  RESTART AN I/O OPERATION OR TO INDICATE THAT  THE  DEVICE  CAN NO  * A2306160
*  LONGER BE USED.                                                    * A2306170
*  THE ROUTINE LINKS TO THE READ COMMAND ROUTINE (TYPRD) AND  ANALY-  * A2306180
*  ZES THE COMMAND READ. IT EXITS TO  A  SPECIFIC I/O ROUTINE   (RE-  * A2306190
*  START) OR THE CALLING PROGRAM (STOP).                              * A2306200
*                                                                     * A2306210
*********************************************************************** A2306220
         SPACE                                                          A2306230
SWEXT    LA    I,MESOP-10              SIMULATE SVC 18 CALL TO READ     A2306240
         BAL   LINK,TYPRD              * OPERATOR'S COMMAND FROM 1052.  A2306250
         L     I,CALLSQ                RESTORE I (ADDR. OF USER'S CALL) A2306260
         OC    BUFMES+1(5),BLANK       REPLACE LOWER BY UPPER CASE      A2306270
         CLC   BUFMES+1(4),START       IS START ANSWER                  A2306280
         BCR   8,LNKMES                YES, GO TO RETRY I/O OPERATION   A2306290
         CLC   BUFMES+1(4),STOP        IS STOP ANSWER                   A2306300
         BC    8,RETCAL                YES, BRANCH                      A2306310
         CNOP  0,4                     NO, ANSWER NOT RECOGNIZED. CALL  A2306320
         BAL   LINK,CNSLB              * WRITE ROUTINE TO REQUEST A     A2306330
SWEXT1   DC    FL1'19'                 * NEW ANSWER                     A2306340
         DC    AL3(ASSIGN)             *                                A2306350
```

```
         BC    15,SNEXT                GO TO ANALYZE IT.                A2306360
RETCAL   MVI   TABSW,X'02'            RETURN TO CALLER WITH ERROR      A2306370
         XC    RETRSW(3),RETRSW        * INDICATION '02' = DEVICE       A2306380
         BC    15,EXCRET               * UNSERVICEABLE (OPERATOR        A2306390
*                                      * DECISION).                     A2306400
         SPACE 2                                                        A2306410
*********************************************************************** A2306420
*                                                                     * A2306430
*              BINARY TO HEXADECIMAL CONVERSION ROUTINE               * A2306440
*                                                                     * A2306450
*             (X)    = BINARY DATA TO BE CONVERTED (RIGHT JUSTIFIED)  * A2306460
*             (BYTE) = A(DESTINATION FIELD)                           * A2306470
*             (WORK6)= A(BYTE COUNT)                                  * A2306480
*                                                                     * A2306490
*********************************************************************** A2306500
         SPACE                                                          A2306510
CVRTM    LH    BYTCNT,0(WORK6)         * NUMBER OF 4-BIT DIGITS         A2306520
         AR    BYTE,BYTCNT             *                                A2306530
CNVRTA   LR    NXTBYT,X                NEXT HEX CHAR. TO NXTBYT         A2306540
         BCTR  BYTE,0                  *                                A2306550
         SRL   X,4                     POSITION X FOR NEXT CHAR.        A2306560
         N     NXTBYT,B28T31           CONVERT NXTBYT                   A2306570
         IC    NXTBYT,HEXTAB(NXTBYT)   *                                A2306580
         STC   NXTBYT,0(BYTE)          PLACE IN MESSAGE                 A2306590
         BCT   BYTCNT,CNVRTA           BYTCNT-1 TO BYTCNT               A2306600
         BC    15,2(WORK6)             RETURN TO CALLER                 A2306610
         EJECT                                                          A2306620
*********************************************************************** A2306630
*                                                                     * A2306640
*            MESSAGES ISSUED FROM I/O PACKAGE PROGRAM                 * A2306650
*                                                                     * A2306660
*  THE FOLLOWING BLOCK CONTAINS THE MESSAGES WHICH  THE I/O  SUPPORT  * A2306670
*  PACKAGE MAY PRINT AT ANY TIME IT USES A SYSTEM/360 DEVICE TO PER-  * A2306680
*  FORM AN I/O OPERATION.                                             * A2306690
*                                                                     * A2306700
*  THE SPECIFICATION OF A PARTICULAR MESSAGE  FORMS  A SET GROUP  ON  * A2306710
*  THE MODEL OF MESSAGE 20. THESE GROUPS ARE COLLECTED IN ONE BLOCK.  * A2306720
*  THE TWO LETTERS IDENTIFYING THE SIMULATOR SYSTEM  ARE  AT THE BE-  * A2306730
*  GINNING OF THE BLOCK AND APPLY TO ALL THE MESSAGES.                * A2306740
*  THE COMMENTS FORM ONE GROUP AT THE END OF THE MESSAGE BLOCK.       * A2306750
*  THE BUFFER USED TO STORE OPERATOR'S COMMAND AND THE COMMANDS  RE-  * A2306760
*  COGNIZED FORM A LAST GROUP.                                        * A2306770
*                                                                     * A2306780
*********************************************************************** A2306790
         SPACE                                                          A2306800
HEADER   DC    C' A2'                  MESSAGE CODE (COMPONENT,PHASE)   A2306810
NUMBER   DC    C'    '                    -    SEQUENTIAL NUMBER       A2306820
DVADDR   DC    C'    '                 DEVICE ADDRESS (HEXADECIMAL)     A2306830
*                                                                     * A2306840
PGRMES   DC    FL1'24'                 MESSAGE 20- TEXT LENGTH          A2306850
         DC    AL3(PGRCHK)             *           TEXT ADDRESS         A2306860
         DC    C'20W'                  *           SEQUENTIAL NUMBER    A2306870
PGRCHK   DC    C' PROGRAM ERROR,'      *           TEXT                 A2306880
         DC    C'LOAD DUMP'            *                                A2306890
         DC    X'15'                   *                                A2306900
```

```
CATMES   DC    FL1'20'              MESSAGE 21.                          A2306910
         DC    AL3(CATAS)           *                                    A2306920
         DC    C'21W'               *                                    A2306930
CATAS    DC    C' UNRECOVERABLE E'  *                                    A2306940
         DC    C'RROR'              *                                    A2306950
         DC    X'15'                *                                    A2306960
UEMES    DC    FL1'12'              MESSAGE 22.                          A2306970
         DC    AL3(UEXCP)           *                                    A2306980
         DC    C'22I'               *                                    A2306990
UEXCP    DC    C' END OF FILE'      *                                    A2307000
         DC    X'15'                *                                    A2307010
EOTMES   DC    FL1'12'              MESSAGE 23.                          A2307020
         DC    AL3(ENDTPE)          *                                    A2307030
         DC    C'23I'               *                                    A2307040
ENDTPE   DC    C' END OF TAPE'      *                                    A2307050
         DC    X'15'                *                                    A2307060
INTMES   DC    FL1'22'              MESSAGE 24.                          A2307070
         DC    AL3(INTRQ)           *                                    A2307080
         DC    C'24D'               *                                    A2307090
INTRQ    DC    C' INTERVENTION '    *                                    A2307100
         DC    C'REQUIRED'          *                                    A2307110
         DC    X'15'                *                                    A2307120
DATMES   DC    FL1'11'              MESSAGE 25.                          A2307130
         DC    AL3(DATC)            *                                    A2307140
         DC    C'25D'               *                                    A2307150
DATC     DC    C' DATA CHECK'       *                                    A2307160
         DC    X'15'                *                                    A2307170
EQCMES   DC    FL1'16'              MESSAGE 26.                          A2307180
         DC    AL3(EQCHCK)          *                                    A2307190
         DC    C'26D'               *                                    A2307200
EQCHCK   DC    C' EQUIPMENT CHECK'  *                                    A2307210
         DC    X'15'                *                                    A2307220
BOCMES   DC    FL1'14'              MESSAGE 27.                          A2307230
         DC    AL3(BOCHCK)          *                                    A2307240
         DC    C'27D'               *                                    A2307250
BOCHCK   DC    C' BUS OUT CHECK'    *                                    A2307260
         DC    X'15'                *                                    A2307270
OVRMES   DC    FL1'8'               MESSAGE 28.                          A2307280
         DC    AL3(OVRUN)           *                                    A2307290
         DC    C'28D'               *                                    A2307300
OVRUN    DC    C' OVERRUN'          *                                    A2307310
         DC    X'15'                *                                    A2307320
CHNMES   DC    FL1'15'              MESSAGE 29.                          A2307330
         DC    AL3(CHNCHK)          *                                    A2307340
         DC    C'29D'               *                                    A2307350
CHNCHK   DC    C' CHAINING CHECK'   *                                    A2307360
         DC    X'15'                *                                    A2307370
*                                                                        A2307380
OPACT1   DC    FL1'19'              COMMENT 1. SPECIFY COMMANDS (SEE     A2307390
         DC    AL3(ASSIGN)          * BELOW).                            A2307400
ACTMES   DC    FL1'60'              COMMENT 2. SPECIFY ACTION TO BE      A2307410
         DC    AL3(ACTOP)           * TAKEN AND COMMANDS TO TYPE TO      A2307420
ACTOP    DC    C' UNLOAD HOPPER,'   * RETRY I/O OPERATION OR TO DE-      A2307430
         DC    C'RELOAD LAST '      * CLARE THE DEVICE UNSERVICEABLE     A2307440
CRDNUM   DC    C'  CARD(S) AND'     *                                    A2307450
```

```
ASSIGN   DC    C' TYPE '              *                                A2307460
         DC    C'START OR STOP'       *                                A2307470
         DC    X'15'                  *                                A2307480
*                                                                       A2307490
BUFMES   DS    6C                     INPUT BUFFER (FOR OPERATOR       A2307500
         CNOP  2,4                    * COMMANDS)                      A2307510
MESOP    DC    H'5'                   * LENGTH AND ADDRESS OF          A2307520
         DC    A(BUFMES)              * BUFFER.                        A2307530
START    DC    C'STAR'                START COMMAND (RETRY I/O OPER.)  A2307540
STOP     DC    C'STOP'                STOP    -    (DV.UNSERVICEABLE)  A2307550
         EJECT                                                         A2307560
*********************************************************************** A2307570
*                                                                     * A2307580
*                  WRITE USER'S MESSAGES ROUTINE                      * A2307590
*                                                                     * A2307600
*                  NAME= TYPWRT                                       * A2307610
*                                                                     * A2307620
*  THIS ROUTINE IS ENTERED FROM THE I/O REQUEST INITIALIZATION  ROU-  * A2307630
*  TINE TO SEND USER'S MESSAGES ON THE 1052 PRINTER-KEYBOARD.    THE  * A2307640
*  OPERATIONS ARE AS FOLLOW=                                          * A2307650
*  1. THE ROUTINE PERFORMS CERTAIN INITIALIZING STEPS.                * A2307660
*  2. THE ROUTINE BRANCHES TO THE  WRITE I/O PACKAGE MESSAGES ROUTI-  * A2307670
*     NE  WHICH PERFORMS THE WRITE OPERATION AND RETURNS CONTROL  TO  * A2307680
*     THE ROUTINE.                                                    * A2307690
*  3. THE ROUTINE THEN ANALYZES  THE STATUS  SET BY THE CONTROL PRO-  * A2307700
*  GRAM AND EXITS TO THE I/O REQUEST EXITS ROUTINE (MESSAGE WRITTEN)  * A2307710
*  OR TO AN ERROR WAIT.                                               * A2307720
*                                                                     * A2307730
*********************************************************************** A2307740
         SPACE                                                         A2307750
TYPWRT   MVC   CNSWRT+4(1),11(I)      SET UP NEXT CALLING SEQUENCE     A2307760
         MVC   CNSWRT+5(3),13(I)      *                                A2307770
         CNOP  0,4                    *                                A2307780
CNSWRT   BAL   LINK,CNSLB             CALL WRITE MESSAGE ROUTINE       A2307790
         DC    X'00'                  * MESSAGE LENGTH                 A2307800
         DC    AL3(0)                 * BUFFER  ADDRESS                A2307810
         L     LINK,0(LINK)           RETURN. DOES THE 1ST  BYTE OF    A2307820
         TM    0(LINK),X'07'          * BUFFER CONTAIN '07'            A2307830
         BC    12,PGRACT              NO,PROGRAMMING ERROR, BRANCH     A2307840
         BC    15,NRMRET              YES, WRITING OK,RETURN TO CALLER A2307850
         SPACE 2                                                       A2307860
*********************************************************************** A2307870
*                                                                     * A2307880
*                  READ  USER'S COMMANDS ROUTINE                      * A2307890
*                                                                     * A2307900
*                  NAME= TYPRD                                        * A2307910
*                                                                     * A2307920
*  THIS ROUTINE IS ENTERED FROM THE I/O REQUEST INITIALIZATION  ROU-  * A2307930
*  TINE TO READ THE 1052 INPUT COMMAND AIMED TO THE CALLING PROGRAM.  * A2307940
*  THE OPERATIONS ARE AS FOLLOW=                                      * A2307950
*  1. THE ROUTINE SAVES CERTAIN PARAMETERS OF CONTROL PROGRAM.        * A2307960
*  2. THE ROUTINE SET NEW PARAMETERS  (SVC 5).                        * A2307970
*  3. THE ROUTINE SIMULATES AN ATTENTION INTERRUPT BY BRANCHING   TO  * A2307980
*     THE READ INPUT COMMANDS ROUTINE IN CONTROL PROGRAM.             * A2307990
*  4. THE ROUTINE RESTORE INITIAL PARAMETERS  WHEN  READING  IS COM-  * A2308000
```

```
*     PLETED.                                                          * A2308010
*  5. THE ROUTINE EXITS TO THE I/O REQUEST EXITS ROUTINE.              * A2308020
*                                                                      * A2308030
************************************************************************ A2308040
        SPACE                                                           A2308050
*                                                                       A2308060
* SAVE PARAMETERS SET BY THE USER'S PROGRAM (BY MEANS OF SVC 5 CALL).   A2308070
*                                                                       A2308080
TYPRD   ST    LINK,LINKCS             SAVE I/O PACKAGE G.R. LINK        A2308090
        L     INTCDE,CSLID            ADDR. OF ENTRY POINT TO CTRL.PR.  A2308100
        L     WORK2,0(INTCDE)         SAVE READ COMMAND CCW AND ADDR.   A2308110
        MVC   SVCSL(20),0(WORK2)      * OF COMMAND BUFFER, LENGTH AND   A2308120
*                                     * RETURN.                         A2308130
        L     WORK2,4(INTCDE)         SAVE ADDRESS OF AN ENTRY POINT    A2308140
        L     WORK2,0(WORK2)          * TO CONTROL PROG. WHERE CONTROL  A2308150
        MVC   CSLATT(4),DEVATT(WORK2) * IS RETURNED WHEN ATTENTION      A2308160
*                                     * OCCURS.                         A2308170
* SET PARAMETERS TO READ COMMANDS AIMED AT I/O SUPPORT PACKAGE          A2308180
*                                                                       A2308190
        MVC   CSLRD+2(1),11(I)        SET UP SVC 5 SEQUENCE. LENGTH     A2308200
        MVC   CSLRD+3(3),13(I)        * AND ADDR. OF INPUT BUFFER       A2308210
        CNOP  6,8                     * (LOCATED IN USER'S PROGRAM)     A2308220
CSLRD   SVC   5                       *                                 A2308230
        DC    X'00'                   *                                 A2308240
        DC    AL3(0)                  *                                 A2308250
        DC    X'00'                   *                                 A2308260
        DC    AL3(COMRET)             *                                 A2308270
        DS    D                       *                                 A2308280
        L     WORK6,DEVATT(WORK2)     SIMULATE ATTENTION INTERRUPT      A2308290
        MVC   0(8,WORK6),RETPSW       * PSW AT POINT OF INTERRUPT.      A2308300
        AH    WORK6,DEC8              * ADDRESS OF ENTRY POINT TO READ  A2308310
        ST    WORK6,CMDPSW+4          * ROUTINE (IN CONTROL PROGR.)     A2308320
        STM   1,10,TEMPGR             SAVE I/O PACK. G.R.               A2308330
        LM    1,10,CALLGR             RESTORE CALLER G.R.               A2308340
        CNOP  2,4                     *                                 A2308350
        SVC   3                       BRANCH TO READ ROUTINE.           A2308360
        DC    A(CMDPSW)               *                                 A2308370
WTCMDR  SVC   19                      WAIT FOR END OF READ OPERATION.   A2308380
        SVC   3                       *                                 A2308390
        DC    A(RETPSW)               * (BRANCH TO WTCMDR).             A2308400
*                                                                       A2308410
* READING COMPLETED. RESTORE USER'S PARAMETERS IN CONTROL PROGRAM.      A2308420
*                                                                       A2308430
COMRET  ST    IOBASE,0                SAVE CALLER G.R. IOBASE           A2308440
        BALR  IOBASE,0                *                                 A2308450
CMRET1  STM   1,10,CALLGR-CMRET1(IOBASE) * SAVE CALLER G.R. 1-10        A2308460
        LM    1,10,TEMPGR-CMRET1(IOBASE) * RESTORE I/O PACK. G.R. 1-10  A2308470
        MVC   CALLGR+24(4),0          * REPLACE CALLER G.R. IOBASE      A2308480
CMRET2  MVC   DEVATT(4,WORK2),CSLATT  RESTORE PARAMETERS IN CONTR.PRG.  A2308490
        L     INTCDE,CSLID            * ENTRY POINT TO CTRL. PRG. FOR   A2308500
        L     WORK2,0(INTCDE)         * ATTENTION INTERRUPTS.           A2308510
        MVC   0(20,WORK2),SVCSL       * READ COMMAND CCW AND ADDR. OF   A2308520
*                                     * COMMAND BUFFER, LENGTH, RETURN  A2308530
        L     BYTE,CSLRD+2            ADDR. OF INPUT BUFFER (1ST BYTE)  A2308540
        TM    0(BYTE),X'07'           WAS READING O.K.                  A2308550
```

```
         BC    12,PGRACT            NO, PROGRAMMING ERROR, BRANCH    A2308560
         L     LINK,LINKCS          YES, RESTORE I/O PACK. G.R. LINK A2308570
         BCR   15,LINK              * AND RETURN TO EXIT  ROUTINE    A2308580
         EJECT                                                       A2308590
*************************************************************************A2308600
*                                                                     * A2308610
*                        SPECIFIC I/O ROUTINES                        * A2308620
*  THESE ROUTINES ARE IDENTICAL IN THEIR  LOGICAL  STRUCTURE.  THEIR  * A2308630
*  OPERATION  ARE AS FOLLOWS=                                         * A2308640
*  1. THEY PERFORM CERTAIN UNIQUE INITIALYZING  STEPS  FOR  THE  I/O  * A2308650
*     OPERATIONS AND DIFFER ONLY  IN  THE CONSTANTS USED  FOR  THOSE  * A2308660
*     STEPS. (COMMAND CODE, INVALID CHANNEL STATUS AND SENSE BITS).   * A2308670
*  2. THEY BRANCH TO THE I/O REQUEST ROUTINE (CALLA) WHICH  PERFORMS  * A2308680
*     ITS OPERATION AND RETURNS CONTROL TO THE ROUTINES AT OPERATION  * A2308690
*     3.                                                              * A2308700
*  3. THEY THEN PERFORM A SERIES  OF  DEVICE STATUS AND  SENSE BYTES  * A2308710
*     TESTS. IF AN ERROR CONDITION IS FOUND THE ACTION PRESCRIBED BY  * A2308720
*     THE SYSTEMS STANDARDS IS TAKEN= RETRIES, MESSAGE, OPERATOR AC-  * A2308730
*     TION.                                                           * A2308740
*                                                                     * A2308750
*  THESE ROUTINES ARE ENTERED FROM  THE  I/O REQUEST  INITIALIZATION  * A2308760
*  ROUTINE AND EXIT TO THE I/O REQUEST EXITS ROUTINE.                 * A2308770
*                                                                     * A2308780
*************************************************************************A2308790
         SPACE 5                                                     A2308800
*************************************************************************A2308810
*                                                                     * A2308820
*                        CARD READING ROUTINE                         * A2308830
*                                                                     * A2308840
*                    NAME= RDTTTT     (TTTT=1442,2540)               * A2308850
*                                                                     * A2308860
*************************************************************************A2308870
         SPACE                                                       A2308880
RD1442   MVI   SW1442,X'01'          SET SWITCH '1442' ON             A2308890
         MVI   INVDEV,X'83'          SET UP INVALID SENSE BITS        A2308900
         MVI   INVCHN,X'00'          SET UP INVALID CHANNEL BITS      A2308910
         BC    15,RDCAL1             *                                A2308920
RD2540   MVI   SW1442,X'00'          SET SWITCH '1442' OFF            A2308930
         MVI   INVDEV,X'87'          SET UP INVALID SENSE BITS        A2308940
         MVI   INVCHN,X'01'          SET UP INVALID CHANNEL BITS      A2308950
RDCAL1   MVI   CCWPR,X'02'           READ COMMAND IN CCW              A2308960
RDCALL   BAL   LINK,CALLA            SUBMIT  I/O REQUEST AND WAIT     A2308970
*                                    RETURN HERE AT COMPLETION...     A2308980
*                                    ...OF OPERATION-RESTORE (LINK)   A2308990
RDCALA   TM    CSWPR+4,UC            IS UNIT CHECK PRESENT            A2309000
         BC    1,RDCALE              YES- LOOK AT SENSE BITS          A2309010
         TM    CSWPR+4,UE            NO-IS UNIT EXCEPTION             A2309020
         BC    1,INFACT              YES, END OF FILE; LAST CARD READ A2309030
         TM    CSWPR+5,CHN           NO-IS CHAINING CHECK             A2309040
         BC    8,NRMRET              NO RETURN TO CALLER              A2309050
         LA    WORK2,CHNMES          PREPARE MESSAGE FOR OPERATOR     A2309060
         BC    15,TSTCRD             TEST COUNT OF CARDS TO RELOAD    A2309070
*                                                                     A2309080
* UNIT CHECK CONDITION PRESENT. ANALYZE SENSE BYTES                   A2309090
*                                                                     A2309100
```

```
RDCALE   TM    SNSPR,INTREQ           IS INT. REQUIRED
         BC    8,RDCALM               NO-LOOK FOR BUS OUT
         LA    WORK2,INTMES           YES, PREPARE MESSAGE 'INTERVEN-
         MVC   MESACT(4),OPACT1       * TION REQUIRED'.
         BC    15,RDEND               *
*
RDCALM   TM    SNSPR,BUSOUT           IS BUS OUT CHECK PRESENT
         BC    8,RDCALH               NO-LOOK FOR EQUIPMENT CHECK
         TM    INTVSW,X'02'           YES, IS RETRY OPERATION
         BC    1,RDCALI               YES, BRANCH.
         OI    INTVSW,X'02'           NO, SET 'RETRY' SWITCH ON AND
         BC    15,STRTIO+4            * RETRY I/O OPERATION.
RDCALI   LA    WORK2,BOCMES           PREPARE MESSAGE 'BUS OUT CHECK'
         BC    15,TSTCRD              *
*
RDCALH   TM    SNSPR,EQUCHK           IS EQUIPMENT CHECK
         BC    8,RDCALJ               NO, BRANCH
         LA    WORK2,EQCMES           YES, PREPARE MESSAGE  'EQUIPMENT
         BC    15,TSTCRD              * CHECK'.
*
RDCALJ   TM    SNSPR,DATCH            IS  DATA CHECK
         BC    8,RDCALK               NO, OVERRUN (1442)
         LA    WORK2,DATMES           YES, PREPARE MESS. 'DATA CHECK'
         BC    15,TSTCRD              *
*
RDCALK   LA    WORK2,OVRMES           PREPARE MESSAGE 'OVERRUN'
*
* DETERMINE THE NUMBER OF CARDS TO BE RELOADED BEFORE GOING ON.
*
TSTCRD   CLC   CSWPR+1(3),ACCWPR      UNIT CHECK ON INITIAL SELECTION
         BC    4,CRDINT               YES, BRANCH.
         TM    SW1442,X'01'           NO, IS DEVICE A 1442 READER
         BC    8,CARD4                NO, 2540 READER, BRANCH
         MVI   CRDNUM,X'F2'           RELOAD LAST 2 CARDS
         BC    15,WRTMES              *
CARD4    MVI   CRDNUM,X'F4'           RELOAD LAST 4 CARDS
         BC    15,WRTMES              *
CRDINT   TM    SW1442,X'01'           IS DEVICE A 1442 CARD READER
         BC    1,CARD1                YES, BRANCH
         MVI   CRDNUM,X'F3'           NO, 2540 READER, RELOAD 3 CARDS
         BC    15,WRTMES              *
CARD1    MVI   CRDNUM,X'F1'           RELOAD 1 CARD
WRTMES   MVC   MESACT(4),ACTMES       SET UP CALL WRITE MESSAGE SEQ.
RDEND    BAL   LNKMES,HEADMS          WRITE MESSAGE
         BAL   LNKMES,SWEXT           READ OPERATOR ANSWER FROM 1052
         BC    15,STRTIO              RETRY OPERATION
         EJECT
***********************************************************************
*                                                                     *
*                         PRINTING ROUTINE                            *
*                                                                     *
*                    NAME= PRNT                                       *
*                                                                     *
***********************************************************************
         SPACE
```

```
PRNT     LH    LINK,10(I)            DECREASE COUNT BY ONE            A2309660
         BCTR  LINK,0                * SKIP CONTROL CHARACTER (SPACE  A2309670
         STH   LINK,CCWPR+6          * 1 LINE OR SKIP TO CHANNE1 1).  A2309680
         L     LINK,MESLNK           ADDRESS OF DATA= BUFF+2. CHARAC- A2309690
         LA    LINK,2(LINK)          * TER IN BYTE LOCATION BUFF+1 IS A2309700
         ST    LINK,CCWPR            * THE CONTROL CHARACTER.         A2309710
PRNTA    MVI   CCWPR,X'09'           SET COMMAND=WRITE,SINGLE SPACE   A2309720
PRNTB    MVI   INVCHN,X'01'          SET UP INVALID CHANNEL BITS      A2309730
         MVI   INVDEV,X'82'          SET UP INVALID SENSE BITS        A2309740
PRINTA   BAL   LINK,CALLA            SUBMIT I/O REQUEST AND WAIT      A2309750
*                                                                     A2309760
PRINTC   TM    CSWPR+4,UE            IS UNIT EXCEPTION PRESENT        A2309770
         BC    8,PRINTD              NO, SKIP NEXT INSTRUCTION        A2309780
         MVI   SKIPSW,X'01'          YES, SET SKIPSW ON               A2309790
PRINTD   TM    CSWPR+4,UC            IS UNIT CHECK PRESENT            A2309800
         BC    8,PRINTG              NO, BRANCH                       A2309810
*                                                                     A2309820
* UNIT CHECK CONDITION PRESENT. ANALYSE SENSE BYTES                   A2309830
*                                                                     A2309840
PRNTDA   TM    CSWPR+4,DE            IS DEVICE END PRESENT            A2309850
         BC    1,PRINTF              YES-FIND OUT ERROR CAUSE         A2309860
         TM    SNSPR,INTREQ          NO-IS INTERVENTION REQUIRED      A2309870
         BC    1,PRINTE              YES-PRINT OPERATOR MESSAGE       A2309880
PRNTDB   TM    RETRSW,X'02'          NO, BUS OUT CHECK, IS RETRY OP.  A2309890
         BC    1,PRNTDC              YES, CALL SEREP                  A2309900
         OI    RETRSW,X'02'          NO, SET RETRY SWITCH ON AND GO   A2309910
         BC    15,RETSIO             * TO RETRY I/O OPERATION.        A2309920
PRNTDC   MVI   SEREP,X'1F'           SET UP SEREP INTERFACE= I/O DEV. A2309930
         BC    15,SRPACT             * FAILURE.                       A2309940
*                                                                     A2309950
PRINTE   TM    INTVSW,X'08'          IS INTVSW=1                      A2309960
         BC    1,RETSIO              YES, RETRY OPERATION             A2309970
         LA    WORK2,INTMES          NO, PREPARE MESSAGE 'INTERV.REQ' A2309980
         MVI   MESACT,X'00'          * NO ANSWER REQUESTED            A2309990
         BAL   LNKMES,HEADMS         WRITE MESSAGE                    A2310000
         MVI   INTVSW,X'08'          SET INTVSW ON                    A2310010
         BC    15,RETSIO             RETRY OPERATION                  A2310020
PRINTF   TM    SNSPR,PRTCHK          BUS OUT OR EQUIP. CHECK PRESENT  A2310030
         BC    8,PRINTG              NO , BRANCH                      A2310040
         TM    ERPRSW,X'04'          YES, IS ERPRSW ON                A2310050
         BC    1,PRINTG              YES, BRANCH                      A2310060
         OI    ERPRSW,X'04'          NO, SET ERPRSW ON                A2310070
         TM    SNSPR,EQUCHK          IS EQUIPMENT CHECK CONDITION     A2310080
         LA    WORK2,EQCMES          PREPARE MESSAGE AND ...          A2310090
         BC    1,PRINTB              ...BRANCH IF YES                 A2310100
         LA    WORK2,BOCMES          CHANGE MESSAGE IF NO             A2310110
PRINTB   MVI   MESACT,X'00'          NO OPERATOR ACTION REQUIRED      A2310120
         BAL   LNKMES,HEADMS         WRITE MESSAGE                    A2310130
         MVC   CCWPR(8),CCWERR       SET CCW TO PRINT WARNING MESSAGE A2310140
         BC    15,PRNTA              * ON PRINTER AND GO TO PRINT IT. A2310150
*                                                                     A2310160
* CARRIAGE CONTROL OPERATIONS                                         A2310170
*                                                                     A2310180
PRINTG   TM    SKIPSW,X'80'          WAS THE LAST OPERATION A SKIP    A2310190
         BC    1,PRINTH              YES, NORMAL RETURN TO CALLER     A2310200
```

```
         TM    SKIPSW,X'01'          NO, IS SKIPSW ON (CHANNEL 12     A2310210
*                                    * DETECTED DURING LAST OPERAT.)  A2310220
         BC    1,PRNSKP              YES, BRANCH                      A2310230
         L     LINK,MESLNK           NO, DOES THE CALLER WANT A  SKIP A2310240
         CLI   1(LINK),C'1'          * TO 1ST LINE, NEXT PAGE.        A2310250
         BC    8,PRNSKP              YES, BRANCH                      A2310260
PRINTH   MVI   SKIPSW,X'00'          *                                A2310270
         BC    15,NRMRET             * CALLER                         A2310280
*                                                                     A2310290
PRNSKP   MVI   CCWPR,X'8B'           SKIP TO CHAN.1 COMMAND           A2310300
         OI    SKIPSW,X'60'          SET 'SKIP OPERAT.' SWITCH ON     A2310310
         BC    15,PRNTB              *                                A2310320
*                                                                     A2310330
CCWERR   CCW   X'09',ERRLNE,0,24     CCW TO PRINT WARNING MESSAGE ON  A2310340
ERRLNE   DC    C'PRINT ERROR IN '    * PRINTER.                       A2310350
         DC    C'LAST LINE'          *                                A2310360
         EJECT                                                        A2310370
*************************************************************************A2310380
*                                                                     * A2310390
*                    TAPE READ OR WRITE  ROUTINE                      * A2310400
*                                                                     * A2310410
*************************************************************************A2310420
         SPACE                                                        A2310430
*                                                                     A2310440
* ENTRY FOR 7-TRACK, BCD, OPERATIONS.                                 A2310450
*                                                                     A2310460
TP7TOP   MVI   RW7TSW,X'10'           SET SW 7TRACK                    A2310470
         MVO   MODSET(1),14(1,WORK7)   SET UP MODE SET COMMAND- DENSITY A2310480
         NI    MODSET,X'C0'          * TRANSLATOR ON                  A2310490
         OI    MODSET,X'2B'          * EVEN PARITY.                   A2310500
         BAL   LINK,CTRLOP           GO TO PERFORM CONTROL OPERATION  A2310510
         TM    14(WORK7),X'10'       IS READ OPERATION                A2310520
         BC    8,TAPEWR              NO-WRITE OPERATION               A2310530
*                                                                     A2310540
* ENTRY  FOR 9-TRACK READING OPERATION                                A2310550
*                                                                     A2310560
TAPERD   MVI   CCWPR,X'02'           SET COMMAND READ                 A2310570
         OI    RW7TSW,X'01'          SET SWITCH READ                  A2310580
         BC    15,TAPEOP             PERFORM READ OPERATION           A2310590
*                                                                     A2310600
* ENTRY  FOR 9-TRACK WRITING OPERATION                                A2310610
*                                                                     A2310620
TAPEWR   MVI   CCWPR,X'01'           SET COMMAND WRITE                A2310630
         OI    RW7TSW,X'02'          SET SWITCH WRITE                 A2310640
         CLC   CCWPR+6(2),DEC1       IS COUNT EQUAL ONE               A2310650
         BC    7,TAPEOP              NO-PERFORM WRITE                 A2310660
         L     WORK2,CCWPR           YES, IS TAPE MARK CHARACTER (7F) A2310670
         CLI   0(WORK2),X'7F'        *                                A2310680
         BC    7,TAPEOP              NO, GO TO PERFORM WRITE OP.      A2310690
         MVI   WTM,X'1F'             YES, SET UP WRITE T.M. COMMAND   A2310700
         BAL   LINK,CTRLOP           GO TO PERFORM WRITE T.M. OPER.   A2310710
         BC    15,NRMRET             RETURN TO CALLER                 A2310720
*                                                                     A2310730
TAPEOP   BAL   LINK,CALLA            PERFORM OPERATION                A2310740
         TM    CSWPR+4,UE            IS UNIT EXCEPTION                A2310750
```

```
         BC    1,TPEOPK                YES, BRANCH                          A2310760
         TM    CSWPR+5,CHC             NO-IS CHAINING CHECK                 A2310770
         BC    1,TPEOPH                YES, BRANCH                          A2310780
         TM    CSWPR+4,UC              NO-IS UNIT CHECK                     A2310790
         BC    8,NRMRET                NO, RETURN TO CALLER                 A2310800
*                                                                           A2310810
* UNIT CHECK CONDITION PRESENT. ANALYZE SENSE BYTES.                        A2310820
*                                                                           A2310830
TPEOP1   TM    SNSPR,CMDRJT            IS COMMAND REJECT                    A2310840
         BC    8,TPEOPG                NO,BRANCH                            A2310850
         TM    RW7TSW,X'02'            IS READ OPERATION                    A2310860
         BC    8,PGRACT                YES,PROGRAM ERROR,BRANCH             A2310870
         TM    SNSPR+1,FILEPR          NO,IS FILE PROTECT ON                A2310880
         BC    8,PGRACT                NO,PROGRAM ERROR,BRANCH              A2310890
         BC    15,TPEOPL               YES,BRANCH TO ISSUE MESSAGE          A2310900
*                                                                           A2310910
TPEOPG   TM    SNSPR,INTREQ            IS INT. REQ.                         A2310920
         BC    8,TPEOP2                NO LOOK FOR DE                       A2310930
TPEOPA   TM    SNSPR+1,ABSTAT          YES-IS EXISTENT TU                   A2310940
         BC    8,SRPACT                NO-EXIT SEREP                        A2310950
TPEOPL   LA    WORK2,INTMES            YES,ISSUE MESSAGE                    A2310960
         BC    15,OPINIT               *                                    A2310970
*                                                                           A2310980
TPEOP2   TM    CSWPR+4,DE              IS DEVICE END PRESENT                A2310990
         BC    8,TPEOP3                NO-INITIAL SELECTION                 A2311000
         MVI   BCKSPC,X'27'            SET UP  BACK SPACE COMMAND           A2311010
         BAL   LINK,CTRLOP             *                                    A2311020
TPEOP3   TM    SNSPR,BUSOUT            IS BUS OUT CHECK                     A2311030
         BC    8,TPEOP5                NO                                   A2311040
         TM    RETRSW,X'04'            YES, IS THIS A RETRY OPERATION       A2311050
         BC    1,TPEOP4                YES, BRANCH                          A2311060
         OI    RETRSW,X'04'            NO, SET RETRY SWITCH ON  AND         A2311070
         BC    15,RETSIO               * TRY AGAIN.                         A2311080
TPEOP4   LA    WORK2,BOCMES            RETRY UNSUCCESSFUL. ISSUE            A2311090
         BC    15,OPINIT               * MESSAGE 'BUS OUT CHECK'.           A2311100
*                                                                           A2311110
TPEOP5   TM    SNSPR,EQUCHK            IS EQUIP. CHECK                      A2311120
         BC    8,TPEOP6                NO                                   A2311130
         LA    WORK2,EQCMES            YES, ISSUE MESSAGE 'EQUIPMENT        A2311140
         BC    15,OPINIT               * CHECK'.                            A2311150
*                                                                           A2311160
TPEOP6   TM    SNSPR,OVERRN            IS OVERRUN                           A2311170
         BC    8,TPEOP8                NO                                   A2311180
         TM    RETRSW,X'08'            YES-IS FIRST ENTRY                   A2311190
         BC    1,TPEOP7                NO-DECREASE RETRY'S COUNT            A2311200
         OI    RETRSW,X'08'            YES-SET SWITCH ON                    A2311210
         LA    WORK4,5                 SET RETRY'S COUNT TO 5               A2311220
TPEOP7   BCT   WORK4,RETSIO            RETRY 5 TIMES                        A2311230
         LA    WORK2,OVRMES            5 TIMES. ISSUE MESSAGE 'OVERRUN'     A2311240
         BC    15,OPINIT               *                                    A2311250
*                                                                           A2311260
TPEOP8   TM    SNSPR,DATCHK            IS DATA CHECK                        A2311270
         BC    8,TPEOPF                NO                                   A2311280
         TM    RW7TSW,X'02'            YES. IS IT A WRITE OPERATION         A2311290
         BC    8,TPEOPC                NO-READ                              A2311300
```

```
         MVI   ERASE,X'17'          SET UP  ERASE  COMMAND          A2311310
         BAL   LINK,CTRLOP          *                               A2311320
         TM    RETRSW,X'10'         IS SWITCH ON                    A2311330
         BC    1,TPEOP9             YES DECREASE BY ONE             A2311340
         OI    RETRSW,X'10'         NO-SET ON                       A2311350
         LA    WORK4,3              RETRY 3 TIMES                   A2311360
TPEOP9   BCT   WORK4,RETSIO         *                               A2311370
TPEOPB   LA    WORK2,DATMES         RETRIES UNSUCCESSFUL. ISSUE     A2311380
         BC    15,OPINIT            * MESSAGE 'DATA CHECK'.         A2311390
TPEOPC   TM    RW7TSW,X'10'         DATA-CHECK READ-IS 7 TRACK      A2311400
         BC    1,TPEOPD             YES, BRANCH- NO,                A2311410
         MVI   TIE,X'DB'            SET UP TRACK IN ERROR COMMAND   A2311420
         BAL   LINK,CTRLOP          *                               A2311430
TPEOPD   TM    RETRSW,X'20'         IS FIRST ENTRY                  A2311440
         BC    1,TPEOPE             NO, BRANCH-  YES,               A2311450
         OI    RETRSW,X'20'         SET RETRY SWITCH ON             A2311460
         LA    WORK4,10             RETRY 10 TIMES                  A2311470
TPEOPE   BCT   WORK4,RETSIO         DECREASE RETRY'S COUNT          A2311480
         OI    TPCLNR,X'01'         SET UP TAPE CLEANER SWITCH      A2311490
         TM    RETRSW,X'80'         IS BCKSPS. PAST THE TAPE CLEANER A2311500
         BC    1,TPEOPB             * ALREADY PERFORMED. YES, BRANCH A2311510
         MVI   BCKSPC,X'27'         SET UP  BACK SPACE COMMAND      A2311520
TPFRWS   LA    WORK4,3              BACKSPACE TAPE PAST TAPE CLEANER A2311530
TPBCKS   BAL   LINK,CTRLOP          *                               A2311540
         BCT   WORK4,TPBCKS         3 TIMES                         A2311550
         TM    RETRSW,X'80'         IS TAPE REPOSITIONNED           A2311560
         BC    1,TPRETR             YES                             A2311570
         OI    RETRSW,X'80'         NO-REPOSITION TAPE              A2311580
         MVI   FRWSPC,X'37'         SET UP  FORWARD SPACE COMMAND   A2311590
         BC    15,TPFRWS            *                               A2311600
TPRETR   NI    RETRSW,X'DF'         RESET DATA CHECK SWITCH         A2311610
         NI    TPCLNR,X'00'         RESET  TAPE CLEANER SWITCH      A2311620
         BC    15,RETSIO            * AND RETRY OPERATION           A2311630
*                                                                    A2311640
TPEOPF   TM    SNSPR+1,NOISE        IS NOISE                        A2311650
         BC    1,TPEOPD             YES-TREAT AS DATA CHECK         A2311660
         BC    15,NRMRET            NO, OPER. OK. RETURN TO CALLER  A2311670
*                                                                    A2311680
* UNIT EXCEPTION CONDITION PRESENT.                                  A2311690
*                                                                    A2311700
TPEOPK   TM    RW7TSW,X'01'         IS READ OPERATION               A2311710
         BC    1,INFACT             YES, ISSUE MESSAGE 'END OF  FILE'A2311720
         LA    WORK2,EOTMES         NO,  ISSUE MESSAGE 'END OF TAPE' A2311730
         BC    15,INFACT+4          * AND RETURN TO CALLER.         A2311740
*                                                                    A2311750
TPEOPH   TM    CSWPR+4,DE           CHAINING CHECK                  A2311760
         BC    8,TPEOPI             IS DE PRESENT. NO, BRANCH - YES, A2311770
         MVI   BCKSPC,X'27'         SET UP  BACK SPACE COMMAND      A2311780
         BAL   LINK,CTRLOP          *                               A2311790
TPEOPI   TM    RETRSW,X'40'         IS FIRST ENTRY                  A2311800
         BC    1,TPEOPJ             NO-                             A2311810
         OI    RETRSW,X'40'         YES-SET SWITCH ON               A2311820
         LA    WORK4,5              AND RETRIES COUNT  TO 5.        A2311830
TPEOPJ   BCT   WORK4,RETSIO         DECREASE COUNT AND RETRY        A2311840
         LA    WORK2,CHNMES         RETRIES UNSUCCESSFUL. ISSUE     A2311850
```

```
         BC    15,OPINIT              * MESSAGE 'CHAINING CHECK'.      A2311860
         SPACE 2                                                       A2311870
*                                                                      A2311880
* CONTROL OPERATIONS                                                   A2311890
*                                                                      A2311900
CTRLOP   ST    LINK,LNKCTL            SAVE CALLING G.R.                A2311910
         MVC   SAVSNS(12),ERRPR       SAVE SENSE BYTES AND CSW         A2311920
         LA    WORK2,MODSET           SET UP IN CAW CONTROL CCW        A2311930
         ST    WORK2,CALL+4           SET NEW CAW                      A2311940
CTRL1    BAL   LINK,CALLA             EXECUTE CONTROL OPERATION        A2311950
*                                                                      A2311960
         TM    CSWPR+4,UC             IS UNIT CHECK PRESENT            A2311970
         BC    8,CTRLEX               NO, BRANCH- YES,                 A2311980
         TM    SNSPR,CMDRJT           IS COMMAND REJECT                A2311990
         BC    1,PGRACT               YES-EXIT PGR. ERROR              A2312000
         TM    SNSPR,INTREQ           IS INTERVENTION REQUIRED         A2312010
         BC    8,CTRL2                NO                               A2312020
         BC    15,TPEOPA              YES, ISSUE MESSAGE 'INTERV.REQ.' A2312030
CTRL2    TM    CSWPR+4,DE             IS-DEVICE END PRESENT            A2312040
         BC    1,CTRL4                YES,BRANCH.                      A2312050
CTRLA    TM    RETRSW,X'01'           NO-INITIAL SELECTION             A2312060
         BC    1,CTRL3                HAS OPER. BEEN RETRIED           A2312070
         OI    RETRSW,X'01'           NO-SET SW ON                     A2312080
         LA    WORK6,3                AND RETRY 3 TIMES                A2312090
CTRL3    BCT   WORK6,CTRL1            *                                A2312100
         BC    15,SRPACT              RETRIES UNSUCCESSFUL -  SEREP    A2312110
*                                                                      A2312120
CTRL4    TM    MODSET,X'07'           IS  A MOD SET  COMMAND           A2312130
         BC    12,CTRLA               YES-RETRY  3 TIMES               A2312140
         TM    TPCLNR,X'01'           IS MOTION PAST THE TAPE CLEANER  A2312150
         BC    8,SRPACT               NO,EXIT TO SEREP - YES,          A2312160
         TM    SNSPR+1,LOADPT         IS TAPE INTO THE LOAD POINT      A2312170
         BC    8,SRPACT               NO,EXIT TO SEREP                 A2312180
         X     WORK4,B30T31           YES- BACKSPACE TO THE LOAD POINT A2312190
         LA    WORK4,1(WORK4)         COMPUTE COUNT OF FORWARD SPACE   A2312200
         MVI   FRWSPC,X'37'           SET UP  FORWARD SPACE COMMAND    A2312210
         OI    RETRSW,X'80'           * AND EXECUTE REPOSITIONNING     A2312220
         BC    15,TPBCKS+4            *                                A2312230
*                                                                      A2312240
CTRLEX   L     LINK,LNKCTL            RETURN TO CALLER                 A2312250
         MVC   ERRPR(12),SAVSNS       RESTORE SENSE BYTES AND CSW      A2312260
         MVC   CALL+5(3),ACCWPR       RESTORE CAW                      A2312270
         NI    RETRSW,X'FE'           *                                A2312280
         BCR   15,LINK                *                                A2312290
*                                                                      A2312300
OPINIT   MVC   MESACT(4),OPACT1       LOAD EXPECTED ACTION             A2312310
         BAL   LNKMES,HEADMS          WRITE OUT MESSAGE                A2312320
         BAL   LNKMES,SNEXT           WAIT FOR ANSWER                  A2312330
         NI    RETRSW,X'00'           ANSWER= START. RESET SWITCH AND  A2312340
         BC    15,RETSIO              * RETRY I/O OPERATION.           A2312350
         SPACE 2                                                       A2312360
*                                                                      A2312370
* CONSTANTS AREA  PECULIAR TO TAPE ROUTINE                             A2312380
*                                                                      A2312390
         SPACE                                                        A2312400
```

```
LNKCTL   DS    F                       SAVE AREA FOR CALLING G.R  LINK  A2312410
SAVSNS   DS    3C                      -    -    -   SENSE BYTES       A2312420
SAVTIE   DS    C                       -    -    -  TIE BYTE            A2312430
SAVCSW   DS    8C                      -    -    -  CSW                 A2312440
*                                                                       A2312450
MODSET   CCW   X'03',SAVTIE,X'00',1    MODE SET COMMAND                 A2312460
TIE      EQU   MODSET                  *                                A2312470
BCKSPC   EQU   MODSET                  *                                A2312480
FRWSPC   EQU   MODSET                  *                                A2312490
ERASE    EQU   MODSET                  *                                A2312500
WTM      EQU   MODSET                  *                                A2312510
B30T31   DC    X'00000003'             *                                A2312520
         EJECT                                                          A2312530
*********************************************************************** A2312540
*                                                                     * A2312550
*                          CONSTANTS AREA                             * A2312560
*                                                                     * A2312570
*********************************************************************** A2312580
         SPACE                                                          A2312590
CALLGR   DS    10F                     SAVE AREA FOR GENERAL REGISTERS  A2312600
*                                        1-10 AFTER SVC 17/18 INTERUPT  A2312610
TEMPGR   DS    10F                     SAVE AREA FOR GENERAL REGISTERS  A2312620
*                                        1-10 BEFORE SUBMITTING I/O     A2312630
*                                        REQUEST.                       A2312640
MESLNK   DS    F                       SAVE AREA FOR ADDRESS OF I/O     A2312650
*                                        BUFFER LOCATED IN CALLING PRG. A2312660
LINKPR   DS    F                       *                                A2312670
LINKCS   DS    F                       SAVE AREA FOR I/O PACKAGE G.R.   A2312680
*                                        LINK IN TYPRD ROUTINE          A2312690
CALLSQ   DS    F                       SAVE AREA FOR ADDRESS OF SVC     A2312700
*                                        17/18 CALLING SEQUENCE         A2312710
CSLID    DC    A(0)                    SAVE AREA FOR ADDRESS OF ENTRY   A2312720
*                                        POINT TO CONTROL PROGRAM       A2312730
IOSW     DC    X'00'                   USED TO INDICATE TYPE OF LOGICAL A2312740
*                                        OPERATION (SVC 17 OR 18)       A2312750
TABSW    DC    X'00'                   USED TO INDICATE TERMINATION OF  A2312760
*                                        I/O OPERATION (O.K  OR NOT)    A2312770
INVDEV   DC    X'00'                   INVALID DEVICE STATUS BITS       A2312780
INVCHN   DC    X'00'                   INVALID SENSE BITS               A2312790
INTVSW   DC    X'00'                   USED FOR  RETRY OPERATION (IN    A2312800
*                                        ALL I/O ROUTINES)              A2312810
RETRSW   EQU   INTVSW                  *                                A2312820
SKIPSW   EQU   INTVSW                  *                                A2312830
ERPRSW   EQU   INTVSW                  *                                A2312840
SW1442   DC    X'00'                   1442,2501,2520/2540 SWITCH       A2312850
RW7TSW   EQU   SW1442                  7TRACK/9TRACK SWITCH             A2312860
PNSW     EQU   SW1442                  *                                A2312870
TPCLNR   DC    X'00'                   USED IN TAPE ROUTINE             A2312880
SNSPR    EQU   ERRPR+1                 *                                A2312890
*                                                                       A2312900
         DS    0D                      *                                A2312910
RETPSW   DC    X'00040000'             USED  IN READ INPUT COMMAND      A2312920
         DC    A(WTCMDR)                 ROUTINE AS A WAITING LOOP      A2312930
CMDPSW   DC    X'00040000'             USED IN READ INPUT COMMAND RTNE  A2312940
         DC    A(0)                      TO ENTER CONTROL PROGRAM READ  A2312950
```

```
*                                       1052 ROUTINE.                    A2312960
CALPSW   DS    D                        SAVE AREA FOR SVC PSW (AFTER     A2312970
*                                       SVC 17/18)                       A2312980
CSLATT   DS    4C                       SAVE AREA FOR PARAMETERS LOCATED A2312990
*                                       IN CONTROL PROGRAM AND USED      A2313000
SVCSL    DS    20C                      IN READ 1052 COMMAND ROUTINE     A2313010
B28T31   DC    X'0000000F'              *                                A2313020
DEC1     DC    H'1'                     *                                A2313030
DEC8     DC    H'8'                     *                                A2313040
DEC34    DC    H'34'                    *                                A2313050
HEXTAB   DC    C'0123456789ABCDEF'      *                                A2313060
ZEROS    DC    X'0000'                  THESE TWO CONSTANTS MUST         A2313070
BLANK    DC    C'    '                    BE CONTIGUOUS                A2313080
         EJECT                                                           A2313090
*                                                                        A2313100
         DS    0F                       SYMBOL TABLE                     A2313110
TABBEG   DC    A(SYMTAB)                ADDR OF START OF SYMBOL  TABLE   A2313120
TABEND   DC    A(SYMTAB)                ADDR OF END   OF SYMBOL  TABLE   A2313130
SYMTAB   DS    200C                     SYMBOL TABLE. EACH ELEHENT IS    A2313140
*                                         20 BYTES LONG AND CONTAINS=    A2313150
*                                       1. SYMBOLIC NAME OF DEV. 8 BYTES A2313160
*                                       2. ADDRESS OF DEVICE     2 BYTES A2313170
*                                       3. TYPE OF DEVICE        4 BYTES A2313180
*                                       4. TYPE OF OPERATION     1 BYTE  A2313190
*                                       5. UNUSED                1 BYTE  A2313200
*                                       6. ADDRESS OF I/O RTNE.  4 BYTES A2313210
*                                                                        A2313220
         DS    0F                       DEVICES TABLE                    A2313230
DEVID    DC    C'1403'                  DEVICE TYPE                      A2313240
         DC    C'O'                     I/O OPERATION TYPE               A2313250
         DC    AL3(PRNT)                SPECIFIC I/O ROUTINE ADDRESS     A2313260
         DC    C'1443'                  *                                A2313270
         DC    C'O'                     *                                A2313280
         DC    AL3(PRNT)                *                                A2313290
         DC    C'1442'                  *                                A2313300
         DC    C'I'                     *                                A2313310
         DC    AL3(RD1442)              *                                A2313320
         DC    C'2501'                  *                                A2313330
         DC    C'I'                     *                                A2313340
         DC    AL3(RD1442)              *                                A2313350
         DC    C'2520'                  *                                A2313360
         DC    C'I'                     *                                A2313370
         DC    AL3(RD1442)              *                                A2313380
         DC    C'2540'                  *                                A2313390
         DC    C'I'                     *                                A2313400
         DC    AL3(RD2540)              *                                A2313410
         DC    C'1052'                  *                                A2313420
         DC    C'I'                     *                                A2313430
         DC    AL3(TYPRD)               *                                A2313440
         DC    C'1052'                  *                                A2313450
         DC    C'I'                     *                                A2313460
         DC    AL3(TYPWRT)              *                                A2313470
         DC    C'2400'                  *                                A2313480
         DC    C'I'                     *                                A2313490
         DC    AL3(TAPERD)              *                                A2313500
```

```
         DC    C'2400'                 *                                A2313510
         DC    C'0'                    *                                A2313520
         DC    AL3(TAPEWR)             *                                A2313530
TP2400   DC    C'2400'                 7-TRACK TAPE IDENTIFICATION      A2313540
         DC    X'00'                   *                                A2313550
         DC    AL3(TP7TOP)             *                                A2313560
         EJECT                                                          A2313570
*                                                                       A2313580
*  THE NEXT PART, UP TO THE END OF PROGRAM, IS OVERLAID  IF THE USER  * A2313590
*  DOES NOT MAKE USE OF THE CARD PUNCHING ROUTINE. THE CORRESPONDING  * A2313600
*  STORAGE AREA THEN CONTAINS  THE  CHANNEL AND UNIT CONTROL BLOCKS,  * A2313610
*  AND THE FIRST PART OF PROGRAM LOADED (SIMULATOR, UPDATER,...).     * A2313620
*                                                                       A2313630
PNCHRT   DC    C'2540'                 *                                A2313640
         DC    C'0'                    *                                A2313650
         DC    AL3(PN2540)             *                                A2313660
         DC    C'1442'                 *                                A2313670
         DC    C'0'                    *                                A2313680
         DC    AL3(PN1442)             *                                A2313690
         DC    C'2520'                 *                                A2313700
         DC    C'0'                    *                                A2313710
         DC    AL3(PN1442)             *                                A2313720
DEVEND   DC    X'00'                   END OF DEVICE TABLE              A2313730
         SPACE 2                                                        A2313740
************************************************************************ A2313750
*                                                                      * A2313760
*                        CARD PUNCHING ROUTINE                         * A2313770
*                                                                      * A2313780
*                    NAME= PNTTTT     (TTTT= 1442,2540)               * A2313790
*                                                                      * A2313800
************************************************************************ A2313810
         SPACE                                                          A2313820
PN1442   MVI   SW1442,X'04'           SET 'SWITCH 1442' ON             A2313830
         MVI   INVDEV,X'83'           SET UP INVALID SENSE BITS        A2313840
         MVI   INVCHN,X'00'           SET UP INVALID CHANNEL BITS      A2313850
         BC    15,PNCAL1              *                                A2313860
*                                                                       A2313870
PN2540   MVI   SW1442,X'02'           SET 'SWITCH 2540' ON             A2313880
         MVI   INVDEV,X'87'           SET UP INVALID SENSE BITS        A2313890
         MVI   INVCHN,X'01'           SET UP INVALID CHANNEL BITS      A2313900
*                                                                       A2313910
PNCAL1   MVI   CCWPR,X'81'            WRITE COMMAND IN CCW             A2313920
PNCALL   BAL   LINK,CALLA             SUBMIT  I/O REQUEST AND WAIT     A2313930
*                                     RETURN HERE AT COMPLETION ...    A2313940
*                                     ...OF OPERATION-RESTORE (LINK)   A2313950
PNCALA   TM    CSWPR+4,UC             IS UNIT CHECK PRESENT            A2313960
         BC    1,PNCALE               YES- LOOK AT SENSE BITS          A2313970
         TM    CSWPR+4,UE             NO-IS UNIT EXCEPTION             A2313980
         BC    1,PNCALD               YES,BRANCH                       A2313990
         TM    CSWPR+5,CHN            NO-IS CHAINING CHECK             A2314000
         BC    1,PNCALF               YES-TREAT.                       A2314010
         CLI   PNSW,X'02'             NO-SUCCESSFUL OPERATION          A2314020
         BC    7,NRMRET               IS NOT A 2540 PUNCH              A2314030
         TM    RETRSW,X'01'           2540 PUNCH - IS AFTER RETRY      A2314040
         BC    1,STRTIO               YES - PUNCH LAST CARD            A2314050
```

```
*       SAVE  IN  PNBUF LAST PUNCHED CARD                                A2314060
*             IN CASE OF ANY ERROR WHEN PUNCHING NEXT CARD               A2314070
PNREST   L     BYTE,MESLNK                                               A2314080
         LH    WORK5,10(I)                                               A2314090
         STH   WORK5,LGPNBF                                              A2314100
         BCTR  WORK5,0                                                   A2314110
         EX    WORK5,SAVPN                                               A2314120
         BC    15,NRMRET                                                 A2314130
SAVPN    MVC   PNBUF(0),1(BYTE)                                          A2314140
PNCALF   LA    WORK2,CHNMES          PREPARE MESSAGE 'CHAINING CHECK'    A2314150
         BC    15,PNCCRD             *                                   A2314160
*                                                                        A2314170
* UNIT CHECK CONDITION PRESENT. ANALYZE SENSE BYTES.                      A2314180
*                                                                        A2314190
PNCALE   TM    SNSPR,INTREQ          IS INT. REQUIRED                    A2314200
         BC    8,PNCALM              NO-LOOK FOR BUS OUT                 A2314210
         LA    WORK2,INTMES          YES-ISSUE MESSAGE                   A2314220
         MVC   MESACT(4),OPACT1      *                                   A2314230
         BC    15,PNEND              *                                   A2314240
*                                                                        A2314250
PNCALM   TM    SNSPR,BUSOUT          IS  BUS OUT                         A2314260
         BC    8,PNCALH              NO-LOOK FOR EQUIPMENT CHECK         A2314270
         LA    WORK2,BOCMES          PREPARE MESSAGE 'BUS OUT CHECK'     A2314280
         BC    15,PNCCRD             *                                   A2314290
*                                                                        A2314300
PNCALH   TM    SNSPR,EQUCHK          IS EQUIPMENT CHECK                  A2314310
         BC    8,PNCALJ              NO-LOOK FOR DATACHECK               A2314320
         LA    WORK2,EQCMES          PREPARE MESSAGE 'EQUIPM. CHECK'     A2314330
         BC    15,PNCCRD             *                                   A2314340
*                                                                        A2314350
PNCALJ   TM    SNSPR,DATCH           IS  DATA CHECK                      A2314360
         BC    8,PNCALK              NO-OVERRUN 1442                     A2314370
         LA    WORK2,DATMES          PREPARE MESSAGE 'DATA CHECK'        A2314380
         BC    15,PNCCRD             *                                   A2314390
*                                                                        A2314400
PNCALK   LA    WORK2,OVRMES          PREPARE MESSAGE 'OVERRUN'           A2314410
         BC    15,PNCCRD                                                 A2314420
*                                                                        A2314430
PNCALD   LA    WORK2,UEMES           PREPARE MESSAGE 'END OF FILE'       A2314440
         BC    15,OPINIT             *                                   A2314450
*                                                                        A2314460
* DETERMINE THE NUMBER OF CARDS TO BE RELOADED BEFORE GOING ON.          A2314470
*                                                                        A2314480
PNCCRD   CLC   CSWPR+1(3),ACCWPR     IS ERROR ON INITIAL SELECTION       A2314490
         BC    4,CRD1                YES,BRANCH                          A2314500
         TM    PNSW,X'02'            IS 2540 PUNCH                       A2314510
         BC    1,CRD4                YES,BRANCH                          A2314520
         MVI   PNNUM,X'F2'                                               A2314530
         BC    15,PCHMES                                                 A2314540
CRD1     TM    PNSW,X'02'                                                A2314550
         BC    1,CRDT                                                    A2314560
         MVI   PNNUM,X'F1'                                               A2314570
         BC    15,PCHMES                                                 A2314580
CRDT     MVI   PNNUM,X'F3'                                               A2314590
         BC    15,PCHMES                                                 A2314600
```

```
CRD4     MVI   PNNUM,X'F4'                                              A2314610
*                                                                       A2314620
PCHMES   MVC   MESACT(4),PNCMES       SET UP CALL WRITE MESSAGE         A2314630
PNEND    BAL   LNKMES,HEADMS          * SEQUENCE                        A2314640
         BAL   LNKMES,SWEXT           READ OPERATOR ANSWER FROM 1052    A2314650
         NI    INTVSW,X'00'           RESET SWITCH                      A2314660
         TM    PNSW,X'02'             IS 2540 PUNCH                     A2314670
         BC    8,RETSIO               NO - RETRY OPERATION              A2314680
*                                                                       A2314690
*      PUNCH LAST CARD SAVED IN PNBUF                                   A2314700
*           BEFORE PUNCHING CURRENT INFORMATION                         A2314710
*                                                                       A2314720
         MVC   CCWPR+1(3),APNBUF                                        A2314730
         MVC   CCWPR+6(2),LGPNBF                                        A2314740
         MVI   RETRSW,X'01'                                             A2314750
         BC    15,RETSIO                                                A2314760
*                                                                       A2314770
* CONSTANTS PECULIAR TO THE CARD PUNCHING ROUTINE.                      A2314780
*                                                                       A2314790
         DS    0F                     COMMENT (COMPLETE MESSAGES)       A2314800
PNCMES   DC    FL1'54'                *                                 A2314810
         DC    AL3(PNACT)             *                                 A2314820
PNACT    DC    C' RUN OUT- SCRAP '    *                                 A2314830
         DC    C'LAST '               *                                 A2314840
PNNUM    DC    C'  CARD(S) AND TYPE ' *                                 A2314850
         DC    C'START OR STOP'       *                                 A2314860
         DC    X'15'                  *                                 A2314870
*                                                                       A2314880
APNBUF   DC    AL3(PNBUF)             *                                 A2314890
LGPNBF   DC    H'80'                  *                                 A2314900
PNBUF    DC    5CL16' '               TEMPORARY BUFFER                  A2314910
*                                                                       A2314920
         DC    A(CSLID)               ADDR. OF 2 ENTRY POINTS TO I/O    A2314930
         DC    A(PNCHRT)              * PACKAGE PROGRAM  FROM INITIA-   A2314940
*                                     * LIZATION PROGRAM.  THESE TWO    A2314950
*                                     * CONSTANTS MUST BE=              A2314960
*                                     * 1. IN THIS ORDER                A2314970
*                                     * 2. CONTIGUOUS                   A2314980
*                                     * 3. IMMEDIATELY BEFORE THE END   A2314990
*                                     *    STATEMENT.                   A2315000
         END                                                            A2315010
         AOPTN CROSSREF                                                 A2400010
A24B     TITLE 'INITIALIZATION PROGRAM FOR CURRENT SYSTEMS SIMULATORS'  A2400020
********************************************************************** A2400030
*                                                                    * A2400040
*                     INITIALIZATION  PROGRAM                        * A2400050
*                                                                    * A2400060
*                              FOR                                   * A2400070
*                                                                    * A2400080
*              IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620             * A2400090
*                                                                    * A2400100
*              ----------------------------------------              * A2400110
*                                                                    * A2400120
*  THE INITIALIZATION PROGRAM IS DESIGNED TO =                       * A2400130
*  1. INITIALIZE THE CONTROL PROGRAM= -READ/WRITE CONSOLE ROUTINE    * A2400140
```

```
*                                    -CHANNEL CONTROL BLOCKS          * A2400150
*                                    -UNIT CONTROL BLOCKS             * A2400160
*  2. CREATE THE SYMBOL TABLE IN THE I/O SUPPORT PACKAGE PROGRAM.     * A2400170
*  3. INITIALIZE THE RELOCATING LOADER PROGRAM=                       * A2400180
*                                    -PROGRAM SELECTION               * A2400190
*                                    -LOADING TABLE  SIZE             * A2400200
*                                    -OUTPUT DEVICE   (FOR  PROGRAM   * A2400210
*                                                      GENERATOR)     * A2400220
*                                    -CONTROL SECTIONS NOT REQUIRED   * A2400230
*                                                                     * A2400240
*  THE PROGRAM STRUCTURE IS DESCRIBED  IN  THE PROGRAM LOGIC MANUAL.  * A2400250
*  MORE PARTICULARLY, THE FIVE PROGRAM PHASES APPEAR  IN  THE SOURCE  * A2400260
*  LISTING.                                                           * A2400270
*  THREE TYPES OF CARD ARE RECOGNIZED BY THE PROGRAM -DEV360, DEVSUP  * A2400280
*  AND CALL- WHICH ARE RESPECTIVELY USED  FOR  THE ABOVE  FUNCTIONS.  * A2400290
*  THE FORMAT AND CONTENTS OF THESE CARDS,  THE  MESSAGES  FORM  THE  * A2400300
*  SUBJECT OF PROGRAM SPECIFICATION MANUAL.                           * A2400310
*                                                                     * A2400320
*                                  *****                              * A2400330
*                                                                     * A2400340
*  EACH CONTROL CARD CONTAINS=                                        * A2400350
*  1. AN IDENTIFICATION CODE  (/)                                     * A2400360
*  2. AN OPERATION CODE       (DEV360,DEVSUP,CALL)                    * A2400370
*  3. ONE OR MORE OPERANDS    (ADDR=X'... )                           * A2400380
*  IN THE FOLLOWING 'CONTROL INFORMATION' IS USED TO  DESIGNATE EACH  * A2400390
*  CARD COMPONENT.                                                    * A2400400
*                                                                     * A2400410
*                                                            .../...  * A2400420
*                                                                     * A2400430
*********************************************************************** A2400440
      EJECT                                                            A2400450
*********************************************************************** A2400460
*                                                                     * A2400470
*  .../...                                                            * A2400480
*                                                                     * A2400490
*  DICTIONARY (DICT)                                                  * A2400500
*  -----------------                                                  * A2400510
*                                                                     * A2400520
*  THE PROGRAM TRANSLATES THE MNEMONIC TERMS IN THE CARDS  BY  MEANS  * A2400530
*  OF THE DICTIONARY. A DICTIONARY WORD CONSISTS OF 14 BYTES =         * A2400540
*    BYTES  0-7  CONTAIN CONTROL INFORMATION                          * A2400550
*    BYTE   8    CONTAINS THE ACTION TO BE TAKEN, TO WIT THE  NUMBER  * A2400560
*                      OF SUBROUTINE DESIGNED TO  PROCESS  CONTROL   * A2400570
*                      INFORMATION. THERE ARE 9 SUBROUTINES CALLED   * A2400580
*                      OPDPR1,...,OPDPR9.                            * A2400590
*     BYTE   9   CONTAINS THE TYPE OF CONTROL INFORMATION. THE BITS   * A2400600
*                      OF BYTE 9 ARE LOGICALLY ADDED TO THE  BITS    * A2400610
*                      OF BYTE AT LOCATION OPDTYP. THAT ALLOWS TO    * A2400620
*                      CHECK FOR TERMS MISSING WHEN THE  CARD  IS    * A2400630
*                      EXHAUSTED.                                    * A2400640
*    BYTE  10    CONTAINS  ONE  MASK USED  BY  THE SUBROUTINE ABOVE-  * A2400650
*                      MENTIONNED.                                   * A2400660
*    BYTES 11-13 CONTAIN THE ADDRESS OF EITHER A SAVE AREA, TO STORE  * A2400670
*                                 TRANSLATED CONTROL  INFORMA-       * A2400680
*                                 TION.                              * A2400690
```

```
*                                       OR   A ROUTINE  TO STORE ONE  * A2400700
*                                       CONTROL CARD,  IN  CONDENSED  * A2400710
*                                       FORM, IN THE TABLE (TABLE).   * A2400720
*                                                                     * A2400730
*  TABLE                                                              * A2400740
*  -----                                                              * A2400750
*                                                                     * A2400760
*  THIS TABLE COMPRISES TWO PARTS. THE FIRST ONE, USED TO STORE  THE  * A2400770
*  CONDENSED DEV360 CARDS IS CONTIGUOUS TO THE  INITIALIZATION  PRO-  * A2400780
*  GRAM. EACH ENTRY CONSISTS OF 8 BYTES  WHICH CONTAIN=               * A2400790
*    BYTES 0-3  TYPE    OF DEVICE.    EX= 2540                      * A2400800
*    BYTES 4-5  ADDRESS  -   -   .      -  000C                     * A2400810
*    BYTE  6    SPECIAL FEATURES.       -  01   (FOR CRDIMG)        * A2400820
*    BYTE  7    INVALID STATUS BITS.    -  E0   (FOR 2540)          * A2400830
*  THE CONDENSED DEV360 CARD IMAGES ARE  SORTED  IN ORDER OF INCREA-  * A2400840
*  SING CHANNEL/DEVICE ADDRESSES.                                     * A2400850
*  THE SECOND PART OF TABLE CONTAIN ONE ENTRY OF  16  BYTES FOR EACH  * A2400860
*  CONDENSED DEVSUP CARD IMAGE=                                       * A2400870
*                                                                     * A2400880
*    BYTES  0-3 CONTAIN  TYPE   OF DEVICE                             * A2400890
*    BYTES  4-5     -    ADDRESS OF   -                               * A2400900
*    BYTE   6    CONTAINS TYPE OF OPERATION  (INPUT OR OUTPUT)        * A2400910
*    BYTE   7    UNUSED                                               * A2400920
*    BYTE   8-15 CONTAIN SYMBOLIC NAME OF DEVICE                      * A2400930
*                                                                     * A2400940
********************************************************************** A2400950
         EJECT                                                         A2400960
********************************************************************** A2400970
*                                                                     * A2400980
*  THE  3  FOLLOWING STATEMENTS DO NOT APPEAR IN THE SOURCE LISTING.  * A2400990
*                                                                     * A2401000
*  NAME  OPERATION  OPERAND                                 COL. 71   * A2401010
*  ----  ---------  -------                                 -------   * A2401020
*                                                                     * A2401030
*        PRINT      OFF                                              * A2401040
*                                                                     * A2401050
*        PUNCH      'CONTENTS OF LDR CARD COLUMNS  1-55        X     * A2401060
*                   CONTENTS  OF LDR CARD COLUMNS 56-80              * A2401070
*                                                                     * A2401080
*  THE PUNCH ASSEMBLER INSTRUCTION CAUSES THE DATA IN THE OPERAND TO  * A2401090
*  BE PUNCHED IN A CARD. THE OPERAND IS WRITTEN  AS A STRING  OF  80  * A2401100
*  CHARACTERS.  THE  POSITION  IMMEDIATELY  TO  THE RIGHT   OF   THE  * A2401110
*  QUOTATION MARK IS REGARDED AS COLUMN 1 OF THE CARD TO BE PUNCHED.  * A2401120
*                                                                     * A2401130
********************************************************************** A2401140
         SPACE                                                         A2401150
         PRINT OFF                                                     A2401160
         PUNCH ' LDR                                                  XA2401170
                               A24B0000'                               A2401180
         PRINT ON                  *                                   A2401190
         SPACE                                                         A2401200
INIT     START 23264               *                                   A2401210
         USING INIT,BASE           *                                   A2401220
         SPACE                                                         A2401230
********************************************************************** A2401240
```

```
*                                                                     * A2401250
*                    GENERAL REGISTER ASSIGNMENT                      * A2401260
*                                                                     * A2401270
********************************************************************** A2401280
         SPACE                                                        A2401290
BASE     EQU   15                  BASE                               A2401300
BEGR     EQU   1                   INDEX  TO  SCAN TABLE   -ODD       A2401310
STEPR    EQU   2                                           -BEGR+1    A2401320
FINR     EQU   3                                           -BEGR+2    A2401330
FINRA    EQU   4                                           -BEGR+3    A2401340
POINTR   EQU   5                   POINTER TO FIELD ON CARD           A2401350
COLR     EQU   6                   NUMBER OF COLUMNS TO BE SCANNED    A2401360
CONTR    EQU   7                   COUNTER OF BYTES IN OPERAND        A2401370
LINK     EQU   8                   LINK FOR SUBROUTINE CALL(LEVEL1)   A2401380
LINKA    EQU   9                   LINK FOR SUBROUTINE CALL(LEVEL2)   A2401390
LINKB    EQU   STEPR               LINK FOR SUBROUTINE CALL(LEVEL3)   A2401400
WORK     EQU   10                  WORKING REGISTER                   A2401410
WORKA    EQU   FINRA               WORKING REGISTER                   A2401420
WORKB    EQU   COLR                WORKING REGISTER                   A2401430
BEGRA    EQU   POINTR              INDEX  TO  SORT TABLE   -ODD       A2401440
STEPRA   EQU   COLR                                        -POINTR+1  A2401450
FINRB    EQU   CONTR                                       -POINTR+2  A2401460
EXITR    EQU   FINR                OUTPUT REGISTER                    A2401470
LOCCTR   EQU   11                  LOCATION COUNTER                   A2401480
         EJECT                                                        A2401490
********************************************************************** A2401500
*                                                                     * A2401510
*          DEFINITION OF PARAMETERS RELATED TO THE DICTIONARY         * A2401520
*                                                                     * A2401530
********************************************************************** A2401540
         SPACE                                                        A2401550
ARG      EQU   0                   ARGUMENT TO REFERENCE THE          A2401560
FUNCT    EQU   0                   * DICTIONARY                       A2401570
ACTION   EQU   8                   NUMBER OF OPERAND PROCESSING       A2401580
*                                  * SUBROUTINE                       A2401590
TYPE     EQU   9                   OPERAND TYPE                       A2401600
MASK     EQU   10                  IMMEDIATE DATA                     A2401610
ADDRSS   EQU   11                  ADDRESS OF SAVE AREA OR PROCES-    A2401620
*                                  * SING ROUTINE                     A2401630
         SPACE                                                        A2401640
********************************************************************** A2401650
*                                                                     A2401660
*         DEFINITION OF PARAMETERS RELATED TO THE CONTROL CARDS       * A2401670
*                                                                     A2401680
********************************************************************** A2401690
         SPACE                                                        A2401700
DVTYPE   EQU   0                   DEVICE TYPE                        A2401710
CHANL    EQU   4                   CHANNEL UNIT ADDRESS               A2401720
DEVICE   EQU   5                   DEVICE  UNIT ADDRESS               A2401730
SPEFT    EQU   6                   SPECIAL FEATURES                   A2401740
IOPTYP   EQU   6                   I/O OPERATION TYPE                 A2401750
INVSTS   EQU   7                   INVALID DEVICE STATUS BITS         A2401760
SYMBOL   EQU   8                   DEVICE SYMBOLIC NAME               A2401770
*                                  INDICATORS=                        A2401780
HEX      EQU   X'01'               HEXADECIMAL INFORMATION            A2401790
```

```
DEC      EQU   X'02'                  DECIMAL     INFORMATION          A2401800
CHAR     EQU   X'04'                  CHARACTERS  INFORMATION          A2401810
         SPACE                                                         A2401820
*********************************************************************** A2401830
*                                                                     * A2401840
*         DEFINITION OF  PARAMETERS  RELATED  TO THE                  * A2401850
*                  CHANNEL  CONTROL  BLOCKS                           * A2401860
*                                                                     * A2401870
*********************************************************************** A2401880
         SPACE                                                         A2401890
DEVTAB   EQU   8                                                       A2401900
DEVADD   EQU   0                                                       A2401910
         EJECT                                                         A2401920
*********************************************************************** A2401930
*                                                                     * A2401940
*         DEFINITION OF  PARAMETERS  RELATED  TO  THE                 * A2401950
*                  UNIT     CONTROL  BLOCKS                           * A2401960
*                                                                     * A2401970
*********************************************************************** A2401980
         SPACE                                                         A2401990
DEVTYP   EQU   0                      DEVICE TYPE                      A2402000
DEV360   EQU   4                        -    ADDRESS                   A2402010
DEVSPF   EQU   6                      SPECIAL FEATURES                 A2402020
INVST    EQU   23                     INVALID DEVICE STATUS BITS       A2402030
DEVATT   EQU   24                     USED IF AN ATTENTION INTERRUPT   A2402040
*                                     * CAN OCCUR ON CONCERNED DEVICE  A2402050
ATT      EQU   X'80'                  ATTENTION                        A2402060
*                                                                      A2402070
OIOPSW   EQU   56                     OLD I/O PSW                      A2402080
CSW      EQU   64                     CHANNEL STATUS WORD              A2402090
NIOPSW   EQU   120                    NEW I/O PSW                      A2402100
         EJECT                                                         A2402110
*********************************************************************** A2402120
*        PHASE 1                                                      * A2402130
*********************************************************************** A2402140
         SPACE                                                         A2402150
*********************************************************************** A2402160
*                                                                     * A2402170
*                        INITIALIZATION ROUTINE                       * A2402180
*                                                                     * A2402190
*                    NAME= INIT                                       * A2402200
*                                                                     * A2402210
*  THIS ROUTINE IS THE FIRST EXECUTED AND IS ENTERED ONCE DURING EXE- * A2402220
*  CUTION OF THE INITIALIZATION PROGRAM. THE STORAGE AREA OCCUPIED BY * A2402230
*  THIS ROUTINE IS OVERLAID BY THE INPUT  BUFFER  (INBUFF)  AND   THE * A2402240
*  TABLE (TABLE).                                                     * A2402250
*  THE ABSOLUTE LOADER PROGRAM TRANSFERS IN=                          * A2402260
*        GENERAL REGISTER 1. THE CURRENT VALUE OF THE LOCATION COUN-  * A2402270
*                            TER= ADDRESS OF THE 1ST BYTE  FOLLOWING  * A2402280
*                            THE I/O SUPPORT PACKAGE.                 * A2402290
*        GENERAL REGISTER 2. THE ADDRESS OF THE LAST  BYTE  OF   THE  * A2402300
*                            CONTROL PROGRAM.                         * A2402310
*        GENERAL REGISTER 3. THE ADDRESS AT WHICH  CONTROL   MUST BE  * A2402320
*                            TRANSFERRED AT THE END OF THE INITIALI-  * A2402330
*                            ZATION PROGRAM (RELOCATING LOADER).      * A2402340
```

```
*                                                                     * A2402350
*********************************************************************** A2402360
         SPACE                                                        A2402370
         BALR  BASE,0                  *                              A2402380
BEGIN    L     BASE,AINIT-BEGIN(0,BASE) *                             A2402390
         SH    2,DEC20                 (2)=A(ENTRY POINT TO CTRL.PROG.) A2402400
         STM   1,3,LOCCNT              SAVE G.R. 1,2,3.               A2402410
         MVC   AJCONS(16),4(2)         SAVE 4 CONSTANTS OF CONTROL PRG. A2402420
         LR    LOCCTR,1                *                              A2402430
         LR    WORK,LOCCTR             *                              A2402440
         SH    WORK,DEC8               (WORK)=A(ENTRY TO I/O PACK.PRG.) A2402450
         MVC   IOPNCH(4),4(WORK)       SAVE A(CARD PUNCHING ROUTINE)  A2402460
         L     WORK,0(WORK)            TRANSFER INTO I/O PACK.PROG. THE A2402470
         MVC   0(4,WORK),CPADDR        * ADDR. OF AN ENTRY TO CTL.PROG. A2402480
INIT1    MVC   IPLDEV(2),2             SAVE IPL DEVICE ADDRESS        A2402490
*                                                                     A2402500
INIT2    MVC   TEMP(8),NIOPSW          SAVE CURRENT NEW I/O PSW  AND  A2402510
         LA    WORK,ATIRUP             * SET NEW RETURN ADDRESS       A2402520
         ST    WORK,NIOPSW+4           *                              A2402530
         CNOP  2,4                     *                              A2402540
ATWAIT   SVC   3                       WAIT FOR CONSOLE ATTENTION     A2402550
         DC    A(ATTPSW)               *                              A2402560
*                                                                     A2402570
         DS    0D                      *                              A2402580
ATTPSW   DC    X'FF06FFFF'             WAIT PSW-ENABLE I/O INTERRUPTS A2402590
         DC    X'FFFFFFFF'             *                              A2402600
*                                                                     A2402610
         EJECT                                                        A2402620
*                                      ENTER HERE WHEN AN I/O INTERRUPT A2402630
ATIRUP   TM    CSW+4,ATT               OCCURS.                        A2402640
         BC    12,ATWAIT               RETURN TO WAIT STATE IF NOT ATT. A2402650
         MVC   NIOPSW(8),TEMP          RESTORE NIOPSW                 A2402660
         LH    WORK,0IOPSW+2           PICK-UP CONSOLE ADDRESS AND    A2402670
         STH   WORK,CONSLE+4           STORE INTO 1052 UCB,           A2402680
         STC   WORK,CH0LST+8           STORE INTO 1052 CCB,           A2402690
         L     WORKA,ACONSL            AND                            A2402700
         STH   WORK,0(0,WORKA)         READ-WRITE CONSOLE ROUTINE     A2402710
         STH   WORK,AD1052             *                              A2402720
         L     WORKA,AJCONS            TRANSFER TO CONTROL PROGRAM    A2402730
         MVC   1(3,WORKA),CH0LST+9     ADDRESS OF UCB 1052            A2402740
*                                                                     A2402750
INIT3    L     WORKA,ACHTAB            A(CHANNEL TABLE). (LOCATED IN  A2402760
         SRL   WORK,8                  * CONTROL PROGRAM).            A2402770
         SLL   WORK,2                  *                              A2402780
         LA    WORKA,0(WORK,WORKA)     A(CHANNEL TABLE ELEMENT)       A2402790
         L     WORKB,0(0,WORKA)        *                              A2402800
         LTR   WORKB,WORKB             IS THERE A DEVICE ON THE CHANNEL A2402810
         BC    7,INIT4                 YES, BRANCH. NO,               A2402820
         BC    0,INIT5                 *                              A2402830
         MVI   *-3,X'F0'               SET SWITCH '1ST PASS' OFF.     A2402840
         ST    WORKA,CHTAB0            SAVE ENTRY POINT TO CHAN.TABLE A2402850
         MVC   0(4,WORKA),CH0LST+4     A(CHANNEL LIST) TO CHAN.TABLE  A2402860
*                                      SET UP CONTR. PROG. TO PROCESS A2402870
*                                      * EXTERNAL INTERRUPTS, INPUT   A2402880
         BAL   LINK,INITA              * COMMANDS AND PROGR. INTERRUPTS A2402890
```

```
         BAL   LINKA,ASSNES           REQUEST CONTROL CARD INPUT DEV.  A2402900
         MVC   SYSINP(4),CMDBUF+1     SET DEV. TYPE IN   PROPER UCB    A2402910
         LA    POINTR,CMDBUF+8        CONVERT DEVICE ADDRESS TO BINARY A2402920
         OC    0(3,POINTR),INIT72     LOWER TO UPPER CASE              A2402930
         LA    CONTR,3                *                                A2402940
         BAL   LINKA,CVRTH1           *                                A2402950
         STH   EXITR,SYSINP+4         STORE IT IN PROPER UCB AND CCB   A2402960
         STC   EXITR,CH1LST+8         *                                A2402970
         LR    WORK,EXITR             *                                A2402980
         BC    15,INIT3               *                                A2402990
*                                                                       A2403000
INIT4    MVC   CH0LST+12(8),CH1LST+8  A(CHANNEL LIST) TO CHANNEL TABLE A2403010
         BC    15,INIT6               *                                A2403020
INIT5    MVC   0(4,WORKA),CH1LST+4    *                                A2403030
*                                                                       A2403040
INIT6    MVC   INIT71(2),SYSINP+4     SUBMIT LOGICAL I/O REQUEST TO    A2403050
         MVC   INIT72(4),SYSINP       * ASSIGN CONTROL CARD INPUT DEV. A2403060
         CNOP  0,4                    *                                A2403070
INIT7    SVC   17                     *                                A2403080
         DC    C'SIM2INF '            * DEVICE SUPPORT FUNCTION (NAME) A2403090
INIT71   DC    X'0000'                *    -   ADDRESS                 A2403100
INIT72   DC    C'    '                *    -   TYPE                    A2403110
         DC    C'I'                   * I/O OPERATION TYPE             A2403120
         DC    AL3(IOPER1)            * ERROR RETURN                   A2403130
         OI    CRDERC+1,X'F0'         MODIFY ERROR SUBROUTINE          A2403140
         EJECT                                                         A2403150
*************************************************************************A2403160
*        PHASE 2                                                      * A2403170
*************************************************************************A2403180
         SPACE                                                         A2403190
*************************************************************************A2403200
*                                                                     * A2403210
*                 READ CONTROL CARD (IMAGE) ROUTINE                   * A2403220
*                                                                     * A2403230
*                   NAME= GETCRD                                      * A2403240
*                                                                     * A2403250
*  THIS ROUTINE IS USED TO READ ONE CARD OR CARD IMAGE (ON  TAPE  OR  * A2403260
*  FROM THE 1052 PRINTER-KEYBOARD) AND TO LIST ITS CONTENTS  ON  THE  * A2403270
*  1052 PRINTER-KEYBOARD.                                             * A2403280
*                                                                     * A2403290
*************************************************************************A2403300
         SPACE                                                         A2403310
GETCRD   MVI   INBUFF,X'40'           CLEAR INPUT BUFFER               A2403320
         MVC   INBUFF+1(80),INBUFF    *                                A2403330
         CNOP  0,4                    SUBMIT LOGICAL I/O REQUEST       A2403340
         SVC   18                     * (READ)                         A2403350
         DC    C'SIM2INF '            * DEVICE SUPPORT FUNCTION (NAME) A2403360
         DC    FL2'72'                * NUMBER OF BYTES OF DATA        A2403370
         DC    A(INBUFF)              * ADDRESS OF INPUT BUFFER        A2403380
         TM    INBUFF,X'07'           *                                A2403390
         BC    12,EXCRET              BRANCH IF READ NOT O.K           A2403400
*                                                                       A2403410
LSTCRD   LA    POINTR,INBUFF          *                                A2403420
         LA    COLR,72                *                                A2403430
LSTCR1   STC   COLR,LSTCR3+4          SCAN INPUT BUFFER FROM RIGHT TO  A2403440
```

```
         STC   COLR,LSTCR0+3          * LEFT TO DETERMINE LENGTH OF    A2403450
LSTCR0   MVI   0(POINTR),X'15'        * CONTROL INFORMATION  (COLR).   A2403460
         BCTR  COLR,0                 *                                A2403470
         STC   COLR,LSTCR2+3          *                                A2403480
LSTCR2   CLI   0(POINTR),X'40'        *                                A2403490
         BC    8,LSTCR1               *                                A2403500
LSTCR3   BAL   LINK,MSDG3             LIST CONTROL INFORMATION ON 1052 A2403510
         DC    X'00'                  * PRINTER-KEYBOARD               A2403520
         DC    AL3(INBUFF)            *                                A2403530
         EJECT                                                         A2403540
*************************************************************************A2403550
*                                                                       * A2403560
*              CONTROL CARD (IMAGE) ANALYZIS ROUTINE                    * A2403570
*                                                                       * A2403580
*                   NAME= CRDAN                                         * A2403590
*                                                                       * A2403600
*  THIS ROUTINE IS ENTERED FROM THE GETCRD ROUTINE. ITS OPERATION IS    * A2403610
*  AS FOLLOWS=                                                          * A2403620
*    1. THE ROUTINE DELIMITS ONE CONTROL INFORMATION (CARD IDENTIFI-    * A2403630
*       CATION, OPERATION CODE, OPERANDS).                              * A2403640
*    2. IT LOOKS FOR THIS INFORMATION IN THE DICTIONARY (DICT).         * A2403650
*    3. IT LINKS TO THE SUBROUTINE DESIGNED TO PROCESS THIS TYPE  OF    * A2403660
*       INFORMATION.                                                    * A2403670
*       IF THERE IS NO MORE INFORMATION IN THE INPUT BUFFER THE ROU-     * A2403680
*       TINE PERFORMS OPERATION 4. OTHERWISE IT PERFORMS OPERATION 1     * A2403690
*    4. THE ROUTINE LINKS TO THE ROUTINE DESIGNED TO STORE THE  CON-    * A2403700
*       TROL INFORMATIONS EXTRACTED FROM THE CARD (IMAGE).               * A2403710
*    5. THE ROUTINE EXITS TO THE GETCRD ROUTINE.                        * A2403720
*                                                                       * A2403730
*  THIS ROUTINE AND THE SUBROUTINES MENTIONNED ABOVE PERFORM A DIAG-     * A2403740
*  NOSTIC CHECK ON THE CONTROL INFORMATION, CHECKING FOR SUCH THINGS     * A2403750
*  AS MISSING OR INVALID CONTROL INFORMATION. MESSAGES  ARE  PRINTED     * A2403760
*  TO INFORM THE OPERATOR OF ANY ERROR DETECTED. THE CONTROL CARD IS     * A2403770
*  IGNORED.                                                             * A2403780
*                                                                       * A2403790
*************************************************************************A2403800
         SPACE                                                         A2403810
CRDAN    LA    POINTR,INBUFF+1        ADDRESS OF INPUT BUFFER          A2403820
         AR    COLR,POINTR            END OF USEFUL BUFFER AREA        A2403830
         XC    TEMP(8),TEMP           CLEAR TEMPORARY AREA             A2403840
         MVI   TEMP+8,X'40'           0 TO FIRST PART                  A2403850
         MVC   TEMP+9(7),TEMP+8       BLANK TO SECOND PART             A2403860
         XC    INFTYP(10),INFTYP                                       A2403870
*                                                                      A2403880
CRDAN1   SR    CONTR,CONTR                                             A2403890
CRDAN2   MVI   CRDAN4+1,X'40'                                          A2403900
         BAL   LINK,CRDAN3            SEARCH FOR SEPARATOR ' '         A2403910
         MVI   CRDAN4+1,X'7D'                                          A2403920
         BAL   LINK,CRDAN3            SEARCH FOR SEPARATOR '''         A2403930
         MVI   CRDAN4+1,X'7C'                                          A2403940
         BAL   LINK,CRDAN3            SEARCH FOR SEPARATOR '''(026)    A2403950
         MVI   CRDAN4+1,C','                                           A2403960
         BAL   LINK,CRDAN3            SEARCH FOR SEPARATOR ','         A2403970
         MVI   CRDAN4+1,C'='                                           A2403980
         BAL   LINK,CRDAN3            SEARCH FOR SEPARATOR '='         A2403990
```

```
         MVI   CRDAN4+1,X'7B'                                          A2404000
         BAL   LINK,CRDAN3          SEARCH FOR SEPARATOR '='(026)      A2404010
         MVI   CRDAN4+1,X'15'                                          A2404020
         BAL   LINK,CRDAN3          SEARCH FOR SEPARATOR 'NL'          A2404030
         LA    CONTR,1(CONTR)       LENGTH OF CONTROL INFORMATION      A2404040
         BC    15,CRDAN2            CONTINUE SCANNING                  A2404050
         EJECT                                                         A2404060
*                                   (POINTR)= A(CONTROL INFORMATION)   A2404070
*                                   (CONTR) = LENGTH OF CONTROL INF.   A2404080
CRDAN3   LA    WORK,0(POINTR,CONTR) *                                  A2404090
         CLR   WORK,COLR            IS BUFFER COMPLETELY SCANNED       A2404100
         BC    4,CRDAN4-4           NO                                 A2404110
         BC    2,CRDAN9             YES                                A2404120
         MVI   0(WORK),X'40'        *                                  A2404130
         OI    0(WORK),X'40'        LOWER TO UPPER CASE                A2404140
CRDAN4   CLI   0(WORK),X'00'        IS CHARACT. ANALYZED A SEPARATOR   A2404150
         BCR   7,LINK               NO,RESUME SCANNING. YES,           A2404160
         LTR   WORK,CONTR           ARE THERE 2 CONSECUTIVE SEPARAT.   A2404170
         BC    8,CRDAN8             YES, BRANCH                        A2404180
*                                                                      A2404190
         STC   WORK,CRDANA+3                                           A2404200
         BCTR  WORK,0                                                  A2404210
         STC   WORK,CRDAN6+1                                           A2404220
         LM    BEGR,FINR,DICT       INITIALIZE DICTIONARY LOOK-UP      A2404230
CRDAN5   BXH   BEGR,STEPR,CRDAN7     *                                  A2404240
CRDAN6   CLC   ARG(1,BEGR),0(POINTR) IS CTRL. INFORMATION IN DICTION.   A2404250
         BC    7,CRDAN5             NO, BRANCH. YES,                   A2404260
         CLI   CRDAN6+1,X'07'       IS LENGTH CORRECT                  A2404270
         BC    8,CRDAN7             *                                  A2404280
CRDANA   CLI   0(BEGR),X'40'        *                                  A2404290
         BC    6,CRDAN5             NO,BRANCH                          A2404300
*                                                                      A2404310
CRDAN7   LA    LINKA,OPDPR          YES                                A2404320
         MVC   *+9(1),ACTION(BEGR)  BRANCH TO CONTROL INFORMATION      A2404330
         BAL   LINK,0(LINKA)        * PROCESSING SUBROUTINE            A2404340
CRDAN8   LA    POINTR,1(CONTR,POINTR) UPDATE POINTR                    A2404350
         BC    15,CRDAN1            RESUME CARD ANALYSIS               A2404360
*                                                                      A2404370
CRDAN9   TM    LABLSW,LABFCT        ARE CARD IDENTIFICATION  AND       A2404380
         BC    12,CRDER2            *OPERTION CODE CORRECT-NO,BRANCH   A2404390
         NI    INFTYP,HEX+DEC       CARD SCANNING COMPLETED-           A2404400
         BC    8,CRDAN8             ANY HEXAD. OR DEC. DATA MISSING    A2404410
         XC    CRDTYP(1),INFTYP     YES-IS IT AN EXCEPTIONAL CASE-NO   A2404420
CRDANB   XI    CRDTYP,X'FF'         CHECK IF ALL OPERANDS              A2404430
         OC    CRDTYP(1),OPDTYP     * REQUIRED ARE PRESENT             A2404440
         TM    CRDTYP,X'FF'         *                                  A2404450
         BC    12,CRDER3            NO,ERROR,BRANCH                    A2404460
CRDANC   OI    CARDSW,X'00'         *                                  A2404470
         MVI   CRDANC+1,X'00'       *                                  A2404480
         L     LINKA,CRDTYP         BRANCH TO CONTROL INFORMATION      A2404490
         BALR  LINK,LINKA           * STORING ROUTINE                  A2404500
         BC    15,GETCRD            GET NEXT CARD                      A2404510
         EJECT                                                         A2404520
*********************************************************************** A2404530
*                                                                     * A2404540
```

```
*            SUBROUTINES TO PROCESS CONTROL INFORMATIONS                 * A2404550
*                                                                        * A2404560
*                  NAME= OPDPRN       WHERE N RANGE FROM 0 UP TO  9     * A2404570
*                                     ACCORDING TO THE TYPE OF CON-      * A2404580
*                                     TROL INFORMATION.                  * A2404590
*                                                                        * A2404600
************************************************************************ A2404610
         SPACE                                                           A2404620
OPDPR    BC    15,OPDPR0              ENTRIES  TABLE                     A2404630
         BC    15,OPDPR1              *                                  A2404640
         BC    15,OPDPR2              *                                  A2404650
         BC    15,OPDPR3              *                                  A2404660
         BC    15,OPDPR4              *                                  A2404670
         BC    15,OPDPR5              *                                  A2404680
         BC    15,OPDPR6              *                                  A2404690
         BC    15,OPDPR7              *                                  A2404700
         BC    15,OPDPR8              *                                  A2404710
         BC    15,OPDPR9              *                                  A2404720
*                                                                        A2404730
OPDPR0   XC    CRDTYP(1),MASK(BEGR)   CTRL. INFO. = CARD IDENTIF. (/)    A2404740
         BC    8,CRDER2               INVALID CARD IN INPUT,BRANCH       A2404750
         BCR   15,LINK                                                   A2404760
*                                                                        A2404770
OPDPR1   XC    CRDTYP(1),TYPE(BEGR)   CTRL. INFO. = OPERAT. CODE         A2404780
         BC    4,CRDER2               INVALID CARD IN INPUT,BRANCH       A2404790
         OI    LABLSW,LABFCT          LABEL AND FUNCTION ARE LEGAL       A2404800
         MVC   CRDTYP,MASK(BEGR)      SAVE MASK AND ADDRESS OF CARD      A2404810
         BCR   15,LINK                PROCESSING ROUTINE                 A2404820
*                                                                        A2404830
OPDPR2   BAL   LINKA,OPDPRX           OPERAND=DEVICE TYPE                A2404840
         STC   WORK,*+5                                                  A2404850
         MVC   0(1,WORKA),FUNCT(BEGR) *STORE FUNCT  =DEVICE TYPE         A2404860
         MVC   INVSTS(1,WORKA),MASK(BEGR) *STORE MASK=INV.STATUS BITS    A2404870
         BCR   15,LINK                                                   A2404880
*                                                                        A2404890
OPDPR3   BAL   LINKA,OPDPRX           OPERAND=CONSTANT PREFIX            A2404900
         NI    INFTYP,HEX+DEC         IS AN HEXAD. OR DECIMAL DATA       A2404910
         BC    8,OPDP13               *EXPECTED-NO,BRANCH                A2404920
         NC    INFTYP(1),CRDTYP       YES,SHOULD THIS DATA BE PRESENT    A2404930
         BC    8,CRDER3               YES BRANCH TO ERROR ROUTINE- NO,   A2404940
OPDP13   ST    WORKA,INFTYP           POINT OUT'HEX.OR DEC.DATA'EXPTED   A2404950
         BCR   15,LINK                                                   A2404960
*                                                                        A2404970
OPDPR4   NC    INFTYP(1),MASK(BEGR)   OPERAND=CONSTANT TYPE(HEX,DEC..)   A2404980
         BC    8,CRDER4               INCORRECT TYPE OF DATA,BRANCH      A2404990
         BCR   15,LINK                                                   A2405000
         EJECT                                                           A2405010
*                                                                        A2405020
OPDPR5   TM    INFTYP,HEX+DEC         OPERAND=CONSTANT OR SYMBOL         A2405030
         BC    8,OPDPR6               BR IF SYMBOL                       A2405040
         TM    INFTYP,HEX                                                A2405050
         BC    8,OPDP15                                                  A2405060
         BAL   LINKA,CVRTH1           CONVERT HEXAD-OPERAND TO BINARY    A2405070
         XI    INFTYP,HEX                                                A2405080
         BC    15,OPDP25                                                 A2405090
```

```
OPDP15   BAL   LINKA,CVRTD1            CONVERT DEC-OPERAND TO BINARY    A2405100
         XI    INFTYP,DEC                                              A2405110
OPDP25   L     LINKA,INFTYP                                            A2405120
         STH   EXITR,0(0,LINKA)       STORE OPERAND CONVERTED          A2405130
         BCR   15,LINK                                                 A2405140
*                                                                      A2405150
OPDPR8   MVC   CRDANC+1(1),MASK(BEGR) FUNCTION SIMSYS OR SIMOUT        A2405160
OPDPR6   CH    WORK,DEC8              IS OPERAND LENGTH G.T 8          A2405170
         BC    10,CRDER4              YES-BRANCH                       A2405180
         LA    LINKA,TEMP+8           *                                A2405190
         TM    INFTYP,CHAR            IS OPERAND SYMBOL OR DATA        A2405200
         BC    12,OPDP16              BRANCH IF SYMBOL                 A2405210
         XI    INFTYP,CHAR            RESET INFORMATION TYPE INDICATOR A2405220
         L     LINKA,INFTYP           LOAD NEW INFORMATION TYPE        A2405230
OPDP16   EX    WORK,OPDP36            SAVE SYMBOL OR DATA              A2405240
         BC    15,OPDPR3              RETURN TO CARD SCANNING SUBROUT. A2405250
OPDP36   MVC   0(1,LINKA),0(POINTR)   *                                A2405260
*                                                                      A2405270
OPDPR7   BAL   LINKA,OPDPRX           OPERAND= FEATURE,I/O OP.TYPE,... A2405280
         OC    0(1,WORKA),MASK(BEGR)                                   A2405290
         BCR   15,LINK                                                 A2405300
*                                                                      A2405310
OPDPR9   BAL   LINKA,OPDPRX           OPERAND= TYPWRT                  A2405320
         MVC   AD1052+2(2),0(WORKA)   SAVE 1052 DEVICE ADDRESS         A2405330
         BCR   15,LINK                                                 A2405340
*                                                                      A2405350
OPDPRX   OC    OPDTYP(1),TYPE(BEGR)   SUB-SUBROUTINE                   A2405360
         MVC   WKAREA(4),MASK(BEGR)                                    A2405370
         L     WORKA,WKAREA                                            A2405380
         BCR   15,LINKA               RETURN TO SUBROUTINE             A2405390
         EJECT                                                         A2405400
*************************************************************************A2405410
*                                                                     * A2405420
*           ROUTINE TO STORE CONTROL INFORMATION IN TABLE             * A2405430
*                                                                     * A2405440
*                  NAME= CTLPR                                        * A2405450
*                                                                     * A2405460
*  THIS ROUTINE HAS TWO ENTRIES -CTLPR AND DEVPR-  USED WHEN CONTROL  * A2405470
*  INFORMATION EXTRACTED FROM CONTROL CARDS -RESPECTIVELY DEV360 AND  * A2405480
*  DEVSUP- ARE BEING STORED IN TABLE. THIS TABLE IS DESCRIBED IN THE  * A2405490
*  PROGRAM LOGIC MANUAL AND IN THE PRESENT PROGRAM LIST, IN THE SEC-  * A2405500
*  TION DEALING WITH THIS TABLE.                                      * A2405510
*                                                                     * A2405520
*************************************************************************A2405530
         SPACE                                                         A2405540
CTLPR    LM    BEGR,FINRA,ADDTAB+8      STORE  CTL.INFO. FROM DEV360  A2405550
         CR    BEGR,STEPR               *                             A2405560
         BC    7,CTLPR1                 CHECKS AND ADJUST TABLE IF NOT A2405570
         AR    STEPR,FINR               ENOUGH ROOM BETWEEN 1ST AND   A2405580
         MVC   0(16,FINRA),0(STEPR)     2ND PART                      A2405590
         AR    FINRA,FINR               *                             A2405600
CTLPR1   LM    BEGRA,STEPRA,ADDTAB      SORT 1ST PART OF TABLE IN ORDER A2405610
         LR    FINRB,BEGR               * OF INCREASING               A2405620
CTLPR2   BXLE  BEGRA,STEPRA,CTLPR4      * CHANNEL/UNIT ADDRESSES      A2405630
         CLC   TEMP+4(2),CHANL(BEGRA)   *                             A2405640
```

```
         BC    8,CTLPR3                DEVICE ALREADY DEFINED,BRANCH    A2405650
         BC    2,CTLPR2                *                                A2405660
         MVC   0(8,BEGR),TEMP          *                                A2405670
         MVC   TEMP(8),0(BEGRA)        *                                A2405680
         MVC   0(8,BEGRA),0(BEGR)      *                                A2405690
         BC    15,CTLPR2               *                                A2405700
CTLPR3   MVC   0(8,BEGRA),TEMP         KEEP NEW DEFINITION OF DEVICE    A2405710
         BCR   15,LINK                 AND GET NEXT CARD                A2405720
CTLPR4   MVC   0(8,BEGR),TEMP          SAVE OPERANDS IN TABLE(1ST PART) A2405730
         CLI   CHANL(BEGR),X'07'       *                                A2405740
         BC    10,CRDER4               CHANNEL GREATER THAN 6,BRANCH    A2405750
         A     BEGR,DEC8               *                                A2405760
         BC    15,DEVPR2               *                                A2405770
*                                      STORE CTL.INFO. FROM DEVSUP      A2405780
DEVPR    LM    BEGR,FINRA,ADDTAB+8     INITIALIZE 2ND TABLE LOOK-UP     A2405790
         MVC   0(16,FINRA),TEMP        SAVE OPERANDS IN TABLE(2ND PART) A2405800
         CLC   SIMSYS(8),SYMBOL(FINRA) DOES THE DEVSUP CARD DEFINE THE  A2405810
         BC    7,DEVPR1                *SUPPORT FCT.USED BY RELOC.LOADR A2405820
         MVC   MESS07+5(8),SYMBOL(FINRA) *NO,BRANCH-YES,SET UP MESSAGE  A2405830
         CLC   INPFCT(1),IOPTYP(FINRA) *CHECK IF OPERATUON IS 'I'       A2405840
         BC    7,CRDER7                *NO,ERROR,BRANCH                 A2405850
         MVC   LODDEV(2),CHANL(FINRA)  *YES,SAVE DEVICE ADDRESS         A2405860
DEVPR1   AR    FINRA,FINR              UPDATE ADDR. OF END OF 2ND PART  A2405870
DEVPR2   STM   BEGR,FINRA,ADDTAB+8     * OF TABLE AND STORE IT.         A2405880
         BCR   15,LINK                 GET NEXT CARD                    A2405890
         EJECT                                                          A2405900
*********************************************************************** A2405910
*                                                                     * A2405920
*        ROUTINE TO PROCESS CONTROL INFORMATION FROM CALL CARD        * A2405930
*                                                                     * A2405940
*                    NAME= CALLPR                                     * A2405950
*                                                                     * A2405960
*  THE FUNCTIONS OF THIS ROUTINE ARE FULLY DESCRIBED IN THE  PROGRAM  * A2405970
*  LOGIC MANUAL.                                                      * A2405980
*                                                                     * A2405990
*********************************************************************** A2406000
         SPACE                                                          A2406010
CALLPR   MVC   PGM(8),TEMP+8           SAVE NAME OF PROGR. TO BE LOADED A2406020
         CLI   EDIT,X'40'              IS EDIT FUNCTION REQUIRED        A2406030
         BC    8,CALLP3                NO,BRANCH - YES,                 A2406040
         LM    BEGR,FINR,SYMTAB        INITIALIZE TABLE LOOK-UP         A2406050
         BXLE  BEGR,STEPR,CALERR       BRANCH IF TABLE EMPTY            A2406060
CALLP1   CLC   EDIT(8),SYMBOL(BEGR)    IS SYMBOL IDENTICAL TO EDIT FCT. A2406070
         BC    8,CALLP2                YES,BRANCH - NO,                 A2406080
         BXH   BEGR,STEPR,CALLP1       BRANCH IF TABLE NOT EXHAUSTED    A2406090
CALERR   MVC   MESS05+5(8),EDIT        EDIT SUPPORT FUNCTION NOT FOUND  A2406100
         BC    15,CRDER5               BRANCH TO ERROR SUBROUTINE       A2406110
CALLP2   CLC   TAPTYP(4),DVTYPE(BEGR)  IS THE  DEVICE A 2400 TAPE UNIT  A2406120
         BC    7,CTLBL                 NO,CARD PUNCH,BRANCH - YES,      A2406130
         MVC   EDITDV(2),CHANL(BEGR)   SAVE DEVICE ADDRESS AND EITHER   A2406140
CALLP3   BC    0,CTLBL                 BRANCH IF PUNCH NOT REQUIRED     A2406150
         L     LOCCTR,IOPNCH           OR OVERLAY PUNCH SUBROUTINE IN   A2406160
         XC    0(1,LOCCTR),0(LOCCTR)   * IO SUPPORT PACKAGE PROGRAM     A2406170
         LA    LOCCTR,1(LOCCTR)        *                                A2406180
         ST    LOCCTR,LOCCNT           *                                A2406190
```

```
         EJECT                                                          A2406200
*************************************************************************A2406210
*        PHASE 3                                                       * A2406220
*************************************************************************A2406230
         SPACE                                                          A2406240
*************************************************************************A2406250
*                                                                      * A2406260
*        ROUTINE TO BUILD CHANNEL AND DEVICE UNIT CONTROL BLOCKS       * A2406270
*                                                                      * A2406280
*************************************************************************A2406290
         SPACE                                                          A2406300
CTLBL    LA    LINKA,3                 ADJUST LOCATION COUNTER ON FULL  A2406310
         BAL   LINK,ADJLC1             *WORD BOUNDARY                   A2406320
         LA    WORKA,CHTAB             IMAGE OF CHANNEL TABLE           A2406330
         XC    0(32,WORKA),0(WORKA)    *                                A2406340
CTLBL1   LH    BEGR,FINR,ADDTAB        INITIALIZE 1ST TABLE PART LOOKUP A2406350
         BXLE  BEGR,STEPR,IOPACK       BR. IF NO DEV360 CARD            A2406360
CTLBL2   SR    WORK,WORK                                                A2406370
         IC    WORK,CHANL(BEGR)        CHANNEL UNIT ADDRESS             A2406380
         SLL   WORK,2                                                   A2406390
         L     WORKB,0(WORK,WORKA)     CHANNEL LIST (CCB)               A2406400
         LTR   WORKB,WORKB             HAS A DEV. BEEN DEFINED ON CCB   A2406410
CTLBL3   BC    0,CTLBL6                (NO-OP WHEN BUILDING CCB'S)      A2406420
         BC    7,CTLBL4                YES, BR. NO, BUILD A NEW CHANNEL A2406430
         XC    0(8,LOCCTR),0(LOCCTR)   * LIST (CCB).                    A2406440
         ST    LOCCTR,0(WORK,WORKA)    A(CHANNEL LIST) TO CHTAB         A2406450
         ST    LOCCTR,4(LOCCTR)                                         A2406460
         AH    LOCCTR,DEC12                                             A2406470
CTLBL4   SH    LOCCTR,DEC4                                              A2406480
         MVC   DEVADD(1,LOCCTR),DEVICE(BEGR) *ADD 1 ELEMENT TO CHN.LIST A2406490
         XC    4(4,LOCCTR),4(LOCCTR)                                    A2406500
         AH    LOCCTR,DEC8             ADD 8 TO LOCCTR                  A2406510
         BXH   BEGR,STEPR,CTLBL2       BR. IF 1ST PART OF TABLE NOT     A2406520
         MVI   CTLBL3+1,X'F0'          * EXHAUSTED.                     A2406530
         BC    15,CTLBL1               *                                A2406540
*                                       BUILD UNIT CONTROL BLOCKS        A2406550
CTLBL5   AH    WORKB,DEC4              *                                A2406560
CTLBL6   CLC   AD1052+2(2),CHANL(BEGR) IS PROCESSED DEVICE THE 1052     A2406570
         BC    6,*+8                   * DEFINED WITH TYPWRT. NO,BRANCH A2406580
         ST    LOCCTR,JCONS1           YES, SAVE ADDRESS OF UCB         A2406590
         CLC   AD1052(2),CHANL(BEGR)   IS PROCESSED DEVICE THE 1052     A2406600
         BC    6,*+8                   * USED NOW. NO,BRANCH            A2406610
         ST    LOCCTR,JCONS            YES,SAVE ADDRESS OF UCB          A2406620
         CLC   DEVTAB(1,WORKB),DEVICE(BEGR) * LOOK FOR THE CCB ELEMENT  A2406630
         BC    6,CTLBL5                * CORRESPONDING TO DEV. PROCES.  A2406640
         ST    LOCCTR,DEVTAB(0,WORKB)  FOUND. BUILD UNIT CONTROL BLOCK  A2406650
         MVC   DEVTAB(1,WORKB),DEVICE(BEGR)                             A2406660
         XC    0(28,LOCCTR),0(LOCCTR)                                   A2406670
         MVC   DEVTYP(7,LOCCTR),DVTYPE(BEGR)                            A2406680
         MVC   INVST(1,LOCCTR),INVSTS(BEGR)                             A2406690
         TM    INVST(LOCCTR),ATT       IS THE DEVICE A 1052 P.K.        A2406700
         BC    1,*+8                   NO, BRANCH. YES, ADD 1 FULL WORD A2406710
         AH    LOCCTR,DEC4             * TO THE UCB.                    A2406720
         AH    LOCCTR,DEC24            UPDATE LOCCTR(LOCATION COUNTER)  A2406730
         BXH   BEGR,STEPR,CTLBL2       BR IF TABLE(1ST PART)NOT EXHAUS. A2406740
```

```
         EJECT                                                                  A2406750
*************************************************************************** A2406760
*        PHASE 4                                                          * A2406770
*************************************************************************** A2406780
         SPACE                                                                  A2406790
*************************************************************************** A2406800
*                                                                         * A2406810
*              ROUTINE TO CREATE THE SYMBOL TABLE  IN THE                 * A2406820
*                          I/O SUPPORT PACKAGE                            * A2406830
*                                                                         * A2406840
*                     NAME= IOPACK                                        * A2406850
*                                                                         * A2406860
*  SEE PROGRAM LOGIC MANUAL, PHASE 4.                                     * A2406870
*  IF THE SIMULATOR SUPPORT DEVICE (USED TO LOAD THE PROGRAM CALLED)      * A2406880
*  OR THE 1052 PRINTER-KEYBOARD ARE NOT DEFINED AT THE END  OF  THIS      * A2406890
*  ROUTINE, THE PROGRAM ISSUES AN ERROR MESSAGE AND ENTER  THE  WAIT      * A2406900
*  STATE.                                                                 * A2406910
*                                                                         * A2406920
*************************************************************************** A2406930
         SPACE                                                                  A2406940
IOPACK   LM    BEGR,FINR,SYMTAB        INITIALIZE 2ND TABLE PART LOOKUP        A2406950
         BXLE  BEGR,STEPR,IOPAC3       BR IF NO SYMBOLIC DEVICE                A2406960
IOPAC0   MVC   IOPAC2+2(8),SYMBOL(BEGR) *                                       A2406970
         MVC   IOPAC2+10(2),CHANL(BEGR) *                                       A2406980
         MVC   IOPAC2+12(4),DVTYPE(BEGR) *                                      A2406990
         MVC   IOPAC2+16(1),IOPTYP(BEGR) *                                      A2407000
         TM    IOPAC2+16,X'0F'         SET UP I/O OPERATION TYPE               A2407010
         BC    5,IOPAC1                *                                       A2407020
         XI    IOPAC2+16,X'D9'         *                                       A2407030
         TM    IOPAC2+16,X'10'         *                                       A2407040
         BC    12,IOPAC1               *                                       A2407050
         XI    IOPAC2+16,X'0F'         *                                       A2407060
IOPAC1   L     WORKA,CHTAB0            SAVE PART OF CHANNEL TABLE USED         A2407070
         L     WORKB,0(WORKA)          TO PRINT MESSAGES                       A2407080
         L     CONTR,ACHTAB                                                    A2407090
         MVC   0(32,CONTR),CHTAB       NEW CHANNEL TABLE                       A2407100
         CNOP  0,4                     SUBMIT LOGICAL I/O REQUEST              A2407110
IOPAC2   SVC   17                      (ASSIGN)                                A2407120
         DC    C'        '             * DEVICE SYMBOLIC NAME                  A2407130
         DC    X'0000'                 *   -    ADDRESS                        A2407140
         DC    C'    '                 *   -    TYPE                           A2407150
         DC    C' '                    * I/O OPERATION TYPE                    A2407160
         DC    AL3(IOPER2)             * ERROR RETURN                          A2407170
         ST    WORKB,0(WORKA)          RESTORE CHANNEL TABLE (PART)            A2407180
         BXH   BEGR,STEPR,IOPAC0       BR.IF 2ND PART OF TABLE NOT EXH.        A2407190
IOPAC3   MVC   MESS05+5(8),SIMSYS      SET UP MESSAGE AND CHECK IF             A2407200
         IC    WORK,SIMIN              DEVICE SUPPORT SIMSYS IS PRESENT        A2407210
         EX    WORK,INITZ1             *                                       A2407220
         BC    8,CRDER5                NO,ERROR,BRANCH - YES,                  A2407230
         MVC   MESS05+5(8),DV1052      SET UP MESSAGE AND CHECK IF             A2407240
         CLC   JCONS(3),JCONS+1        1052 DEV360 HAS BEEN ENCOUNTE-          A2407250
         BC    8,CRDER5                -RED - NO,ERROR,BRANCH - YES,           A2407260
         MVC   MESS05+5(8),SIMOUT      SET UP MESSAGE FOR SIMOUT               A2407270
         TM    ERRSW,ERRBIT            HAS ANY ERROR BEEN DETECTED             A2407280
         BC    8,LOCATE                DURING THE RUN-NO,BRANCH.               A2407290
```

```
         BAL   LINKA,DECIDE           WRITE MESSAGE 10                A2407300
         EJECT                                                        A2407310
*************************************************************************A2407320
*        PHASE 5                                                      * A2407330
*************************************************************************A2407340
         SPACE                                                        A2407350
*************************************************************************A2407360
*                                                                     * A2407370
*                          EXIT ROUTINE                               * A2407380
*                                                                     * A2407390
*                     NAME= LOCATE                                    * A2407400
*                                                                     * A2407410
*  FUNCTIONS= SEE PROGRAM LOGIC MANUAL, PHASE 5.                      * A2407420
*  WHEN THE CORRECT PROGNAME CARD HAS BEEN PROCESSED THE PROGRAM EN-  * A2407430
*  TERS THIS ROUTINE AT LOCATION INITZ.                               * A2407440
*  THE OUTPUT SUPPORT DEVICE USED BY THE RELOCATING LOADER  AND  THE  * A2407450
*  360/DUMP PROGRAMS MUST HAVE BEEN DEFINED IF=                       * A2407460
*        - THE RELOCATING LOADER MUST PRINT LOADING MESSAGES          * A2407470
*        - AND/OR THE DUMP PROGRAM IS PRESENT IN THE PROGRAM CALLED.  * A2407480
*  OTHERWISE ONE MESSAGE INFORMS THE OPERATOR OF THE ERROR  AND  THE  * A2407490
*  PROGRAM STOPS.                                                     * A2407500
*                                                                     * A2407510
*************************************************************************A2407520
         SPACE                                                        A2407530
LOCATE   ST    LOCCTR,LOCCNT          *                              A2407540
         LA    LINKA,7                ADJUST LOCATION COUNTER ON A   A2407550
         BAL   LINK,ADJLC1            *DOUBLE WORD BOUNDARY          A2407560
         LH    BEGR,FINRA,AJCONS      TRANSFER TO CONTROL PROGRAM-   A2407570
         MVC   0(4,BEGR),JCONS        ADDR.OF TYPEWRITER UCB         A2407580
         MVC   0(2,STEPR),AD1052      ADDR.OF TYPEWRITER DEVICE      A2407590
         MVC   0(32,FINR),CHTAB       AND CHANNEL TABLE              A2407600
         MVC   0(2,FINRA),LOCCNT+2    ADDR OF 360 DUMP PROGRAM       A2407610
         MVC   TRFPSW+5(3),RLADDR+1   ADDR OF RELOC. LOADER (ENTRY)  A2407620
         OI    CRDERB+1,X'F0'         MODIFY ERROR SUBROUTINE (RETURN) A2407630
LOCAT1   LH    LINKA,IPLDEV           LOAD IPL DEVICE ADDRESS        A2407640
         BAL   LINK,VERIFY            IS THIS DEVICE A 2400 TAPE UNIT A2407650
         BC    15,LOCAT2              NO,BRANCH - YES,COM-           A2407660
         CH    LINKA,LODDEV           * PARE WITH SUPPORT DEV. SIMSYS A2407670
         BC    8,LOCAT4               EQUAL,BRANCH                   A2407680
         BAL   LINK,REWIND            NOT EQUAL,REWIND AND           A2407690
LOCAT2   LH    LINKA,LODDEV           LOAD DEV.ADDR.OF DEV.SUP. SIMSYS A2407700
         BAL   LINK,VERIFY            IS SIMSYS A MAGNETIC TAPE UNIT A2407710
         SR    LINKA,LINKA            NO.                            A2407720
         BCR   0,0                    *                              A2407730
         STH   LINKA,IPLDEV           YES,STORE ITS DEVICE ADDRESS   A2407740
         BC    15,LOCAT5              AND GET NEXT CARD IMAGE        A2407750
LOCAT3   LH    LINKA,IPLDEV           PROGRAM FOUND HAS NOT THE RIGHT A2407760
         LTR   LINKA,LINKA            NAME-IS SIMSYS DEV. A 2400 TAPE A2407770
         BC    8,LOCERR               NO,BRANCH TO ERROR(PROG.NOT FND) A2407780
LOCAT4   BAL   LINK,FSFILE            YES,FORWARD SPACE FILE         A2407790
         CNOP  0,4                    I/O REQUEST                    A2407800
LOCAT5   SVC   18                     * (GET A CARD IMAGE)           A2407810
         DC    C'SIM2SYS '            * INPUT DEVICE SUPPORT OF LOADER A2407820
         DC    FL2'80'                * INPUT RECORD LENGTH          A2407830
         DC    A(INBUFF)              * INPUT BUFFER ADDRESS         A2407840
```

```
         LA    COLR,71                 * NUMBER OF USEFUL BYTES          A2407850
         EJECT                                                           A2407860
         TM    INBUFF,X'07'            * ANALYZE ERROR TYPE              A2407870
         BC    1,CRDAN                 * NORMAL RETURN                   A2407880
         TM    INBUFF,X'03'            * EXCEPTIONAL RETURN              A2407890
         BC    12,STOP                 *   ERROR I/O,BRANCH              A2407900
         BC    1,LOCERR                *   END OF FILE-(PROG.NOT FND)    A2407910
*                                                                        A2407920
INITZ    NC    IBSIZE(1),LIST          IS OPTION 'LIST' SPECIFIED        A2407930
         BC    8,INITZZ                NO, BRANCH. YES,                  A2407940
         NI    CRDERB+1,X'0F'          MODIFY ERROR SUBROUTINE           A2407950
         MVC   *+7(1),SIMOU            DEVICE SUPPORT SIMOUT IS PRESENT  A2407960
INITZ1   TM    CARDSW,X'00'            *                                 A2407970
         BC    12,CRDER5               NO,ERROR BRANCH - YES,            A2407980
INITZZ   LA    1,PHLIST                LOAD A(PARAMETER  LIST)           A2407990
*                                      *1052 DEVICE ADDRESS,1052 UCB     A2408000
*                                      *ADDRESS,LOADING TABLE SIZE,EDIT  A2408010
*                                      *AND LOAD SUPPORT FUNCTIONS,      A2408020
*                                      *NAMES OF C.S. TO BE SKIPPED      A2408030
*                                      *AND LOCATION COUNTER.            A2408040
         CNOP  2,4                                                      A2408050
         SVC   3                       INITIALISATION COMPLETED         A2408060
         DC    A(TRFPSW)               TRANSFER CONTROL TO RELOC.LOADER A2408070
         EJECT                                                          A2408080
*********************************************************************** A2408090
*                                                                     * A2408100
*                ADJUST LOCATION COUNTER SUBROUTINE                   * A2408110
*                                                                     * A2408120
*                  NAME= ADJLC1                                       * A2408130
*                                                                     * A2408140
*            CALLING SEQUENCE - L   LOCCTR,LOCCNT                     * A2408150
*                               IC  LINKA,-NN-                        * A2408160
*                               BAL LINK,ADJLC1                       * A2408170
*            ADJUSTMENT - NN=00,NONE                                  * A2408180
*                         NN=01,HALF   WORD BOUNDARY                  * A2408190
*                         NN=03,FULL   -      -                       * A2408200
*                         NN=07,DOUBLE -      -                       * A2408210
*            EXIT - LINK                                              * A2408220
*                                                                     * A2408230
*  THIS ROUTINE ADJUSTS THE LOCATION COUNTER (LOCCTR AND LOCCNT)  TO  * A2408240
*  THE BOUNDARY SPECIFIED IN THE CALLING SEQUENCE.                    * A2408250
*                                                                     * A2408260
*********************************************************************** A2408270
         SPACE                                                          A2408280
ADJLC1   EX    LINKA,ADJLC2                                            A2408290
         BCR   8,LINK                                                   A2408300
         LA    LOCCTR,1(LOCCTR)                                        A2408310
         ST    LOCCTR,LOCCNT                                           A2408320
         BC    15,ADJLC1                                               A2408330
ADJLC2   TM    LOCCNT+3,X'00'                                          A2408340
         SPACE 2                                                        A2408350
*********************************************************************** A2408360
*                                                                     * A2408370
*            ROUTINE TO SET PARAMETERS IN CONTROL PROGRAM             * A2408380
*                                                                     * A2408390
```

```
*                    NAME= INITA                                          * A2408400
*                                                                         * A2408410
*************************************************************************** A2408420
         SPACE                                                            A2408430
         CNOP  2,8                      SET UP PARAMETERS FOR PROGRAM     A2408440
INITA    SVC   6                        * INTERRUPTS                      A2408450
         DC    A(PRGCHK)                                                  A2408460
PRGPSW   DS    D                                                          A2408470
*                                                                        A2408480
         CNOP  6,8                      SET UP PARAMETERS FOR EXTERNAL    A2408490
         SVC   10                       * INTERRUPTS                      A2408500
         DC    A(0)                                                       A2408510
         DC    A(INTRET)                                                  A2408520
EXTPSW   DS    D                                                          A2408530
*                                                                        A2408540
         CNOP  6,8                      SET UP PARAMETERS FOR ATTENTION   A2408550
         SVC   5                        * INTERRUPTS                      A2408560
         DC    FL1'11'                  INPUT COMMAND BUFFER (SIZE        A2408570
         DC    AL3(CMDBUF)              * AND ADDRESS)                    A2408580
         DC    X'00'                                                      A2408590
         DC    AL3(COMRET)                                                A2408600
COMPSW   DS    D                                                          A2408610
         BCR   15,LINK                  *                                 A2408620
         EJECT                                                            A2408630
*************************************************************************** A2408640
*                                                                         * A2408650
*              ROUTINE TO PROCESS EXTERNAL INTERRUPT                      * A2408660
*                                                                         * A2408670
*                   NAME= INTRET                                          * A2408680
*                                                                         * A2408690
*  THIS ROUTINE IS ENTERED FROM THE CONTROL PROGRAM EACH TIME A CON-      * A2408700
*  SOLE INTERRUPT OCCURS. TIMER AND EXTERNAL SIGNAL INTERRUPTS ARE        * A2408710
*  IGNORED BY THE CONTROL PROGRAM.                                        * A2408720
*  THE ROUTINE RETURNS CONTROL TO THE POINT OF INTERRUPT.                 * A2408730
*                                                                         * A2408740
*************************************************************************** A2408750
         SPACE                                                            A2408760
INTRET   OI    REPLSW,X'01'                                               A2408770
         CNOP  2,4                                                        A2408780
RESUM1   SVC   3                        RETURN TO POINT OF INTERRUPT      A2408790
         DC    A(EXTPSW)                                                  A2408800
         SPACE                                                            A2408810
*************************************************************************** A2408820
*                                                                         * A2408830
*              ROUTINE TO PROCESS INPUT COMMANDS                          * A2408840
*                                                                         * A2408850
*                   NAME= COMRET                                          * A2408860
*                                                                         * A2408870
*  THIS ROUTINE IS ENTERED FROM THE CONTROL PROGRAM  EACH TIME  THE       * A2408880
*  OPERATOR SEND A COMMAND. THE ROUTINE ANALYZES THE STATUS  SET  BY      * A2408890
*  THE CONTROL PROGRAM IN FIRST LOCATION OF INPUT BUFFER.    IF  THE     * A2408900
*  COMMAND HAS BEEN READ SUCCESSFULLY  CONTROL  IS  RETURNED  TO THE      * A2408910
*  POINT OF INTERRUPT (ATTENTION). OTHERWISE THE ROUTINE REQUESTS  A      * A2408920
*  NEW COMMAND OR ENTERS THE WAIT STATE (STOP).                           * A2408930
*                                                                         * A2408940
```

```
*********************************************************************** A2408950
         SPACE                                                          A2408960
CMDBUF   DC    XL12'0'                 INPUT COMMAND BUFFER            A2408970
COMRET   SVC   9                       ENABLE I/O, EXT. INTERRUPTS     A2408980
         TM    CMDBUF,X'07'            IS COMMAND CORRECTLY READ       A2408990
         BC    8,*-4                   NO, READING NOT COMPLETED       A2409000
         BC    1,COMRE1                YES, BRANCH                     A2409010
         BC    15,STOP                 UNRECOVERABLE ERROR             A2409020
COMRE1   CLC   CMDBUF+1(4),MESTOP      IS INPUT COMMAND 'STOP'         A2409030
         BC    8,STOP                  YES,BRANCH                      A2409040
         OI    REPLSW,X'02'            NO, RETURN TO POINT OF INTERRUPT A2409050
         CNOP  2,4                     *                               A2409060
RESUM2   SVC   3                       *                               A2409070
         DC    A(COMPSW)               *                               A2409080
*                                                                       A2409090
         CNOP  2,4                     *                               A2409100
STOP     SVC   3                       SET THE SYSTEM/360 IN WAIT STATE A2409110
         DC    A(STOPSW)               *                               A2409120
MESTOP   DC    C'STOP'                 *                               A2409130
         EJECT                                                          A2409140
*********************************************************************** A2409150
*                                                                     * A2409160
*                       WRITE MESSAGE ROUTINE                         * A2409170
*                                                                     * A2409180
*                     NAME= MSDG1                                     * A2409190
*                                                                     * A2409200
*  THIS ROUTINE SETS UP TO ISSUE A WRITE MESSAGE REQUEST.  WHEN  THE  * A2409210
*  OPERATION HAS BEEN COMPLETED, THE ROUTINE ANALYZES THE STATUS SET  * A2409220
*  BY THE CONTROL PROGRAM IN FIRST LOCATION OF OUTPUT BUFFER.         * A2409230
*  THE ROUTINE HAS 3 ENTRY POINTS=                                    * A2409240
*                                                                     * A2409250
*    MSDG1     USED WHEN THE OPERATOR MUST ENTER  A  COMMAND   AFTER  * A2409260
*              MESSAGE PRINTING.                                      * A2409270
*    MSDG2     USED WHEN THE OPERATOR MUST PUSH INTERRUPT KEY  AFTER  * A2409280
*              MESSAGE PRINTING.                                      * A2409290
*    MSDG3     USED WHEN NO OPERATOR ACTION IS REQUIRED.              * A2409300
*                                                                     * A2409310
*********************************************************************** A2409320
         SPACE                                                          A2409330
MSDG1    NI    REPLSW,X'FD'            SET SWITCH 'REQUEST COMMAND' ON A2409340
         BC    15,*+8                                                  A2409350
MSDG2    NI    REPLSW,X'FE'            SET SWITCH 'PUSH INTER. KEY' ON A2409360
MSDG3    MVC   CONCCW(4),0(LINK)       SET-UP NEXT CALLING SEQUENCE    A2409370
         SVC   9                       ENABLE I/O, EXT. INTERRUPTS     A2409380
         CNOP  2,4                     SUBMIT 'WRITE MESSAGE'          A2409390
CONSOL   SVC   4                       * REQUEST                       A2409400
CONCCW   DC    X'00'                   * LENGTH OF MESSAGE             A2409410
         DC    XL3'0'                  * ADDRESS -     -               A2409420
         L     WORK,CONCCW             *                               A2409430
         TM    0(WORK),X'07'           IS MESSAGE CORRECTLY WRITTEN    A2409440
         BC    8,*-4                   NO, WRITING NOT COMPLETED       A2409450
         BC    1,WAITIR                YES, BRANCH                     A2409460
         TM    0(WORK),X'03'           NO, IS ERROR UNRECOVERABLE      A2409470
         BC    1,STOP                  YES,BRANCH.  NO,                A2409480
*                                                                       A2409490
```

```
WAITIR   TM    REPLSW,X'03'          IS OPERATOR ACTION REQUIRED      A2409500
         BC    1,4(LINK)             NO, RETURN TO CALLER (ERROR RTN) A2409510
         SVC   19                    YES, WAIT FOR ACTION             A2409520
         BC    15,WAITIR             *                                A2409530
         EJECT                                                        A2409540
*************************************************************************A2409550
*                                                                      * A2409560
*                  I/O DEVICE VERIFICATION SUBROUTINE                  * A2409570
*                                                                      * A2409580
*                   NAME= VERIFY                                       * A2409590
*                                                                      * A2409600
*  THIS ROUTINE VERIFY THAT AN I/O DEVICE WITH A PARTICULAR   SYSTEM   * A2409610
*  360 ADDRESS IS A MAGNETIC TAPE UNIT.                                * A2409620
*                                                                      * A2409630
*    CALLING SEQUENCE   LH  LINKA,DEVICE ADDRESS                       * A2409640
*                       BAL LINK,VERIFY                                * A2409650
*                                                                      * A2409660
*    RETURN             4(LINK)   DEVICE HAS 2400 TYPE                 * A2409670
*                       0(LINK)     -     -  TYPE OTHER THAN 2400      * A2409680
*                                                                      * A2409690
*************************************************************************A2409700
         SPACE                                                        A2409710
VERIFY   STH   LINKA,VERIF1          SET DEV.ADDR. IN THE SVC SEQ.    A2409720
         CNOP  0,4                   I/O VERIFICATION                 A2409730
         SVC   0                     *                                A2409740
VERIF1   DC    X'0000'               * DEVICE ADDRESS                 A2409750
         DC    C'2400'               * DEVICE TYPE (MAGNETIC TAPE)    A2409760
         DC    X'00'                 * SPECIAL FEATURES               A2409770
         DC    AL3(VERIF2)           *                                A2409780
         BC    15,4(LINK)            RETURN TO CALLER-DVCE =2400 TAPE A2409790
VERIF2   BC    15,0(LINK)            RETURN TO CALLER-DVCE NOTE TAPE  A2409800
         EJECT                                                        A2409810
*************************************************************************A2409820
*                                                                      * A2409830
*              REWIND, FORWARD SPACE FILE SUBROUTINE                   * A2409840
*                                                                      * A2409850
*  THIS ROUTINE HAS TWO ENTRY POINTS=                                  * A2409860
*                                                                      * A2409870
*    FSFILE    USED FOR A FORWARD SPACE FILE OPERATION ON TAPE         * A2409880
*    REWIND     -    -  A REWIND MAGNETIC TAPE OPERATION              * A2409890
*                                                                      * A2409900
*    CALLING SEQUENCE   LH  LINKA,DEVICE ADDRESS                       * A2409910
*                       BAL LINK,FSFILE OR REWIND                      * A2409920
*    RETURN             0(LINK)                                        * A2409930
*                                                                      * A2409940
*************************************************************************A2409950
         SPACE                                                        A2409960
FSFILE   MVI   REWCCW,X'3F'          FORM I/O COMMAND FORW.SPACE FILE A2409970
         BC    15,REWIN1             *                                A2409980
*                                                                     A2409990
REWIND   MVI   REWCCW,X'07'          FORM I/O COMMAND REWIND          A2410000
REWIN1   STH   LINKA,REWIN2          SET DEV.ADDR. IN SVC CALLING SEQ A2410010
         CNOP  4,8                   I/O REQUEST AND CONTINUE         A2410020
         SVC   13                    * (CONTROL OPERATION)            A2410030
REWIN2   DC    X'0000'               * DEVICE ADDRESS                 A2410040
```

```
         DC    A(REWCCW)              * CONTROL CCW                     A2410050
         DS    C                      * STATUS BYTE                     A2410060
         DS    3C                     * SENSE BYTES                     A2410070
         DS    D                      * CSW                             A2410080
REWPSW   DS    D                      * PSW AT POINT OF I/O INTERRUPT   A2410090
         DC    A(REWIN3)              * ADDR. OF RETURN WHEN I/O        A2410100
         DC    A(REWIN3)              * INTERUPTS OCCUR                 A2410110
         BC    15,0(LINK)             * RETURN WHEN OPERATION STARTED   A2410120
         CNOP  2,4                    *                                 A2410130
REWIN3   SVC   3                      RETURN TO POINT OF INTERRUPT      A2410140
         DC    A(REWPSW)              *ADDR. OF PSW AT INTERRUPT POINT  A2410150
*                                                                       A2410160
REWCCW   CCW   X'00',*,X'00',1        CONTROL COMMAND CCW (2400 TAPE)   A2410170
         EJECT                                                          A2410180
*********************************************************************** A2410190
*                                                                     * A2410200
*           HEXADECIMAL TO BINARY CONVERSION SUBROUTINE               * A2410210
*                                                                     * A2410220
*                    NAME= CVRTH1                                     * A2410230
*                                                                     * A2410240
*              CALLING SEQUENCE - L   POINTR,START OF FIELD ADDRESS   * A2410250
*                                 L   CONTR,LENGTH OF FIELD(BYTES)    * A2410260
*                                 BAL LINKA,CVRTH1                    * A2410270
*              RESULT RETURNED IN EXITR (RIGHT JUSTIFIED)             * A2410280
*                                                                     * A2410290
*********************************************************************** A2410300
         SPACE                                                          A2410310
CVRTH1   MVI   CVRTH2+3,X'10'                                           A2410320
CVRTHA   SR    EXITR,EXITR                                              A2410330
CVRTH2   LA    WORK,0                                                   A2410340
         LA    WORKA,HEXTAB(WORK)                                       A2410350
CVRTH3   BCTR  WORKA,0                                                  A2410360
         CLC   0(1,POINTR),0(WORKA)                                     A2410370
         BC    8,CVRTH4                                                 A2410380
         BCT   WORK,CVRTH3                                              A2410390
         BC    15,CRDER4              INVALID CHARACTER,BRANCH          A2410400
CVRTH4   BCTR  WORK,0                                                   A2410410
         SLL   EXITR,4                                                  A2410420
         OR    EXITR,WORK                                               A2410430
         LA    POINTR,1(POINTR)                                         A2410440
         BCT   CONTR,CVRTH2                                             A2410450
         BCR   15,LINKA               CONVERSION FINISHED. RETURN       A2410460
*                                                                       A2410470
HEXTAB   DC    C'0123456789ABCDEF'                                      A2410480
         SPACE                                                          A2410490
*********************************************************************** A2410500
*                                                                     * A2410510
*             DECIMAL TO BINARY CONVERSION SUBROUTINE                 * A2410520
*                                                                     * A2410530
*                    NAME= CVRTD1                                     * A2410540
*                                                                     * A2410550
*              CALLING SEQUENCE - L   POINTR,START OF FIELD ADDRESS   * A2410560
*                                 L   CONTR,LENGTH OF FIELD(BYTES)    * A2410570
*                                 BAL LINKA,CVRTD1                    * A2410580
*              RESULT RETURNED IN EXITR  (RIGHT JUSTIFIED)            * A2410590
```

```
*                                                                     * A2410600
***********************************************************************A2410610
         SPACE                                                        A2410620
CVRTD1   MVI   CVRTH2+3,X'0A'                                         A2410630
         LR    LINKB,LINKA             SAVE RETURN ADDRESS            A2410640
         BAL   LINKA,CVRTHA                                          A2410650
         SLL   EXITR,4                                               A2410660
         XC    WKAREA,WKAREA                                         A2410670
         ST    EXITR,WKAREA+4                                        A2410680
         OI    WKAREA+7,X'0F'          SET SIGN PLUS                 A2410690
         CVB   EXITR,WKAREA                                          A2410700
         BCR   15,LINKB                RETURN TO CALLER              A2410710
         EJECT                                                       A2410720
***********************************************************************A2410730
*                                                                     * A2410740
*                       ERROR MESSAGES ROUTINE                        * A2410750
*                                                                     * A2410760
*  THIS ROUTINE SETS UP TO ENTER THE WRITE MESSAGE  ROUTINE  (MSDG).  * A2410770
*  IT IS ENTERED EVERY TIME A MESSAGE IS TO BE WRITTEN ON  THE  1052  * A2410780
*  PRINTER-KEYBOARD.                                                  * A2410790
*                                                                     * A2410800
*  MESSAGES ARE PRINTED IN SEVERAL TIMES=                             * A2410810
*    1. MESSAGE CODE (COMPONENT,SERIAL NUMBER,CHARACTER I,A,...).     * A2410820
*    2. COMMON PART, IF NEED BE.                                      * A2410830
*    3. MESSAGE TEXT.                                                 * A2410840
*                                                                     * A2410850
***********************************************************************A2410860
         SPACE                                                        A2410870
*  VARIOUS ENTRIES.                                                   A2410880
*                                                                     A2410890
LOCERR   MVC   MESS08+5(8),PGM        SET UP MESSAGE TEXT            A2410900
         NI    CRDERB+1,X'0F'         MODIFY ERROR SUBROUTINE AND    A2410910
         BC    15,CRDERB              BRANCH TO WRITE 'CARD INVALID' A2410920
*                                                                     A2410930
PRGERR   LA    WORK,MESS09            PROGRAM ERROR - BRANCH TO      A2410940
         BC    15,MESCOD              * WRITE MESSAGE AND STOP       A2410950
*                                                                     A2410960
PRGCHK   EQU   PRGERR                 PROGRAM ERROR RETURN           A2410970
*                                                                     A2410980
DECIDE   LA    WORK,MESS10            CARD ERROR - WRITE MESSAGE AND A2410990
         BC    15,MESCOD              * WAIT FOR OPERATOR DECISION   A2411000
*                                                                     A2411010
ASSMES   LA    WORK,MESS00            INITIALIZATION - WRITE MESSAGE A2411020
         BC    15,MESCOD              * AND WAIT FOR INPUT COMMAND   A2411030
*                                                                     A2411040
EXCRET   TM    INBUFF,X'03'           CONTROL INFO.INPUT- EXCEP.RETURN A2411050
         BC    12,STOP                UNRECOVERABLE ERROR            A2411060
         BC    15,CRDER2              END OF FILE (SHOULD NOT OCCUR) A2411070
*                                                                     A2411080
*  ENTRIES USED TO INFORM THE OPERATOR OF ERRORS DETECTED DURING      A2411090
*  CREATION OF SYMBOL TABLE IN I/O PACKAGE                            A2411100
*                                                                     A2411110
IOPER2   LA    WORK,IOPAC2                                           A2411120
         ST    WORKB,0(WORKA)         RESTORE CHTAB TO PRINT MESSAGE A2411130
         ST    WORK,ARETRN            SAVE RETURN ADDRESS            A2411140
```

```
         CLI   16(WORK),X'02'                                          A2411150
         BC    4,IOPERA                ERROR TYPE = 01               A2411160
         MVC   MESS07+5(8),2(WORK)     * NAME OF DEVSUP              A2411170
         LA    WORK,MESS07             MESSAGE 'DEVSUP INCORRECT'    A2411180
         BC    15,IOPERB               *                             A2411190
IOPERA   LA    WORK,MESS06             MESSAGE 'TOO MANY DEVSUP'     A2411200
IOPERB   BAL   LINKA,CRDERA            BRANCH TO WRITE MESSAGE SUBRTNE. A2411210
         L     LINKA,ARETRN            *                             A2411220
         BC    15,20(LINKA)            RETURN TO (LINKA)+20          A2411230
         EJECT                                                       A2411240
*  ENTRIES USED TO INFORM OF ERRORS DETECTED WHILE CONTROL INFORMATIONS A2411250
*  ARE PROCESSED.                                                    A2411260
*                                                                  * A2411270
CRDER2   LA    WORK,MESS02                                           A2411280
         BC    15,CRDERX                                             A2411290
CRDER3   LA    WORK,MESS03                                           A2411300
         BC    15,CRDERX                                             A2411310
CRDER4   LA    WORK,MESS04                                           A2411320
         BC    15,CRDERX                                             A2411330
CRDER5   LA    WORK,MESS05                                           A2411340
         BC    15,CRDERX                                             A2411350
CRDER6   LA    WORK,MESS06                                           A2411360
         BC    15,CRDERX                                             A2411370
CRDER7   LA    WORK,MESS07                                           A2411380
         BC    15,CRDERX                                             A2411390
CRDER8   LA    WORK,MESS08                                           A2411400
CRDERX   OI    MESSW,MESBIT            SET SWITCH 'MESSAGE 2 TO 8' ON A2411410
         LA    LINKA,GETCRD            RETURN ADDRESS (AFTER WRITING) A2411420
CRDERA   OI    ERRSW,ERRBIT            SET SWITCH 'CARD ERROR FOUND' ON A2411430
CRDERB   BC    0,LOCAT3                SWITCH USED DURING LOCATE PHASE A2411440
CRDERC   BC    0,MESCOD                SWITCH USED AT THE BEGINNING  A2411450
         LA    WORK,MESS01             * (INCORRECT COMMAND).        A2411460
         OI    MESSW,MESBIT            RESET SWITCH 'MESSAGE 2 TO 8' A2411470
         XI    MESSW,MESBIT            *                             A2411480
*                                                                    A2411490
MESCOD   MVC   MESHDR+4(3),0(WORK)     MESSAGE-SERIAL NUMBER,CHARACTER A2411500
         LA    LINK,4(WORK)            MESSAGE-CALCULATE AND STORE   A2411510
         ST    LINK,MESTXT             * ADDRESS OF TEXT,            A2411520
         MVC   MESTXT(1),3(WORK)       * LENGTH  OF TEXT             A2411530
         MVC   MESCO1+4(1),MESHDR      SET UP NEXT CALLING SEQUENCE AND A2411540
MESCO1   BAL   LINK,MSDG3              CALL WRITE MESSAGE ROUTINE    A2411550
         DC    FL1'0'                  * MESSAGE CODE                A2411560
         DC    AL3(MESHDR+1)           *                             A2411570
         TM    MESSW,MESBIT            IS  COMMON TEXT  TO BE PRINTED A2411580
         BC    12,MESTX2               NO,BRANCH - YES,              A2411590
         XI    MESSW,MESBIT            RESET SWITCH 'MESSAGE 2 TO 8' A2411600
         MVC   MESTX1+4(1),MESS28      SET UP CALLING SEQUENCE AND   A2411610
MESTX1   BAL   LINK,MSDG3              CALL WRITE MESSAGE ROUTINE    A2411620
         DC    FL1'0'                  * MESSAGE TEXT -COMMON PART   A2411630
         DC    AL3(MESS28+1)           *                             A2411640
MESTX2   LA    LINK,MESTXT             *                             A2411650
         CLI   MESHDR+6,C'D'           EXAMINE MESSAGE TYPE -'D',BRANCH A2411660
         BC    8,MSDG2                 * TO WRITE'SUBRTNE, ENTRY 2.  A2411670
         CLI   MESHDR+6,C'A'           * 'A',BRANCH TO WRITE ROUTINE, A2411680
         BC    8,MSDG1                 * ENTRY 1.                    A2411690
```

```
         CNOP  0,4                    *                                A2411700
         BC    15,MSDG3               * 'I,W,S',BR. TO ENTRY 3.        A2411710
MESTXT   DC    FL1'0'                 * MESSAGE TEXT -PART NOT COMMON  A2411720
         DC    AL3(0)                 * (LENGTH AND ADDRESS)           A2411730
         CLI   MESHDR+6,C'W'          MESSAGE PRINTED - WAS TYPE 'W'   A2411740
         BC    8,STOP                 YES,BRANCH,STOP - NO,            A2411750
         BCR   15,LINKA               RETURN TO CALLER                 A2411760
IOPER1   EQU   CRDERC                 INPUT COMMAND ERROR              A2411770
ARETRN   DC    A(0)                                                    A2411780
DV1052   DC    C'1052PK  '            *                                A2411790
         EJECT                                                         A2411800
*************************************************************************A2411810
*                                                                       *A2411820
*                              MESSAGES                                 *A2411830
*                                                                       *A2411840
*  THE FOLLOWING MESSAGES MAY BE PRINTED ON THE 1052 PRINTER  DURING    *A2411850
*  EXECUTION OF THE INITIALIZATION PROGRAM. THEY ARE LISTED AND  EX-    *A2411860
*  PLAINED IN THE PROGRAM SPECIFICATION MANUAL.                         *A2411870
*  THE SPECIFICATIONS OF ONE MESSAGE FORM A SET GROUP DEFINED ON THE    *A2411880
*  MODEL OF MESSAGE 00. THE TWO LETTERS IDENTIFYING   THE  SIMULATOR    *A2411890
*  SYSTEM ARE DEFINED BY A DC INSTRUCTION  AT  THE BEGINNING  OF THE    *A2411900
*  BLOCK OF MESSAGES. THEY APPLY TO ALL THE MESSAGES (00 TO 10).  IN    *A2411910
*  THE SAME WAY THE TEXT DEFINED IN SET GROUP  MESS28   APPLY TO THE    *A2411920
*  MESSAGES  02,...,08.                                                 *A2411930
*                                                                       *A2411940
*************************************************************************A2411950
         SPACE                                                         A2411960
MESHDR   DC    FL1'6'                 MESSAGE CODE                     A2411970
         DC    C' A2    '                                               A2411980
MESS28   DC    FL1'19'                                                 A2411990
         DC    C' CONTROL CARD ER'    TEXT COMMON TO MESSAGES MESS02,  A2412000
         DC    C'ROR,'                ...,MESS08.                      A2412010
*                                                                       A2412020
MESS00   DC    C'00A'                 SEQ.NUMBER,CHARACTER(A,I,W,D,S)* A2412030
         DC    FL1'33'                LENGTH OF TEXT                   A2412040
         DC    C' ASSIGN CONTROL '    TEXT                             A2412050
         DC    C'CARD INPUT DEVIC'    *                                A2412060
         DC    C'E'                   *                                A2412070
         DC    X'15'                  NEW LINE CHARACTER.              A2412080
MESS01   DC    C'01W'                                                  A2412090
         DC    FL1'30'                                                 A2412100
         DC    C' COMMAND ERROR,C'                                     A2412110
         DC    C'ANNOT CONTINUE'                                       A2412120
         DC    X'15'                                                   A2412130
MESS02   DC    C'02W'                                                  A2412140
         DC    FL1'29'                                                 A2412150
         DC    C' INVALID CARD,CA'                                     A2412160
         DC    C'NNOT CONTINUE'                                        A2412170
         DC    X'15'                                                   A2412180
MESS03   DC    C'03I'                                                  A2412190
         DC    FL1'26'                                                 A2412200
         DC    C' TERMS ABSENT,CA'                                     A2412210
         DC    C'RD IGNORED'                                           A2412220
         DC    X'15'                                                   A2412230
MESS04   DC    C'04I'                                                  A2412240
```

```
         DC    FL1'27'                                                  A2412250
         DC    C' INVALID TERMS,C'                                      A2412260
         DC    C'ARD IGNORED'                                           A2412270
         DC    X'15'                                                    A2412280
MESS05   DC    C'05W'                                                   A2412290
         DC    FL1'38'                                                  A2412300
         DC    C'          NOT AS'                                      A2412310
         DC    C'SIGNED,CANNOT CO'                                      A2412320
         DC    C'NTINUE'                                                A2412330
         DC    X'15'                                                    A2412340
         EJECT                                                          A2412350
MESS06   DC    C'06W'                                                   A2412360
         DC    FL1'40'                                                  A2412370
         DC    C' TOO MANY DEVSUP'                                      A2412380
         DC    C' SYMBOLS,CANNOT '                                      A2412390
         DC    C'CONTINUE'                                              A2412400
         DC    X'15'                                                    A2412410
MESS07   DC    C'07I'                                                   A2412420
         DC    FL1'42'                                                  A2412430
         DC    C'          INCORR'                                      A2412440
         DC    C'ECTLY DEFINED,CA'                                      A2412450
         DC    C'RD IGNORED'                                            A2412460
         DC    X'15'                                                    A2412470
MESS08   DC    C'08W'                                                   A2412480
         DC    FL1'33'                                                  A2412490
         DC    C'          INVALI'                                      A2412500
         DC    C'D,CANNOT CONTINU'                                      A2412510
         DC    C'E'                                                     A2412520
         DC    X'15'                                                    A2412530
MESS09   DC    C'09W'                                                   A2412540
         DC    FL1'35'                                                  A2412550
         DC    C' PROGRAM ERROR,C'                                      A2412560
         DC    C'ANNOT CONTINUE,D'                                      A2412570
         DC    C'UMP'                                                   A2412580
         DC    X'15'                                                    A2412590
MESS10   DC    C'10D'                                                   A2412600
         DC    FL1'61'                                                  A2412610
         DC    C' INCORRECT CARDS'                                      A2412620
         DC    C' - EITHER START '                                      A2412630
         DC    C'NEW RUN OR PUSH '                                      A2412640
         DC    C'INTERRUPT KEY'                                         A2412650
         DC    X'15'                                                    A2412660
         EJECT                                                          A2412670
*********************************************************************** A2412680
*                                                                     * A2412690
*                          CONSTANTS AREA                             * A2412700
*                                                                     * A2412710
*********************************************************************** A2412720
         SPACE                                                          A2412730
CHTAB    DS    8F                    TEMPORARY BUFFER USED TO BUILD     A2412740
*                                      CHANNEL TABLE                  A2412750
CHTAB0   DS    1F                    SAVE AREA                          A2412760
*                                                                       A2412770
*  TEMPORARY CHANNEL LISTS AND UNIT CONTROL BLOCKS.                     A2412780
*  (SEE PROGRAM LOGIC MANUAL, PHASE 1).                                 A2412790
```

```
*  IF THE 1052 P.K. AND THE CONTROL INFO. INPUT DEVICE ARE ON THE SAME  A2412800
*  CHANNEL THE PROGRAM GROUPS THE TWO CHANNEL LISTS.                    A2412810
*                                                                       A2412820
CH0LST   DC    A(0)                                                    A2412830
         DC    A(CH0LST)                                               A2412840
         DC    X'00'                                                   A2412850
         DC    AL3(CONSLE)                                             A2412860
         DC    F'0'                                                    A2412870
*                                                                       A2412880
CH1LST   DC    A(0)                                                    A2412890
         DC    A(CH1LST)                                               A2412900
         DC    X'00'                                                   A2412910
         DC    AL3(SYSINP)                                             A2412920
         DC    F'0'                                                    A2412930
*                                                                       A2412940
CONSLE   DC    C'1052'                 PRINTER-KEYBOARD                A2412950
         DC    X'0000'                 DEVICE ADDRESS                  A2412960
         DC    X'0000'                                                 A2412970
         DC    3F'0'                                                   A2412980
         DC    XL3'0'                                                  A2412990
         DC    X'60'                                                   A2413000
         DC    A(0)                                                    A2413010
SYSINP   DC    C'    '                 CONTROL INFO. INPUT DEVICE      A2413020
         DC    X'0000'                 DEVICE ADDRESS                  A2413030
         DC    X'0000'                                                 A2413040
         DC    3F'0'                                                   A2413050
         DC    XL3'0'                                                  A2413060
         DC    X'E0'                                                   A2413070
         EJECT                                                         A2413080
INBUFF   EQU   GETCRD-81               CONTROL INFO. INPUT BUFFER      A2413090
AINIT    DC    A(INIT)                                                 A2413100
PGNMPR   EQU   INITZ                  A(PROGNAME CARD PROCESS. RTNE)  A2413110
*                                                                       A2413120
TABLE    EQU   INBUFF-3                *                               A2413130
ADDTAB   DC    A(TABLE)                START  OF 1ST PART              A2413140
DECM8    DC    F'-8'                   LENGTH OF 1ST PART ELEMENT      A2413150
         DC    A(TABLE-8)              END    -   -   -                A2413160
SYMTAB   DC    A(TABLE-8)              START  -  2ND  -                A2413170
DECM16   DC    F'-16'                  LENGTH -   -   -   ELEMENT      A2413180
         DC    A(TABLE-24)            END      -   -   -               A2413190
*                                                                       A2413200
DEC4     DC    H'4'                                                    A2413210
DEC8     DC    H'8'                                                    A2413220
DEC12    DC    H'12'                                                   A2413230
DEC20    DC    H'20'                                                   A2413240
DEC24    DC    H'24'                                                   A2413250
*                                                                       A2413260
LODDEV   DC    H'0'                                                    A2413270
JCONS    DC    A(0)                    UCB ADDR. OF TYPEWRITER USED NOW A2413280
IOPNCH   DC    A(0)                    A(I/O SUP.PACKAGE PUNCH ROUTINE) A2413290
*                                                                       A2413300
*  PARAMETER  LIST. FOLLOWING CONSTANTS MUST BE CONTIGUOUS AND ORDERED  A2413310
*                AS FOLLOW .                                            A2413320
*                                                                       A2413330
AD1052   DC    A(0)                    TYPEWRITER DEVICE ADDRESS       A2413340
```

```
JCONS1   DC    A(0)                      -      UCB    ADDRESS        A2413350
TBSIZE   DC    A(0)                LOADER TABLES SIZE                A2413360
EDIT     DC    C'      '           EDIT SUPPORT FUNCTION-NAME AND    A2413370
EDITDV   DC    H'0'                * ADDRESS OF DEVICE               A2413380
IPLDEV   DC    H'0'                LOAD SUPPORT FUNCTION-DEV. ADDR.  A2413390
         DC    XL4'0'              *                                 A2413400
LOCCNT   DS    1F                  LOCATION COUNTER                  A2413410
CPADDR   DS    1F                  ENTRY POINT TO CONTROL PROGRAM    A2413420
RLADDR   DS    1F                  -     -     - RELOCATING LOADER   A2413430
AJCONS   DS    1F                  ADDRESSES OF 1052 UCB,            A2413440
ACONSL   DS    1F                  CONSOL AND CHANNEL TABLE          A2413450
ACHTAB   DS    1F                  IN CONTROL PROGRAM                A2413460
ADMP36   DS    1F                  ADDR. OF 360 DUMP PROGRAM         A2413470
PMLIST   EQU   AD1052              ADDRESS OF PARAMETER LIST         A2413480
*                                                                    A2413490
         DS    0D                  *                                 A2413500
TRFPSW   DC    X'000500000F'       TRANSFER PSW (GIVE CONTROL TO     A2413510
         DC    AL3(0)              * RELOCATING LOADER).             A2413520
STOPSW   DC    X'00060000'         STOP PSW= WAIT BIT ON,INTERRUPTS  A2413530
         DC    X'00000000'         * DISABLED (EXCEPT MACHINE CH.)   A2413540
         EJECT                                                       A2413550
*********************************************************************A2413560
*                                                                   *A2413570
*              DICTIONARY OF CONTROL INFORMATIONS                   *A2413580
*                                                                   *A2413590
*********************************************************************A2413600
         SPACE                                                       A2413610
DICT     DC    A(DICT-2)           START OF DICTIONARY               A2413620
         DC    F'14'               LENGTH OF DICTIONARY ELEMENT      A2413630
         DC    A(DICTZ-14)         END OF DICTIONARY                 A2413640
*                                                                    A2413650
         DC    C'/      '          CONTROL CARD AND PROG.HEADER ID.  A2413660
         DC    FL1'0'              ACTION                            A2413670
         DC    X'00'               TYPE                              A2413680
         DC    X'80'               MASK                              A2413690
         DC    AL3(0)              ADDRESS                           A2413700
         DC    C'DEV360 '          CONTROL CARD CODE OPERATION       A2413710
         DC    FL1'4'              ACTION                            A2413720
         DC    X'80'               TYPE                              A2413730
         DC    X'60'               MASK                              A2413740
         DC    AL3(CTLPR)          ADDRESS                           A2413750
         DC    C'DEVSUP '          CONTROL CARD CODE OPERATION       A2413760
         DC    FL1'4'              ACTION                            A2413770
         DC    X'80'               TYPE                              A2413780
         DC    X'54'               MASK                              A2413790
         DC    AL3(DEVPR)          ADDRESS                           A2413800
         DC    C'CALL   '          CONTROL CARD CODE OPERATION       A2413810
         DC    FL1'4'              ACTION                            A2413820
         DC    X'80'               TYPE                              A2413830
         DC    X'11'               MASK                              A2413840
         DC    AL3(CALLPR)         ADDRESS                           A2413850
         DC    C'PROGNAME'         PROGRAM HEADER OPERATION CODE     A2413860
         DC    FL1'4'              ACTION                            A2413870
         DC    X'80'               TYPE                              A2413880
         DC    X'88'               MASK                              A2413890
```

```
DC    AL3(PGNMPR)          ADDRESS                                A2413900
DC    C'SYSINEND'          TRAILER RECORD OPERATION CODE          A2413910
DC    FL1'4'               ACTION                                 A2413920
DC    X'80'                TYPE                                   A2413930
DC    X'00'                MASK                                   A2413940
DC    AL3(LOCERR)          ADDRESS                                A2413950
DC    C'X      '           CTRL CARD DATA TYPE (HEXADECIM.)       A2413960
DC    FL1'16'              ACTION                                 A2413970
DC    X'00'                TYPE                                   A2413980
DC    X'01'                MASK                                   A2413990
DC    AL3(0)               ADDRESS                                A2414000
DC    C'F      '           CTRL CARD DATA TYPE    (DECIMAL)       A2414010
DC    FL1'16'              ACTION                                 A2414020
DC    X'00'                TYPE                                   A2414030
DC    X'02'                MASK                                   A2414040
DC    AL3(0)               ADDRESS                                A2414050
DC    C'ADDR   '           CTRL CARD DATA PREFIX                  A2414060
DC    FL1'12'              ACTION                                 A2414070
DC    X'20'                TYPE                                   A2414080
DC    X'01'                MASK                                   A2414090
DC    AL3(TEMP+4)          ADDRESS                                A2414100
DC    C'TBSIZ  '           CTRL CARD DATA PREFIX                  A2414110
DC    FL1'12'              ACTION                                 A2414120
DC    X'08'                TYPE                                   A2414130
DC    X'02'                MASK                                   A2414140
DC    AL3(TBSIZE+2)        ADDRESS                                A2414150
DC    C'1442   '           CTRL CARD OPERAND                      A2414160
DC    FL1'8'               ACTION                                 A2414170
DC    X'40'                TYPE                                   A2414180
DC    X'E0'                MASK                                   A2414190
DC    AL3(TEMP)            ADDRESS                                A2414200
DC    C'2520   '           CTRL CARD OPERAND                      A2414210
DC    FL1'8'               ACTION                                 A2414220
DC    X'40'                TYPE                                   A2414230
DC    X'E0'                MASK                                   A2414240
DC    AL3(TEMP)            ADDRESS                                A2414250
DC    C'2501   '           CTRL CARD OPERAND                      A2414260
DC    FL1'8'               ACTION                                 A2414270
DC    X'40'                TYPE                                   A2414280
DC    X'E0'                MASK                                   A2414290
DC    AL3(TEMP)            ADDRESS                                A2414300
DC    C'1052   '           CTRL CARD OPERAND                      A2414310
DC    FL1'8'               ACTION                                 A2414320
DC    X'40'                TYPE                                   A2414330
DC    X'60'                MASK                                   A2414340
DC    AL3(TEMP)            ADDRESS                                A2414350
DC    C'1443   '           CTRL CARD OPERAND                      A2414360
DC    FL1'8'               ACTION                                 A2414370
DC    X'40'                TYPE                                   A2414380
DC    X'E0'                MASK                                   A2414390
DC    AL3(TEMP)            ADDRESS                                A2414400
DC    C'2540   '           CTRL CARD OPERAND                      A2414410
DC    FL1'8'               ACTION                                 A2414420
DC    X'40'                TYPE                                   A2414430
DC    X'E0'                MASK                                   A2414440
```

```
         DC    AL3(TEMP)          ADDRESS                          A2414450
         DC    C'1403    '        CTRL.CARD OPERAND                A2414460
         DC    FL1'8'             ACTION                           A2414470
         DC    X'40'              TYPE                             A2414480
         DC    X'E0'              MASK                             A2414490
         DC    AL3(TEMP)          ADDRESS                          A2414500
TAPTYP   DC    C'2400    '        CTRL.CARD OPERAND                A2414510
         DC    FL1'8'             ACTION                           A2414520
         DC    X'40'              TYPE                             A2414530
         DC    X'80'              MASK                             A2414540
         DC    AL3(TEMP)          ADDRESS                          A2414550
         DC    C'2311    '        CTRL.CARD OPERAND                A2414560
         DC    FL1'8'             ACTION                           A2414570
         DC    X'40'              TYPE                             A2414580
         DC    X'80'              MASK                             A2414590
         DC    AL3(TEMP)          ADDRESS                          A2414600
         DC    C'7TRACK  '        CTRL.CARD MAGNETIC TAPE FORMAT   A2414610
         DC    FL1'28'            ACTION                           A2414620
         DC    X'00'              TYPE                             A2414630
         DC    X'01'              MASK                             A2414640
         DC    AL3(TEMP+6)        ADDRESS                          A2414650
         DC    C'DATCVR  '        CTRL.CARD MAGNETIC TAPE FEATURE  A2414660
         DC    FL1'28'            ACTION                           A2414670
         DC    X'00'              TYPE                             A2414680
         DC    X'02'              MASK                             A2414690
         DC    AL3(TEMP+6)        ADDRESS                          A2414700
         DC    C'9TRACK  '        CTRL.CARD MAGNETIC TAPE FORMAT   A2414710
         DC    FL1'28'            ACTION                           A2414720
         DC    X'00'              TYPE                             A2414730
         DC    X'04'              MASK                             A2414740
         DC    AL3(TEMP+6)        ADDRESS                          A2414750
         DC    C'COLBIN  '        CTRL.CARD CRD READER-PCH FEATURE A2414760
         DC    FL1'28'            ACTION                           A2414770
         DC    X'00'              TYPE                             A2414780
         DC    X'01'              MASK                             A2414790
         DC    AL3(TEMP+6)        ADDRESS                          A2414800
         DC    C'132BAR  '        CTRL.CARD PRINTER FEATURE        A2414810
         DC    FL1'28'            ACTION                           A2414820
         DC    X'00'              TYPE                             A2414830
         DC    X'01'              MASK                             A2414840
         DC    AL3(TEMP+6)        ADDRESS                          A2414850
         DC    C'144BAR  '        CTRL.CARD PRINTER FEATURE        A2414860
         DC    FL1'28'            ACTION                           A2414870
         DC    X'00'              TYPE                             A2414880
         DC    X'01'              MASK                             A2414890
         DC    AL3(TEMP+6)        ADDRESS                          A2414900
         DC    C'CRDIMG  '        CTRL.CARD CRD READER-PCH FEATURE A2414910
         DC    FL1'28'            ACTION                           A2414920
         DC    X'00'              TYPE                             A2414930
         DC    X'01'              MASK                             A2414940
         DC    AL3(TEMP+6)        ADDRESS                          A2414950
         DC    C'I       '        CTRL CARD SUPPORT FUNCTION TYPE  A2414960
         DC    FL1'28'            ACTION                           A2414970
         DC    X'04'              TYPE                             A2414980
INPFCT   DC    X'10'              MASK                             A2414990
```

```
         DC    AL3(TEMP+6)          ADDRESS                              A2415000
         DC    C'0       '          CTRL.CARD SUPPORT FUNCTION TYPE      A2415010
         DC    FL1'28'              ACTION                               A2415020
         DC    X'04'                TYPE                                 A2415030
         DC    X'00'                MASK                                 A2415040
         DC    AL3(TEMP+6)          ADDRESS                              A2415050
         DC    C'200     '          CTRL.CARD SUP.FCT. TAPE DENSITY      A2415060
         DC    FL1'28'              ACTION                               A2415070
         DC    X'00'                TYPE                                 A2415080
         DC    X'02'                MASK                                 A2415090
         DC    AL3(TEMP+6)          ADDRESS                              A2415100
         DC    C'556     '          CTRL.CARD SUP.FCT. TAPE DENSITY      A2415110
         DC    FL1'28'              ACTION                               A2415120
         DC    X'00'                TYPE                                 A2415130
         DC    X'05'                MASK                                 A2415140
         DC    AL3(TEMP+6)          ADDRESS                              A2415150
         DC    C'800     '          CTRL.CARD SUP.FCT. TAPE DENSITY      A2415160
         DC    FL1'28'              ACTION                               A2415170
         DC    X'00'                TYPE                                 A2415180
         DC    X'08'                MASK                                 A2415190
         DC    AL3(TEMP+6)          ADDRESS                              A2415200
SIMSYS   DC    C'SIM2SYS '          * CTRL.CARD SUP.FCT.SYSTEM INPUT     A2415210
         DC    FL1'32'              ACTION                               A2415220
         DC    X'10'                TYPE                                 A2415230
SIMIN    DC    X'21'                MASK                                 A2415240
         DC    AL3(TEMP+4)          ADDRESS                              A2415250
SIMOUT   DC    C'SIM2PRNT'          * CTRL.CARD SUP.FCT.SYSTEM OUTP.     A2415260
         DC    FL1'32'              ACTION                               A2415270
         DC    X'10'                TYPE                                 A2415280
SIMOU    DC    X'11'                MASK                                 A2415290
         DC    AL3(TEMP+4)          ADDRESS                              A2415300
TYPWRT   DC    C'TYPWRT  '          CTRL.CARD OPERAND (ASSIGN 1052)      A2415310
         DC    FL1'36'              ACTION                               A2415320
         DC    X'00'                TYPE                                 A2415330
         DC    X'00'                MASK                                 A2415340
         DC    AL3(TEMP+4)          ADDRESS                              A2415350
         DC    C'INIT    '          CTRL.CARD OPERAND (CALL EDIT)        A2415360
         DC    FL1'12'              ACTION                               A2415370
         DC    X'00'                TYPE                                 A2415380
         DC    X'04'                MASK                                 A2415390
         DC    AL3(EDIT)            ADDRESS                              A2415400
         DC    C'PUNCH   '          CTRL.CARD OPERAND (PUNCH OPTION)     A2415410
         DC    FL1'28'              ACTION                               A2415420
         DC    X'00'                TYPE                                 A2415430
         DC    X'F0'                MASK                                 A2415440
         DC    AL3(CALLP3+1)        ADDRESS                              A2415450
         DC    C'LIST    '          CTRL.CARD OPERAND (LOADING MES.)     A2415460
         DC    FL1'28'              ACTION                               A2415470
         DC    X'00'                TYPE                                 A2415480
LIST     DC    X'03'                MASK                                 A2415490
         DC    AL3(TBSIZE)          ADDRESS                              A2415500
PGM      DC    C'        '          CTRL.CARD PROGRAM CALLED (NAME)      A2415510
         DC    FL1'20'              ACTION                               A2415520
         DC    X'80'                TYPE                                 A2415530
         DC    X'00'                MASK                                 A2415540
```

```
         DC    AL3(TEMP+4)          ADDRESS                                A2415550
DICTZ    DC    C'ZZZZZZZZ'          CTRL.CARD DATA (HEX.,DEC.,CHAR.)       A2415560
         DC    FL1'20'              ACTION                                 A2415570
         DC    X'10'                TYPE                                   A2415580
         DC    X'01'                MASK                                   A2415590
         DC    AL3(TEMP+4)          ADDRESS                                A2415600
*                                                                           A2415610
WKAREA   DS    1D                   WORKING AREA                           A2415620
TEMP     DS    2D                   SAVE AREA FOR CONTROL INFORMAT.        A2415630
*                                     EXTRACTED FROM CONTROL CARD          A2415640
INFTYP   DC    A(0)                 *                                      A2415650
CRDTYP   DC    A(0)                 *                                      A2415660
OPDTYP   DC    X'00'                *                                      A2415670
LABLSW   DC    X'00'                *                                      A2415680
CARDSW   DC    X'00'                *                                      A2415690
REPLSW   DC    X'03'                BIT 7-SWITCH 'WAIT FOR EXT.INT.'       A2415700
*                                   BIT 6-SWITCH 'WAIT FOR INP.CMD.'       A2415710
ERRSW    EQU   REPLSW               BIT 3-SWITCH 'CARD ERR.DETECTED'       A2415720
MESSW    EQU   REPLSW               BIT 2-SWITCH 'MESSAGE 2 TO 8'          A2415730
LABFCT   EQU   X'10'                                                       A2415740
ERRBIT   EQU   X'10'                *                                      A2415750
MESBIT   EQU   X'20'                *                                      A2415760
         END   INIT                                                        A2415770
         AOPTN CROSSREF                                                    A2500010
A25B     TITLE 'RELOCATING LOADER FOR CURRENT SYSTEMS SIMULATORS'          A2500020
RELLDR   START 28048                  *                                  A2500030
         USING LALPHA-1028,LBAZRG     *                                  A2500040
         SPACE 2                                                           A2500050
*********************************************************************** A2500060
*                                                                     * A2500070
*                  RELOCATING  LOADER  PROGRAM                        * A2500080
*                                                                     * A2500090
*                              FOR                                    * A2500100
*                                                                     * A2500110
*             IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620               * A2500120
*                                                                     * A2500130
*             ----------------------------------------                * A2500140
*                                                                     * A2500150
*  THE RELOCATING LOADER IS DESIGNED TO =                             * A2500160
*  1. RELOCATE AND LOAD ASSEMBLED  PROGRAMS  INTO  STORAGE LOCATIONS  * A2500170
*     OTHER THAN THOSE ASSIGNED BY THE ASSEMBLER.                     * A2500180
*  2. PERMITS CHANGES WITHIN THE ASSEMBLED PROGRAMS AT LOAD TIME.     * A2500190
*  3. PROVIDE LINKAGE BETWEEN SEPARATELY ASSEMBLED PROGRAMS.          * A2500200
*  4. REEVALUATE ADDRESS CONSTANTS USED BY THE ASSEMBLED PROGRAMS.    * A2500210
*  5. GENERATE, IF NEED BE, FROM PROGRAM LOADED, PROGRAM IN  A  FORM  * A2500220
*  SUITABLE FOR LOADING WITH THE SYSTEM/360 IPL PROCEDURE.            * A2500230
*                                                                     * A2500240
*  THE VERSION OF RELOCATING LOADER PROGRAM SUPPLIED BY IBM OCCUPIES  * A2500250
*  HIGH STORAGE LOCATIONS AND LOADS INTO MAIN STORAGE  IN  FRONT  OF  * A2500260
*  ITS OWN AREA. TO MODIFY THE LOADER TO ACCOMODATE  LARGER SYSTEMS,  * A2500270
*  THE USER MUST RE-ASSEMBLY THE RELOCATING LOADER  PROGRAM  AT  THE  * A2500280
*  LOCATION DESIRED.                                                  * A2500290
*                                                                     * A2500300
*                                 *****                               * A2500310
*                                                                     * A2500320
```

```
*  THE LOADER PROCESSES THE DECK FOR A PROGRAM MODULE IN   3 STAGES=  * A2500330
*      STAGE A   EXTERNAL SYMBOL DICTIONARY (ESD) CARDS.              * A2500340
*      STAGE B   TEXT (TXT) CARDS.                                    * A2500350
*      STAGE C   REPLACE (REP) AND                                    * A2500360
*                RELOCATION LIST DICTIONARY (RLD) CARDS.              * A2500370
*  ALL STAGE A  CARDS MUST PRECEDE STAGE B  CARDS  AND  ALL STAGE  B  * A2500380
*  CARDS MUST PRECEDE STAGE C  CARDS. EACH PROGRAM MODULE  IS TERMI-  * A2500390
*  NATED BY A LOADER END (END) CARD GENERATED BY THE ASSEMBLER.       * A2500400
*                                                                     * A2500410
*                                                          .../...    * A2500420
*                                                                     * A2500430
*********************************************************************** A2500440
         EJECT                                                         A2500450
*********************************************************************** A2500460
*                                                                     * A2500470
*  LOADING TABLES                                                     * A2500480
*  --------------                                                     * A2500490
*                                                                     * A2500500
*  THE FOLLOWING THREE TABLES ARE FORMED AND USED BY THE LOADER =     * A2500510
*                                                                     * A2500520
*  1. EXTERNAL SYMBOL DICTIONARY (LDDIC)                              * A2500530
*                                                                     * A2500540
*  THIS TABLE CONTAINS ONE ENTRY OF THREE FULLWORDS FOR EACH TERM IN  * A2500550
*  AN EXTERNAL SYMBOL DICTIONARY CARD =                               * A2500560
*      CONTROL SECTION NAME (TERM TYPE 1),                            * A2500570
*      ENTRY POINT (TERM TYPE 2),                                     * A2500580
*      EXTERNAL SYMBOL (TERM TYPE 3).                                 * A2500590
*                                                                     * A2500600
*      BYTES  0-5  CONTAIN  THE CONTROL SECTION NAME OR               * A2500610
*                           THE ENTRY POINT NAME OR                   * A2500620
*                           THE EXTERNAL SYMBOL NAME.                 * A2500630
*      BYTES  6-8     -     THE ORIGIN  OF  THE  CONTROL SECTION  AS   * A2500640
*                             DEFINED DURING LOADING (TYPE 1) OR      * A2500650
*                           THE ADDRESS OF THE ENTRY POINT AS         * A2500660
*                             DEFINED DURING LOADING (TYPE 2) OR      * A2500670
*                           24 ZEROS (TYPE 3).                        * A2500680
*      BYTES  9-10    -     THE NUMBER OF BYTES IN THE CONTROL  SEC-   * A2500690
*                             TION (TYPE 1) OR                        * A2500700
*                           THE NUMBER (ESID) ASSIGNED  TO  THE CON-  * A2500710
*                             TROL SECTION IN WHICH THE  ENTRY POINT  * A2500720
*                             OCCURS (TYPE 2) OR                      * A2500730
*                           16 ZEROS (TYPE 3).                        * A2500740
*      BYTE  11    CONTAINS THE TYPE OF ESD TERM (TYPES  1,2 AND 3)   * A2500750
*                                                                     * A2500760
*  STARTING ADDRESS OF LDDIC DICTIONARY IS CALLED  LDDICA             * A2500770
*  CURRENT     -         -         -          -     LDDICZ            * A2500780
*  INITIALLY THE DICTIONARY IS EMPTY AND  LDDICA= LDDICZ              * A2500790
*                                                                     * A2500800
*  2. EXTERNAL SYMBOL REFERENCE TABLE (LDREF)                         * A2500810
*                                                                     * A2500820
*  THIS TABLE CONTAINS ONE ENTRY OF TWO  FULLWORDS  FOR EACH TERM,IN  * A2500830
*  AN ESD CARD, OF THE FOLLOWING TYPES=                               * A2500840
*      CONTROL SECTION NAME (TERM TYPE 1)                             * A2500850
*      EXTERNAL SYMBOL (TERM TYPE 3)                                  * A2500860
*                                                                     * A2500870
```

```
*      BYTES  0-3  CONTAIN  THE ADDRESS OF THE  CORRESPONDING  ENTRY  * A2500880
*                            IN THE EXTERNAL SYMBOL DICTIONARY       * A2500890
*      BYTE   4    CONTAINS THE TYPE OF ESD TERM (TYPES 1 AND 3)      * A2500900
*      BYTES  5-6  CONTAIN  THE ADDRESS OF THE CONTROL SECTION    AS  * A2500910
*                            DEFINED BY THE ASSEMBLER (TYPE 1)   OR  * A2500920
*                           24 ZEROS (TYPE 3).                       * A2500930
*                                                                    * A2500940
*                                                          .../...   * A2500950
*                                                                    * A2500960
********************************************************************** A2500970
        EJECT                                                         A2500980
********************************************************************** A2500990
*                                                                   * A2501000
*  .../...                                                          * A2501010
*                                                                   * A2501020
*  STARTING ADDRESS OF LDREF TABLE IS CALLED  LDREFA                * A2501030
*  CURRENT     -        -    -       -      LDREFZ                  * A2501040
*  WHEN LOADING BEGINS THIS TABLE IS EMPTY AND LDREFA= LDREFZ.   THE * A2501050
*  LDREF TABLE IS CLEARED AT THE END OF EACH MODULE, I.E. EACH  TIME * A2501060
*  A LOAD END (END) CARD IS ENCOUNTERED.                            * A2501070
*                                                                   * A2501080
*  3. RELOCATION LIST (LDLST)                                       * A2501090
*                                                                   * A2501100
*  THE RELOCATION LIST CONTAINS ONE ENTRY OF TWO FULLWORDS FOR  EACH * A2501110
*  ITEM FROM RELOCATION LIST DICTIONARY (RLD) CARD  WHICH REFERS  TO * A2501120
*  A CONTROL SECTION NOT YET RELOCATED.                             * A2501130
*                                                                   * A2501140
*      BYTES  0-1  CONTAIN  THE POSITION HEADER= NUMBER  (ESID)  OF * A2501150
*                            THE CONTROL SECTION WHICH THE CONSTANT * A2501160
*                            ADDRESS BELONGS TO.                    * A2501170
*      BYTES  2-3     -     THE RELOCATION HEADER= NUMBER (ESID)  OF * A2501180
*                            THE CONTROL SECTION WHICH  THE  SYMBOL * A2501190
*                            APPEARING IN THE CONSTANT  BELONGS  TO * A2501200
*      BYTES  4    CONTAINS THE SIZE, COMPLEMENT   AND  CONTINUATION * A2501210
*                            FLAGS.                                 * A2501220
*      BYTES  5-7  CONTAIN  THE RELOCATED ADDRESS  OF  THE  CONSTANT * A2501230
*                            ADDRESS.                               * A2501240
*                                                                   * A2501250
*  STARTING ADDRESS OF LDLST  LIST IS CALLED  LDLSTA                * A2501260
*  CURRENT     -        -    -       -      LDLSTZ                  * A2501270
*  INITIALLY THE LIST IS EMPTY AND  LDLSTA= LDLSTZ.  EACH  TIME  THE * A2501280
*  LOADER ENDS A LOADING PROCESS (END CARD) IT SCANS THE LDLST  LIST * A2501290
*  AND FINISHES PROCESSING OF CONSTANT ADDRESSES WHICH REFER TO SYM- * A2501300
*  BOLS DEFINED IN THE PROGRAM MODULE LOADED.                       * A2501310
*                                                                   * A2501320
*  WHEN THE SELECTIVE LOADING FUNCTION IS USED THE RELOCATING LOADER * A2501330
*  CREATES A FOURTH TABLE (LDOPT). EACH TABLE ELEMENT (2  FULLWORDS) * A2501340
*  CONTAINS THE NAME OF A CONTROL SECTION THAT MUST  NOT  BE LOADED. * A2501350
*  THESE CONTROL SECTION NAMES FORM A  PART  OF  THE  PARAMETER LIST * A2501360
*  FROM THE INITIALIZATION PROGRAM.                                 * A2501370
*                                                                   * A2501380
*      BYTES  0-5  CONTAIN THE CONTROL SECTION NAME                 * A2501390
*      BYTES  6-7  ARE UNUSED                                       * A2501400
*                                                                   * A2501410
*  STARTING ADDRESS OF LDOPT TABLE IS CALLED  LDOPTA                * A2501420
```

```
*  CURRENT     -         -     -         -     LDOPTZ                  * A2501430
*  INITIALLY THE TABLE IS EMPTY AND  LDOPTA= LDOPTZ                    * A2501440
*                                                                      * A2501450
*                                                              .../... * A2501460
*                                                                      * A2501470
************************************************************************ A2501480
        EJECT                                                           A2501490
************************************************************************ A2501500
*                                                                      * A2501510
*  TABLES MAPPING                                                      * A2501520
*  --------------                                                      * A2501530
*                                                                      * A2501540
*  THE ABOVE FOUR TABLES ARE STORE AS SHOWN BELOW, IN FRONT  OF  THE   * A2501550
*  RELOCATING LOADER PROGRAM. THEY OVERLAY THE SELF LOADING  PROGRAM   * A2501560
*  GENERATOR AND INITIALIZATION ROUTINES IF  THE  GENERATOR FUNCTION    * A2501570
*  IS NOT TO BE USED.                                                  * A2501580
*                                                                      * A2501590
*  L           L   L         L   L           LL      L                 * A2501600
*  D           D   D         D   D           DD      D                 * A2501610
*  L           L   R         R   D           DO      O                 * A2501620
*  S           S   E         E   I           IP      P                 * A2501630
*  T           T   F         F   C           CT      T                 * A2501640
*  A           Z   A         Z   Z           AZ      A                 * A2501650
*  ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++    * A2501660
*  +           +   +         +   +           +       +       +         * A2501670
*  + RELOCATION+   + REFERENCE +   + DICTIONARY + LDOPT + LOADER +     * A2501680
*  +   LIST    +   +   TABLE   +   +           + TABLE + PROGR. +     * A2501690
*  +           +   +         +   +           +       +       +         * A2501700
*  ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++    * A2501710
*  +                                                       +          * A2501720
*  +---------------- LOADING TABLE BLOCK ----------------+            * A2501730
*                                                                      * A2501740
*  THE LDOPT TABLE IS FILLED BACKWARDS, THAT IS, THE 1ST. ELEMENT IS   * A2501750
*  CONTIGUOUS TO THE RELOCATING LOADER PROGRAM, AND THE TABLE IS EX-   * A2501760
*  TENDED TO THE FRONT AS NEW ELEMENTS ARE ADDED. LDOPT   IS CREATED   * A2501770
*  AT THE TIME THE LOADER IS INITIALIZED AND REMAINS UNCHANGED   DU-   * A2501780
*  RING LOADING PROCESS.                                               * A2501790
*  THE LOADING TABLE BLOCK IS CONTIGUOUS TO THE LDOPT TABLE. THE TA-   * A2501800
*  BLE BLOCK LENGTH IS ONE ITEM OF THE PARAMETER LIST FROM  THE INI-   * A2501810
*  TIALIZATION PROGRAM. INITIALLY  LDLSTA= LDLSTZ,                     * A2501820
*                                  LDDICA= LDDICZ = LDREFA = LDREFZ.   * A2501830
*                                                                      * A2501840
*  THE ABOVE FIGURE REPRESENTS THE TABLES DURING  THE  LOADING.  THE   * A2501850
*  POSITION OF REFERENCE TABLE IS FLEXIBLE AND IS ADJUSTED  TO MAIN-   * A2501860
*  TAIN THE TWO AREA OF UNUSED TABLE SPACE. ELEMENTS ARE  ENTERED IN   * A2501870
*  RELOCATION LIST  AND   REFERENCE TABLE  FROM LEFT  TO  RIGTH  BUT   * A2501880
*  FROM  RIGHT TO LEFT IN THE DICTIONARY.                              * A2501890
*                                                                      * A2501900
*                                 *****                                * A2501910
*                                                                      * A2501920
*  THE RELOCATING LOADER IS FUNCTIONALLY DIVISIBLE INTO  3  PARTS=     * A2501930
*      - SELF LOADING GENERATOR                                        * A2501940
*      - INPUT/OUTPUT OPERATIONS                                       * A2501950
*      - CARD  PROCESSING                                              * A2501960
*                                                                      * A2501970
```

```
*  THE RELOCATING LOADER PROGRAM IS DIVISED INTO SECTIONS WHICH  ARE  * A2501980
*  CLEARLY MARKED AND EXPLAINED ON THE LOADER ASSEMBLY LISTING.  THE  * A2501990
*  SECTIONS CALLED 'ROUTINES' THROUGHOUT THIS LISTING  ARE LOGICALLY  * A2502000
*  DISTINCT ENTITIES. THE SECTIONS CALLED 'SUBROUTINES' ARE  AVAILA-  * A2502010
*  BLE SUCH AS TO THE ROUTINES. EX= HEXADECIMAL-TO-BINARY SUBROUTINE  * A2502020
*                                                                     * A2502030
********************************************************************** A2502040
        EJECT                                                          A2502050
********************************************************************** A2502060
*                                                                    * A2502070
*                  SELF-LOADING PROGRAM GENERATOR ROUTINE            * A2502080
*                                                                    * A2502090
*                    NAME= LDEDIT                                    * A2502100
*                                                                    * A2502110
*  THIS ROUTINE HAS ONE ENTRY POINT, LOCATION LDEDIT. IT IS  ENTERED * A2502120
*  AT THE END OF THE LOADING PROCESS  WHICH  PLACES   THE  FOLLOWING * A2502130
*  OBJECT PROGRAMS IN STORAGE = CONTROL PROGRAM,                     * A2502140
*                               I/O SUPPORT PACKAGE PROGRAM,         * A2502150
*                               PROGRAM JUST LOADED.                 * A2502160
*  THE RELOCATING LOADER DECLARES  THE  FOLLOWING  INFORMATIONS  TO  * A2502170
*  LDEDIT =                                                          * A2502180
*   1. THE LOWEST STORAGE ADDRESS OCCUPIED  BY THE  CONTROL  PROGRAM * A2502190
*      (LOCATION LDAREA).                                            * A2502200
*   2. THE HIGHEST STORAGE ADDRESS PLUS 1  OCCUPIED  BY THE  PROGRAM * A2502210
*      LOADED (LOCATION LOCCTR).                                     * A2502220
*   3. THE PROGRAM INITIAL PSW (LOCATION LDTPSW).                    * A2502230
*  THE OUTPUT DEVICE (SYMBOLIC NAME AND TYPE) WAS PROVIDED BY  MEANS * A2502240
*  OF 2 CONTROL CARDS, DEVSUP AND CALL -SEE INITIALIZATION PROGRAM-. * A2502250
*                                                                    * A2502260
*  SEQUENCE OF OPERATIONS                                            * A2502270
*                                                                    * A2502280
*  1.  THE ROUTINE FIRST ESTABLISHES A PSW IMAGE FOR  THE  FIRST IPL * A2502290
*      CARD OR RECORD.                                               * A2502300
*  2.  THE ROUTINE CREATES AN IMAGE OF REGENERATED PROGRAMS=         * A2502310
*        - IT COPIES MAIN STORAGE BETWEEN LOCATIONS (LDAREA)  AND    * A2502320
*          (LOCCTR)-1 INTO MAIN STORAGE BEYOND LOCATION (LOCCTR).   * A2502330
*        - IT ESTABLISHES SEVERAL CONSTANTS FOR  A NEW   IBM 1052    * A2502340
*          PRINTER KEYBOARD IF THE 1052 USED UP TO NOW IS NOT THE   * A2502350
*          SAME AS THAT TO BE USED AFTERWARDS.                      * A2502360
*  IF THE OUTPUT DEVICE IS A CARD PUNCH THE ROUTINE PERFORMS  OPERA- * A2502370
*  RATION 4 . IF IT IS A 2400 MAGNETIC TAPE UNIT,                    * A2502380
*  3.  THE ROUTINE ISSUES A WRITE REQUEST FOR=                       * A2502390
*        - ONE 24-BYTE RECORD CONTAINING THE IPL RECORD (IPLPSW ,    * A2502400
*          IPLCCW).                                                 * A2502410
*        - ONE RECORD CONTAINING  THE  REGENERATED PROGRAMS, THAT    * A2502420
*          IS, THE CONTENTS OF (LOCCTR) THROUGH  2X(LOCCTR).        * A2502430
*      AND COMPLETES PROCESSING BY PERFORMING OPERATION  5.          * A2502440
*  4.  THE ROUTINE ISSUES A WRITE REQUEST FOR=                       * A2502450
*        - ONE 24-BYTE CARD CONTAINING THE IPL RECORD   (IPLPSW ,    * A2502460
*          IPLCCW'S).                                               * A2502470
*        - ONE 56-BYTE CARD CONTAINING A READ ROUTINE   (7  CCW'S    * A2502480
*          CHAINED).                                                * A2502490
*        - A SERIES OF 80-BYTE CARDS(TXT CARD FORMAT), CONTAINING    * A2502500
*          THE REGENERATED PROGRAM. THESE CARDS WILL BE SEQUENCED   * A2502510
*          IN COLUMNS 77-80  AND IDENTIFIED IN COLUMNS 73-76 WITH   * A2502520
```

```
*          THE FIRST 4 CHARACTERS OF THE OUTPUT DEVICE NAME.          * A2502530
*        - ONE END CARD WHICH WILL CAUSE THE READIND PROCESS          * A2502540
*          TO BE TERMINATED.                                          * A2502550
*  5.  PROCESSING IS COMPLETED AND TERMINATES IN THE WAIT STATE.      * A2502560
*                                                                     * A2502570
*                                                            .../...  * A2502580
*                                                                     * A2502590
*********************************************************************** A2502600
         EJECT                                                        A2502610
*********************************************************************** A2502620
*                                                                     * A2502630
*  .../...                                                            * A2502640
*                                                                     * A2502650
*  EXITS                                                              * A2502660
*                                                                     * A2502670
*  THIS ROUTINE HAS NO EXITS, AS SUCH. PROCESSING TERMINATES IN  ONE  * A2502680
*  OF TWO WAIT STATES. THESE WAIT STATES, AND THEIR ASSOCIATED  CON-  * A2502690
*  SOLE MESSAGES, ARE=                                                * A2502700
*    NORMAL END OF JOB            'END OF INITIALIZATION'             * A2502710
*    INSUFFICIENT SPACE BEYOND    'INITIALIZATION ERROR, CANNOT       * A2502720
*    LOCATION  (LOCCTR)            CONTINUE'                          * A2502730
*                                                                     * A2502740
*********************************************************************** A2502750
         SPACE                                                        A2502760
LDEDIT   MVC   IPLPSW(8),LDTPSW          SET-UP PSW FOR IPL 1 RECORD A2502770
         LM    LWK1RG,LWK3RG,LDAREA      START AND END ADDR. OF AREA A2502780
         LM    LDCONT,LDFINL,LDAREA      OCCUPIED BY PROG. GENERATED A2502790
         AR    LDFINL,LDCONT             START ADDR. OF AREA USED TO A2502800
         BCTR  LWK3RG,0                  CREATE PROGRAM IMAGE.       A2502810
LDEDIA   BXH   LWK1RG,LWK2RG,LDEDIC      BRANCH TO ADJUST LAST MOVE. A2502820
         EX    LWK2RG,LDEDIB             PLACE PROGR. IN IMAGE AREA. A2502830
         AR    LDCONT,LWK2RG             UPDATE 'FROM' ADDRESS       A2502840
         AR    LDFINL,LWK2RG             AND 'TO' ADDRESS.           A2502850
         C     LDFINL,LEDMAX             IS ENOUGH ROOM FOR PROGRAM. A2502860
         BC    12,LDEDIA                 IMAGE. YES, BRANCH - NO,    A2502870
         LA    LEXIT2,MSDG03             PRINT 'INITIALIZAT. ERROR'  A2502880
         BC    15,LDSTOP+4               AND ENTER THE WAIT STATE.   A2502890
LDEDIB   MVC   0(1,LDFINL),0(LDCONT)     *                           A2502900
LDEDIC   LR    LWK1RG,LDCONT             ADJUST BYTE COUNT TO MOVE   A2502910
         BCT   LWK2RG,LDEDIA             LAST BYTES OF PROGRAM.      A2502920
         LH    LWK2RG,LD1052+2           DOES THE GENERATED PROGRAM  A2502930
         LTR   LWK2RG,LWK2RG             USE A 1052 OTHER THAN THAT  A2502940
         BC    8,*+24                    USED NOW. NO,BRANCH - YES,  A2502950
         LM    LWK2RG,LWK3RG,LCONSL      CALCULATE ADDRESSES OF TWO  A2502960
         AR    LWK2RG,LWK1RG             ENTRY POINTS TO THE CONTROL A2502970
         AR    LWK3RG,LWK1RG             PROGRAM (IMAGE).            A2502980
         MVC   1(2,LWK3RG),LD1052+2      PLACE 1052 DEVICE ADDRESS   A2502990
         MVC   1(4,LWK2RG),LD1052+4      AND UNIT CONTROL BLOCK ADD. A2503000
         LM    LDCONT,LDFINL,LDAREA      RESTORE REGISTERS.          A2503010
         LH    LEXIT2,LEDDEV             DETERMINE TYPE OF OUTPUT    A2503020
         LTR   LEXIT2,LEXIT2             DEVICE.                     A2503030
         BC    8,LDEDI2                  CARD PUNCH, BRANCH- 2400TU, A2503040
         SPACE                                                        A2503050
LDEDI1   BAL   LEXIT1,LDREW1             REWIND MAGNETIC T.U.        A2503060
         MVC   IPLCCW+1(3),LDAREA+1      CCW FOR IPL RECORD=ADDRESS, A2503070
```

```
         SR    LDFINL,LDCONT          *                          A2503080
         STH   LDFINL,IPLCCW+6        COUNT,                     A2503090
         MVI   IPLCCW+4,X'20'         COMMAND FLAG (SLI).        A2503100
         LA    LEXIT3,IPLPSW-1        ADDRESS AND LENGTH OF IPL  A2503110
         LA    LEXIT2,24              RECORD.                    A2503120
         BAL   LEXIT1,LWRITE          ISSUE WRITE REQUEST.       A2503130
         LR    LEXIT3,LWK1RG          ADDRESS OF AREA CONTAINING A2503140
         AR    LEXIT3,LDCONT          PROGRAM REGENERATED  AND   A2503150
         LH    LEXIT2,IPLCCW+6        LENGTH OF THIS RECORD.     A2503160
         BAL   LEXIT1,LWRITE          ISSUE WRITE REqUEST.       A2503170
         LH    LEXIT2,LEDDEV          TAPE UNIT DEVICE ADDRESS.  A2503180
         BAL   LEXIT1,LDREW1          REWIND.                    A2503190
         BC    15,LDEDI8              OPERATION COMPLETED.       A2503200
         SPACE                                                   A2503210
LDEDI2   LA    LWK1RG,1(LWK1RG)       INCREMENT LOCATION COUNTER A2503220
         ST    LWK1RG,LOCCTR          BY 1 AND ADJUST IT TO  A   A2503230
         TM    LOCCTR+3,X'07'         DOUBLE WORD BOUNDARY.      A2503240
         BC    5,LDEDI2               *                          A2503250
         BAL   LEXIT1,LDEDID          CLEAR CARD READER-PUNCH    A2503260
         MVC   IPLCCW+1(3),LOCCTR+1   SET LOADING ADDR. OF READ  A2503270
         MVC   IPLCCW+9(3),LOCCTR+1   ROUTINE IN IPL RECORD.     A2503280
         MVC   IPLCW7+1(3),LOCCTR+1   ESTABLISH ONE ADDRESS IN   A2503290
         MVC   IPLEND+5(3),LOCCTR+1   THE END RECORD AND SEVERAL A2503300
         LA    LWK2RG,41(LWK1RG)      ADDR. IN THE READ ROUTINE. A2503310
         ST    LWK2RG,LOCCTR          THIS ROUTINE IS IN THE 2ND A2503320
         MVC   IPLCW2+1(3),LOCCTR+1   RECORD WHICH WILL BE LOADED A2503330
         LA    LWK2RG,5(LWK2RG)       FROM THE IPL PROCEDURE.    A2503340
         ST    LWK2RG,LOCCTR          *                          A2503350
         MVC   IPLCW4+1(3),LOCCTR+1   *                          A2503360
         SR    CRDCNT,CRDCNT          RESET CARD COUNTING REG.   A2503370
         MVC   EDBUFF(24),IPLPSW      PLACE IPL RECORD IN OUTPUT A2503380
         BAL   LEXIT1,LDEDI9          BUFFER AND ISSUE WRITE REQ. A2503390
         MVC   EDBUFF(56),IPLCW1      PLACE READ RTNE. RECORD IN A2503400
         BAL   LEXIT1,LDEDI9          OUTPUT BUFFER AND WRITE IT. A2503410
         MVC   EDBUFF(16),EDCARD      SET UP OUTPUT BUFFER  TO   A2503420
         LR    LWK1RG,LDFINL          WRITE TXT CARD. CALCULATE  A2503430
         AR    LWK1RG,LDCONT          ADDR. OF PROG. IMAGE AREA. A2503440
LDEDI3   ST    LDCONT,EDBUFF+4        LOADING ADDRESS AND LENGTH A2503450
LDEDI4   STH   LDSTEP,EDBUFF+10       OF TEXT PUNCHED IN CARD.   A2503460
         BXLE  LDCONT,LDSTEP,LDEDI6   LAST TXT CARD. NO, BRANCH. A2503470
         L     LDCONT,EDBUFF+4        YES, ADJUST THE BYTE COUNT A2503480
         BCT   LDSTEP,LDEDI4          FOR THE LAST TXT CARD.     A2503490
         BC    15,LDEDI7              ENTIRE PROGRAM REGENERATED. A2503500
LDEDI5   MVC   EDBUFF+16(1),0(LWK1RG) *                          A2503510
LDEDI6   MVI   EDBUFF+4,X'40'         SET BLANK CHAR. IN COL.5.  A2503520
         LR    LWK2RG,LDSTEP          *                          A2503530
         BCTR  LWK2RG,0               *                          A2503540
         EX    LWK2RG,LDEDI5          *                          A2503550
         AR    LWK1RG,LDSTEP          TO OUTPUT BUFFER AND GO TO A2503560
         BAL   LEXIT1,LDEDI9          WRITE AN 80-BYTE TXT CARD. A2503570
         MVI   EDBUFF+16,X'40'        CLEAR OUTPUT BUFFER  (TEXT A2503580
         MVC   EDBUFF+17(55),EDBUFF+16 PART) AND RESUME GENERATING A2503590
         BC    15,LDEDI3              PROCESS.                   A2503600
LDEDI7   MVC   EDBUFF(24),IPLEND      PLACE END RECORD IN OUTPUT A2503610
         BAL   LEXIT1,LDEDI9          BUFFER AND GO TO WRITE IT. A2503620
```

```
         BAL   LEXIT1,LDEDID              EJECT LAST CARD PUNCHED     A2503630
         SPACE                                                        A2503640
LDEDI8   LA    LEXIT2,MSDG02              OPERATION COMPLETED. PRINT  A2503650
         BC    15,LDSTOP+4                'END OF INIT.' AND STOP.    A2503660
         EJECT                                                        A2503670
LDEDID   MVI   EDBUFF,X'40'               CLEAR OUTPUT BUFFER         A2503680
         MVC   EDBUFF+1(79),EDBUFF        *                           A2503690
         BC    15,LDEDIE                  GO TO PUNCH A BLANK CARD    A2503700
         SPACE                                                        A2503710
LDEDI9   BAL   LEXIT3,LBDEC1              NUMBER CARD (COL.77-80).    A2503720
         MVC   EDBUFF+72(4),WRTSUP        IDENTIFY CARD (COL.73-76).  A2503730
LDEDIE   LA    LEXIT2,80                  LENGTH AND                  A2503740
         LA    LEXIT3,WRTBUF              ADDRESS OF OUTPUT BUFFER.   A2503750
         SPACE 2                                                      A2503760
*************************************************************************A2503770
*                                                                     * A2503780
*         WRITE SUBROUTINE TO GENERATE SELF-LOADING PROGRAM           * A2503790
*                                                                     * A2503800
*                  NAME= LDEDIT                                       * A2503810
*                                                                     * A2503820
*         CALLING SEQUENCE  LA  LEXIT3,OUTPUT BUFFER ADDRESS          * A2503830
*                           LA  LEXIT2,  -    RECORD LENGTH           * A2503840
*                           BAL LEXIT1,LWRITE                         * A2503850
*                                                                     * A2503860
*  THIS ROUTINE PUNCHES ONE CARD OR WRITES ONE  RECORD  ON  MAGNETIC  * A2503870
*  TAPE AND TESTS   THE  STATUS  SET  BY THE  I/O PACKAGE PROGRAM IN  * A2503880
*  FIRST LOCATION OF THE OUTPUT BUFFER.                               * A2503890
*  IF THE OPERATION HAS BEEN SUCCESSFULL, CONTROL RETURN TO THE CAL-  * A2503900
*  LER. IF NOT THE CONDITION IS  BEYOND  CORRECTION  AND  GENERATING  * A2503910
*  PROCESS STOPS.                                                     * A2503920
*                                                                     * A2503930
*  NOTE- SEE WRITE MESSAGE ROUTINE, NOTE 1.                           * A2503940
*                                                                     * A2503950
*************************************************************************A2503960
         SPACE                                                        A2503970
LWRITE   ST    LEXIT3,LWRIT2              ADDRESS OF OUTPUT BUFFER.   A2503980
         STH   LEXIT2,LWRIT1              LENGTH OF RECORD.           A2503990
         CNOP  0,4                        I/O REQUEST.                A2504000
         SVC   18                         * (WRITE)                   A2504010
WRTSUP   DC    C'        '                * OUTPUT SUPPORT FUNCTION   A2504020
LWRIT1   DC    FL2'0'                     *    -   RECORD LENGTH      A2504030
LWRIT2   DC    A(0)                       *    -   BUFFER ADDRESS     A2504040
         TM    0(LEXIT3),X'07'            * ERROR TYPE                A2504050
         BCR   1,LEXIT1                   * NORMAL RETURN             A2504060
         TM    0(LEXIT3),X'03'            * EXCEPTIONAL RETURN        A2504070
         BCR   1,LEXIT1                   EOB,RETURN TO CALLER.       A2504080
         BC    15,LDWAIT                  CATASTROPHIC ERROR, STOP.   A2504090
         SPACE 2                                                      A2504100
*************************************************************************A2504110
*                                                                     * A2504120
*                          CONSTANTS  AREA                            * A2504130
*                                                                     * A2504140
*************************************************************************A2504150
         SPACE                                                        A2504160
*                                    IPL RECORD =                     A2504170
```

```
IPLPSW   DS    D                        IPL PSW,                          A2504180
IPLCCW   CCW   X'02',*,X'60',56         CCW TO READ 2ND RECORD,           A2504190
         CCW   X'08',*,X'00',0          TIC TO READ ROUTINE (CARDS)       A2504200
         EJECT                                                            A2504210
*                                       READ ROUTINE (CARD  VERSION       A2504220
*                                       ONLY) IN 2ND RECORD.              A2504230
IPLCW1   CCW   X'02',*,X'B0',5          SKIP 5 BYTES (COL.  1-5 )         A2504240
IPLCW2   CCW   X'02',*,X'A0',3          READ 3   -   (LOADING ADD.)       A2504250
IPLCW3   CCW   X'02',*,X'B0',2          SKIP 2   -   (COL.  9-10)         A2504260
IPLCW4   CCW   X'02',*,X'A0',2          READ 2   -   (TEXT LENGTH)        A2504270
IPLCW5   CCW   X'02',*,X'B0',4          SKIP 4   -   (COL. 13-16)         A2504280
IPLCW6   CCW   X'02',*,X'60',0          READ N   -   (LOAD TEXT)          A2504290
IPLCW7   CCW   X'08',*,X'00',0          TIC  TO  IPLCW1.                  A2504300
         SPACE                                                            A2504310
*                                       TEXT CARD                         A2504320
EDCARD   DC    X'02'                    COL.  1    (12-2-9)               A2504330
         DC    C'TXT'                   COL.  2-4  (TXT)                  A2504340
         DC    A(0)                     COL.  6-8  (LOADING ADDR.)        A2504350
         DC    C'  '                    COL.  9-10                        A2504360
         DC    FL2'0'                   COL. 11-12 (BYTE  COUNT)          A2504370
         DC    C'  '                    COL. 13-14                        A2504380
         DC    X'0001'                  COL. 15-16 (PROGRAM NUMBER)       A2504390
         SPACE                                                            A2504400
*                                       END CARD                          A2504410
IPLEND   DC    X'02'                    COL.  1    (12-2-9)               A2504420
         DC    C'END '                  COL.  2-5  (END )                 A2504430
         DC    AL3(0)                   COL.  6-8  (LOADING ADDR.)        A2504440
         DC    C'  '                    COL.  9-10                        A2504450
         DC    FL2'8'                   COL. 11-12 (BYTE COUNT)           A2504460
         DC    C'  '                    COL. 13-14                        A2504470
         DC    X'0001'                  COL. 15-16 (PROGRAM NUMBER)       A2504480
         CCW   X'03',*,X'20',8          COL. 17-24 (TEXT,STOP READ)       A2504490
         SPACE                                                            A2504500
LD1052   DC    A(0)                     UNIT DEVICE,UCB ADDR. OF          A2504510
         DC    A(0)                     1052 USED BY GENERAT. PROG.       A2504520
         SPACE                                                            A2504530
LEDMAX   DC    A(LDEDIA-56)             *                                 A2504540
         SPACE                                                            A2504550
*********************************************************************** A2504560
*                                                                     * A2504570
*                          CONSOLE  MESSAGES                          * A2504580
*                                                                     * A2504590
*********************************************************************** A2504600
         SPACE                                                          A2504610
MSDG02   DC    FL1'28'                  MEANING= NORMAL END OF JOB        A2504620
         DC    AL3(*+3)                                                  A2504630
         DC    C' AX12A'                                                 A2504640
         DC    C' END OF INITIALI'                                       A2504650
         DC    C'ZATION'                                                 A2504660
         DC    X'15'                                                     A2504670
MSDG03   DC    FL1'43'                  MEANING= JOB TERMINATED,NOT       A2504680
         DC    AL3(*+3)                 ENOUGH ROOM BEYOND LOCATION       A2504690
         DC    C' AX13W'                (LOCCTR)                          A2504700
         DC    C' INITIALIZATION '                                       A2504710
         DC    C'ERROR,CANNOT CON'                                       A2504720
```

```
         DC    C'TINUE'                                                  A2504730
         DC    X'15'                                                     A2504740
         EJECT                                                           A2504750
*****************************************************************       A2504760
*                                                               *       A2504770
*                  LOADER INITIALIZATION ROUTINE                *       A2504780
*                                                               *       A2504790
*                NAME= LDRIN1                                   *       A2504800
*                                                               *       A2504810
*  THIS ROUTINE IS EXECUTED ONCE ONLY  DURING  ANY  PERIOD OF LOADER    *A2504820
*  RESIDENCE, AT THE TIME THE LOADER IS FIRST EXECUTED.  THE STORAGE    *A2504830
*  AREA OCCUPIED BY THIS ROUTINE IS USED AS INPUT/OUTPUT BUFFER  DU-    *A2504840
*  RING THE LOADING PROCESS.                                    *       A2504850
*                                                               *       A2504860
*  THE ROUTINE, FROM THE LIST OF PARAMETERS PREPARED  BY THE INITIA-    *A2504870
*  LIZATION PROGRAM, DETERMINES=                                *       A2504880
*                          THE LENGTH OF LOADING TABLE BLOCK.   *       A2504890
*                          WHETHER THE PROGRAM IS TO BE  REGENERATED    *A2504900
*  AFTER LOADING, AND, IF SO, SAVES=                            *       A2504910
*    - THE ADDRESS AND SYMBOLIC NAME OF OUTPUT DEVICE,          *       A2504920
*    - EVENTUALLY, THE DEVICE AND UNIT CONTROL BLOCK ADDRESSES OF  A     *A2504930
*       NEW 1052 PRINTER.                                       *       A2504940
*                          WHETHER THE LOADER WILL PRINT THE LOADING    *A2504950
*  CONTROL MESSAGES.                                            *       A2504960
*                                                               *       A2504970
*  THE ROUTINE THEN CREATES THE LDOPT TABLE FROM THE SAME  PARAMETER    *A2504980
*  LIST.                                                        *       A2504990
*                                                               *       A2505000
*  FINALLY, IT INITIALIZES THE STARTING AND CURRENT ADDRESSES OF THE    *A2505010
*  LOADING TABLES.                                              *       A2505020
*                                                               *       A2505030
*  THIS ROUTINE IS ENTERED AT THE END OF THE PROGRAM  CHECK  ROUTINE    *A2505040
*  INITIALIZATION AND EXITS TO THE LOADER ENTRIES (LOCATION INIENT).    *A2505050
*                                                               *       A2505060
*****************************************************************       A2505070
         SPACE                                                           A2505080
         USING LALPHA,LBASRG                                             A2505090
LDRIN1   LR    LINKR1,1                 GR.1 = PARAMETER LIST REG.      A2505100
         L     LINKR2,8(LINKR1)         IS LENGTH OF LOADING            A2505110
         LTR   LINKR2,LINKR2            * TABLE BLOCK GIVEN             A2505120
         BC    8,*+8                    NO,BRANCH - YES,                A2505130
         ST    LINKR2,TBSIZE            SAVE TABLE BLOCK LENGTH         A2505140
         MVC   LEDDEV(4),20(LINKR1)     EDIT,LOAD DEVICES (ADDR.)       A2505150
         CLI   12(LINKR1),X'40'         IS EDIT FUNCTION REQUIRED       A2505160
         BC    8,LDRIN2                 NO,BRANCH - YES,                A2505170
         OI    LDLDT6+1,X'F0'           MODIFY LDT SUBROUTINE(EXIT)     A2505180
         MVC   LD1052(8),0(LINKR1)      1052 DEV. AND UCB ADDR.         A2505190
         MVC   WRTSUP(8),12(LINKR1)     EDIT SUPPORT FUNCTION(NAME)     A2505200
         ST    LBAZRG,ADBASE            MODIFY....                      A2505210
LDRIN2   L     LWK1RG,ADBASE            ADDR OF END OF TABLE BLOCK      A2505220
         ST    LWK1RG,LDOPTA            INITIALIZE START AND END OF     A2505230
         ST    LWK1RG,LDOPTZ            * OPTIONAL C.S. TABLE           A2505240
         LA    LINKR2,16(LINKR1)        ADDR. OF OPTIONAL C.S. LIST     A2505250
         LH    LWK1RG,LWK3RG,LDOPTA     INITIALIZE OPT.TBLE LOOK-UP     A2505260
LDRIN3   AR    LWK1RG,LWK2RG            ADDR.OF ENTRY IN LDOPT TBLE     A2505270
```

```
LDRIN4   LA    LINKR2,8(LINKR2)          GO TO NEXT LIST ELEMENT     A2505280
         TM    0(LINKR2),X'03'           EXAMINE CONTR. SECT. LIST   A2505290
         BC    8,LDRIN5                  BR. IF C.S. LIST EXHAUSTED  A2505300
         BC    1,LDRIN4                  BR. IF C.S. TO BE LOADED    A2505310
         MVC   0(6,LWK1RG),2(LINKR2)     STORE IN LDOPT TABLE THE    A2505320
         BC    15,LDRIN3                 * NAME OF C.S.TO BE SKIPPED A2505330
LDRIN5   ST    LWK1RG,LDOPTZ             UPDATE ADDR.OF END OF LDOPT A2505340
         MVC   LOCCTR(4),4(LINKR2)       LOCATION COUNTER VALUE      A2505350
         MVC   LCONSL(8),16(LINKR2)      ADDR. OF ENTRY TO CTRL.PRG. A2505360
         NC    LDLIST(1),TBSIZE          INITIALIZE LPRINT TO WRITE  A2505370
         NI    TBSIZE,X'00'              * LOADING MESSAGES OR NOT   A2505380
         ST    LWK1RG,LDREFA             INITIALIZE START OF REF.    A2505390
         ST    LWK1RG,LDDICA             * TABLE AND START AND END   A2505400
         ST    LWK1RG,LDDICZ             * OF DICTIONARY             A2505410
         S     LWK1RG,TBSIZE             SUBTRACT TABLE BLOCK LENGTH A2505420
         SRL   LWK1RG,2                  ROUND OFF TO FULLWORD       A2505430
         SLL   LWK1RG,2                  *                           A2505440
         ST    LWK1RG,LDTOPP             ADDR OF START OF TABLE BLCK A2505450
         ST    LWK1RG,LDLSTA             INITIALIZE START AND END OF A2505460
         ST    LWK1RG,LDLSTZ             * TABLE 'LIST'              A2505470
         NI    LDSWS,X'00'               RESET CONTROL FLAGS         A2505480
         BC    15,INIENT                 BR. TO INITIAL ENTRY POINT  A2505490
         SPACE 2                                                     A2505500
*************************************************************************A2505510
*                                                                       * A2505520
*                PROGRAM CHECK ROUTINE (INITIALIZATION)                 * A2505530
*                                                                       * A2505540
*                    NAME= LENTRY                                       * A2505550
*                                                                       * A2505560
*  THIS ROUTINE SETS PARAMETERS FOR  PROGRAM  INTERRUPTION.  WHEN  A    * A2505570
*  PROGRAM INTERRUPT OCCURS, THE CONTROL PROGRAM WILL BE ENTERED. IT    * A2505580
*  WILL PLACE  THE PROGRAM OLD PSW IN  THE  DOUBLE WORD  AT LOCATION    * A2505590
*  PRGPSW AND RETURN CONTROL TO THE INSTRUCTION AT LOCATION  LPRGCK.    * A2505600
*                                                                       * A2505610
*************************************************************************A2505620
         SPACE                                                      A2505630
LENTRY   BALR  LBASRG,0                  INITIALIZE BASE REGISTER   A2505640
LBEGIN   L     LBASRG,ADBASE-LBEGIN(0,LBASRG) *                     A2505650
         L     LBAZRG,ADBAZE              *                         A2505660
         CNOP  2,8                        * SET CONTROL PROGRAM FOR A2505670
         SVC   6                          PROGRAM INTERRUPTION      A2505680
         DC    A(LPRGCK)                  RETURN ADDRESS            A2505690
PRGPSW   DS    D                          PROGRAM OLD PSW           A2505700
         BC    15,LDRIN1                  GO TO LOADER INITIALIZATION A2505710
         EJECT                                                      A2505720
*************************************************************************A2505730
*                                                                       * A2505740
*                        PROGRAM CHECK ROUTINE                          * A2505750
*                                                                       * A2505760
*                    NAME= LPRGCK                                       * A2505770
*                                                                       * A2505780
*************************************************************************A2505790
         SPACE                                                      A2505800
LPRGCK   BAL   LEXIT1,LMSDG1             PRINT 'PROGRAM ERROR' AND  A2505810
         DC    C'1'                       *  STOP                    A2505820
```

```
         DC    AL3(MSDG0A)                 (OLD PROGR.PSW AT PRGPSW)   A2505830
         SPACE 2                                                       A2505840
*************************************************************************A2505850
*                                                                      * A2505860
*               STOP (END OF LOADING OWING TO ERROR)                   * A2505870
*                                                                      * A2505880
*                  NAME= LDSTOP                                        * A2505890
*                                                                      * A2505900
*************************************************************************A2505910
         SPACE                                                         A2505920
LDSTOP   LA    LEXIT2,MSDG01               CONSOLE MESSAGE 'LOADING    A2505930
         BAL   LEXIT1,LDCSL1               * ERROR,CANNOT CONTINUE'    A2505940
         CNOP  2,4                         *                           A2505950
LDWAIT   SVC   3                           LOAD WAIT PSW (I/O,EXTERNAL A2505960
         DC    A(STOPSW)                   * INTERRUPTS DISABLED)      A2505970
         EJECT                                                         A2505980
*************************************************************************A2505990
*                                                                      * A2506000
*                      CONSOLE MESSAGE ROUTINE                         * A2506010
*                                                                      * A2506020
*                  NAME = LDCSL1                                       * A2506030
*                                                                      * A2506040
*     CALLING SEQUENCE LA  LEXIT2,MSDG0X (X DEPENDS ON THE MESSAGE)    * A2506050
*                      BAL LEXIT1,LDCSL1                               * A2506060
*                                                                      * A2506070
*************************************************************************A2506080
         SPACE                                                         A2506090
LDCSL1   MVC   LDCSL2+2(4),0(LEXIT2)       SET UP WRITE SEQUENCE       A2506100
         MVC   4(3,LEXIT2),MSDGCX          MESSAGE CODE                A2506110
         CNOP  2,4                         *                           A2506120
LDCSL2   SVC   4                           WRITE MESSAGE               A2506130
         DC    A(0)                        LENGTH AND ADDR. OF MESSAGE A2506140
         TM    0(LEXIT2),X'07'             TEST STATUS BYTE            A2506150
         BC    0,*-4                       BR IF WRITE IN PROGRESS     A2506160
         BCR   1,LEXIT1                    BR IF WRITE FINISHED AND OK A2506170
         BC    15,LDWAIT                   BR IF WRITE NOT OK          A2506180
         SPACE 2                                                       A2506190
*************************************************************************A2506200
*                                                                      * A2506210
*          CALL  WRITE  ROUTINE FOR LOADING CONTROL MESSAGES           * A2506220
*                                                                      * A2506230
*                  NAME= LMSDG1                                        * A2506240
*                                                                      * A2506250
*          CALLING SEQUENCE  BAL LEXIT1,LMSDG1 (OR LPRINT)             * A2506260
*                            DC  C'CONTROL CHARACTER'                  * A2506270
*                            DC  AL3 (ADDRESS OF MESSAGE)              * A2506280
*                                                                      * A2506290
*  THIS ROUTINE IS ENTERED TO WRITE ANY  WARNING  OR  ERROR  LOADING   * A2506300
*  CONTROL MESSAGE. TO WRITE AN INFORMATIVE MESSAGE THE ROUTINE MUST   * A2506310
*  BE ENTERED AT LOCATION LPRINT. THEN THE BYTES WITH ADDRESSES        * A2506320
*            LDXBUF+72,.....LDXBUF+80                                  * A2506330
*  WILL BE WRITTEN WITHOUT ANY MODIFICATION.                           * A2506340
*                                                                      * A2506350
*************************************************************************A2506360
         SPACE                                                         A2506370
```

```
LMSDG1   CLC   LDXBUF+72(7),LDXBUF+73      COL.72-80 ARE BLANKS          A2506380
         BC    6,LMSDG2                   NO,BRANCH                     A2506390
         BAL   LEXIT3,LBDEC2              YES,BRANCH,CONVERT CARD CNT   A2506400
LMSDG2   OI    LDERR,X'02'                SET SWITCH 'ERROR FOUND' ON   A2506410
         EJECT                                                          A2506420
*********************************************************************** A2506430
*                                                                     * A2506440
*                        WRITE  MESSAGE ROUTINE                       * A2506450
*                                                                     * A2506460
*                      NAME= LPRINT                                   * A2506470
*                                                                     * A2506480
*  THIS ROUTINE WRITES MESSAGES ON AN OUTPUT DEVICE(PRINTER,MAGNETIC  * A2506490
*  TAPE,CONSOLE TYPEWRITER)AND TESTS THE STATUS SET BY THE IOPACKAGE  * A2506500
*  PROGRAM IN FIRST LOCATION OF THE OUTPUT BUFFER.                    * A2506510
*                                                                     * A2506520
*  IF THE MESSAGE HAS BEEN WRITTEN SUCCESSFULLY, CONTROL  RETURNS TO  * A2506530
*  THE CALLER. IF NOT THE CONDITION IS BEYOND CORRECTION AND LOADING  * A2506540
*  STOPS.                                                             * A2506550
*                                                                     * A2506560
*  NOTE 1 - IN THE  INPUT/OUTPUT BUFFERS  THE FIRST  BYTE IS A BLANK  * A2506570
*  RESERVED FOR STATUS BITS TO BE SET BY THE IOPACKAGE PROGRAM.THUS,  * A2506580
*  THE NUMBER OF BYTES OF DATA SPECIFIED IN THE SVC CALLING SEQUENCE  * A2506590
*  IS 1 LESS THAN THE LENGTH OF BUFFER.  THE  DATA  WILL  BE FETCHED  * A2506600
*  FROM OR PLACED IN LOCATIONS                                        * A2506610
*              BUFFER+1,BUFFER+2,....BUFFER+N                         * A2506620
*  NOTE 2 - FOR A PRINTING OPERATION (IBM 1403,1443),THE   CHARACTER  * A2506630
*  IN LOCATION BUFFER+1  IS A CONTROL CHARACTER SPECIFYING THE  TYPE  * A2506640
*  OF PRINT COMMAND                                                   * A2506650
*              CHARACTER '1'          WRITE AND SKIP  TO  CHANNEL  1  * A2506660
*                                     AFTER PRINTING                 * A2506670
*              ANY CHARACTER BUT '1'  WRITE AND SPACE  1  LINE AFTER  * A2506680
*                                     PRINTING                       * A2506690
*  THE DATA WILL BE FETCHED FROM LOCATIONS                            * A2506700
*              BUFFER+2,.............BUFFER+N                         * A2506710
*                                                                     * A2506720
*********************************************************************** A2506730
         SPACE                                                          A2506740
LPRINT   TM    LDLIST,X'01'               IS MESSAGE REQUESTED          A2506750
         BC    12,4(LEXIT1)               NO,BRANCH - YES,CONTINUE      A2506760
         MVC   OUTBUF+1(1),0(LEXIT1)      CONTROL CHARACTER (PRINTER)   A2506770
         MVC   OUTBUF+9(3),1(LEXIT1)      *MESSAGE ADDRESS (TEXT)       A2506780
         L     LEXIT2,OUTBUF+8            *                             A2506790
         MVC   OUTBUF+8(8),PRTBUF         SYMBOL OR CARD IDENTIF.       A2506800
         IC    LEXIT3,3(LEXIT2)           *MESSAGE LENGTH (TEXT)        A2506810
         BCTR  LEXIT3,0                   *                             A2506820
         EX    LEXIT3,LPRIN2              MESS.TEXT TO OUTBUF BUFFER    A2506830
         MVC   OUTBUF+4(3),0(LEXIT2)      *MESSAGE CODE                 A2506840
         CNOP  0,4                        I/O REQUEST                   A2506850
         SVC   18                         * (WRITE MESSAGE)             A2506860
         DC    C'SIM2PRNT'                * OUTPUT SUPPORT FUNCTION     A2506870
LPRIN1   DC    FL2'64'                    * OUTPUT RECORD LENGTH        A2506880
         DC    A(OUTBUF)                  * OUTPUT BUFFER ADDRESS       A2506890
         TM    OUTBUF,X'07'               * ERROR TYPE                  A2506900
         BC    12,LPRERR                  * ERROR RETURN                A2506910
         MVC   MESTXT(47),MESTXT-1        * CLEAR BUFFER (TEXT PART)    A2506920
```

```
         BC    15,4(LEXIT1)              * NORMAL RETURN              A2506930
LPRIN2   MVC   MESTXT(1),4(LEXIT2)                                    A2506940
LPRERR   EQU   LDWAIT                    ERROR RETURN                 A2506950
         EJECT                                                        A2506960
*************************************************************************A2506970
*                                                                      * A2506980
*          GET CARD IMAGE ROUTINE  EXCEPTIONAL RETURN                  * A2506990
*                                                                      * A2507000
*                  NAME= LDREAD                                        * A2507010
*                                                                      * A2507020
*  THIS ROUTINE ANALYZES THE STATUS SET BY THE IOPACKAGE PROGRAM  IN   * A2507030
*  LOCATION INPBUF IF ANY DIFFICULTIES HAVE ARISEN DURING READING.     * A2507040
*  THEN DEPENDING ON THE CONDITION,LOADING STOPS OR CONTROL IS GIVEN   * A2507050
*  TO ANOTHER PART OF PROGRAM.                                         * A2507060
*                                                                      * A2507070
*  NOTE- SEE WRITE MESSAGE ROUTINE,NOTE 1.                             * A2507080
*                                                                      * A2507090
*************************************************************************A2507100
         SPACE                                                        A2507110
LDREAD   TM    INPBUF,X'03'              ERROR TYPE                   A2507120
         BC    12,LDWAIT                 CATASTROPHIC ERROR           A2507130
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'EOF BEFORE    A2507140
         DC    C' '                      * END OF LOADING' AND        A2507150
         DC    AL3(MSDG1N)               * BRANCH   TO  LDT           A2507160
         BC    15,LDLDT1+4               * CARD PROCESSING ROUTINE    A2507170
         SPACE 2                                                      A2507180
*************************************************************************A2507190
*                                                                      * A2507200
*                  REWIND MAGNETIC TAPE ROUTINE                        * A2507210
*                                                                      * A2507220
*                  NAME= LDREW1                                        * A2507230
*                                                                      * A2507240
*  THIS ROUTINE IS ENTERED AT THE END OF THE LOADING PROCESS IF  THE   * A2507250
*  DEVICE USED FOR LOADER INPUT IS A TAPE UNIT.                        * A2507260
*                                                                      * A2507270
*              CALLING SEQUENCE LH  LEXIT2,DEVICE ADDRESS              * A2507280
*                               BAL LEXIT1,LDREW1                      * A2507290
*                                                                      * A2507300
*************************************************************************A2507310
         SPACE                                                        A2507320
LDREW1   STH   LEXIT2,LDREW2            SET UP NEXT CALLING SEQ.     A2507330
         SVC   9                        ENABLE I/O,EXT. INTERRUPTS   A2507340
         CNOP  4,8                      I/O REQUEST AND CONTINUE     A2507350
         SVC   13                       * (REWIND MAGNETIC TAPE)     A2507360
LDREW2   DC    X'0000'                  * TAPE UNIT DEVICE ADDRESS   A2507370
         DC    A(REWCCW)                * REWIND CCW                 A2507380
         DS    4C                       * ERROR TYPE,SENSE BYTES     A2507390
         DS    D                        * REWIND CSW                 A2507400
REWPSW   DS    D                        * PSW AT I/O INTERRUPTS      A2507410
         DC    A(LDREW3)                * NORMAL RETURN              A2507420
         DC    A(LDREW3)                * EXCEPTIONAL RETURN         A2507430
         BC    15,0(LEXIT1)             OPERATION STARTED            A2507440
         CNOP  2,4                      *                            A2507450
LDREW3   SVC   3                        RETURN TO CALLER OR TO       A2507460
         DC    A(REWPSW)                POINT OF I/O INTERRUPTION    A2507470
```

```
         EJECT                                                          A2507480
*****************************************************************     A2507490
*                                                               *     A2507500
*                    CONVERT CARD COUNT SUBROUTINE              *     A2507510
*                                                               *     A2507520
*                    NAME= LBDEC1                               *     A2507530
*                                                               *     A2507540
*                 CALLING SEQUENCE  BAL LEXIT3,LBDEC1           *     A2507550
*                                                               *     A2507560
*  THE CONTENTS OF CARD COUNTING REGISTER  CRDCNT  IS  CONVERTED  TO  * A2507570
*  DECIMAL AND RESULT IS STORED IN LOCATIONS                    *     A2507580
*                                                               *     A2507590
*      LDXBUF+72,LDXBUF+73,.....LDXBUF+79  (COL.73-80 OF CARD IMAGE)  * A2507600
*                                                               *     A2507610
*  ONE CHARACTER BLANK IS SET IN LOCATION  LDXBUF+71  (COL. 72).*     A2507620
*                                                               *     A2507630
*  THIS ROUTINE IS ENTERED AT LBDEC2 WHEN THE CARD COUNTING REGISTER  * A2507640
*  IS ALREADY UPDATED.                                          *     A2507650
*                                                               *     A2507660
*****************************************************************     A2507670
         SPACE                                                          A2507680
LBDEC1   LA    CRDCNT,1(CRDCNT)          INCREMENT CARD COUNT BY 1     A2507690
LBDEC2   CVD   CRDCNT,LDXBUF+72          CONVERT CARD COUNT TO DEC.    A2507700
         MVC   LDXBUF+72(4),LDXBUF+76    *  AND STORE IN CARD IMAGE    A2507710
         UNPK  LDXBUF+72(8),LDXBUF+72(4) *  COLUMNS 72 TO 78           A2507720
         OI    LDXBUF+79,X'F0'           SET ZERO IN COLUMN 80 ZONE    A2507730
         MVI   LDXBUF+72,X'40'           SET BLANK IN COLUMN 73        A2507740
         BCR   15,LEXIT3                 RETURN TO CALLER              A2507750
         SPACE                                                          A2507760
*****************************************************************     A2507770
*                                                               *     A2507780
*                          CONSTANTS  AREA                      *     A2507790
*                                                               *     A2507800
*****************************************************************     A2507810
         SPACE                                                          A2507820
         DS    0D                        *                             A2507830
STOPSW   DC    X'0006'                   STOP PSW (WAIT STATE BIT)     A2507840
         DC    XL6'0'                    *                             A2507850
REWCCW   CCW   X'07',*,X'00',1           REWIND CCW                    A2507860
LOMEGA   EQU   PRGPSW                    *                             A2507870
LDXBUF   EQU   LOMEGA-144                INPUT BUFFER(80)              A2507880
INPBUF   EQU   LDXBUF-1                  *                             A2507890
PRTBUF   EQU   LDXBUF+72                 INPUT BUFFER(CRD ID,SYMBOL)   A2507900
OUTBUF   EQU   LDXBUF+80                 OUTPUT BUFFER(MESS.HEADER)    A2507910
MESTXT   EQU   OUTBUF+17                 OUTPUT BUFFER(TEXT)           A2507920
WRTBUF   EQU   INPBUF                    OUTPUT BUFFER USED            A2507930
EDBUFF   EQU   INPBUF+1                  * FOR EDITING                 A2507940
LALPHA   EQU   LDXBUF-4                  BASE ADDRESS                  A2507950
ADBASE   DC    A(LALPHA)                 *                             A2507960
ADBAZE   DC    A(LALPHA-1028)            *                             A2507970
LEDDEV   DC    H'0'                      EDIT DEVICE SUPPORT ADDRESS   A2507980
LODDEV   DC    H'0'                      LOAD DEVICE SUPPORT ADDRESS   A2507990
LCONSL   DC    A(0)                      ENTRIES TO CONTROL...         A2508000
         DC    A(0)                      * ....PROGRAM.                A2508010
LDLIST   DC    X'03'                     SWITCH 'WRITE LOADING         A2508020
```

```
LDERR    EQU   LDLIST                    * CONTROL MESSAGES'          A2508030
         EJECT                                                       A2508040
*********************************************************************A2508050
*                                                                   *A2508060
*                          LOADER  ENTRIES                          *A2508070
*                                                                   *A2508080
*                 NAME= INIENT                                      *A2508090
*                                                                   *A2508100
*  THIS ROUTINE HAS TWO ENTRY POINTS, LOCATION  INIENT  AND LOCATION *A2508110
*  RESENT. LOCATION RESENT IS ENTERED FROM THE INITIALIZATION ROUTI- *A2508120
*  NE, AT THE FIRST LOADING PROCESS. LOCATION RESENT IS ENTERED EACH *A2508130
*  TIME AN END CARD HAS BEEN READ AND PROCESSED.                    *A2508140
*                                                                   *A2508150
*********************************************************************A2508160
         SPACE                                                       A2508170
INIENT   OI    LDSWS,X'50'              INITIALIZE FLAGS             A2508180
         MVC   OUTBUF+1(4),MESHDR       MESSAGE CODE (COMPONENT)     A2508190
         MVC   OUTBUF+5(59),OUTBUF+4    CLEAR OUTBUF BUFFER          A2508200
         SR    CRDCNT,CRDCNT            CLEAR CARD COUNTING REG.     A2508210
RESENT   XI    LDSWS,X'80'              SET ABS. LDR. FLAG           A2508220
         MVC   LDREFZ(4),LDREFA         CLEAR REF. TABLE             A2508230
         BAL   LEXIT1,LDTCK1            CHECK-ADJUST TABLE SPACE     A2508240
         SPACE 2                                                     A2508250
*********************************************************************A2508260
*                                                                   *A2508270
*                    GET  CARD  IMAGE  ROUTINE                      *A2508280
*                                                                   *A2508290
*                 NAME= LDCARD                                      *A2508300
*                                                                   *A2508310
*  THIS ROUTINE HAS ONE FUNCTION= OBTAINING CARDS OR CARD IMAGES. IT *A2508320
*  IS ENTERED EACH TIME  A  CARD HAS BEEN PROCESSED AND ANOTHER CARD *A2508330
*  MUST BE OBTAINED FOR PROCESSING.                                 *A2508340
*                                                                   *A2508350
*********************************************************************A2508360
         SPACE                                                       A2508370
         CNOP  0,4                      I/O REQUEST AND WAIT         A2508380
LDCARD   SVC   18                       * (GET CARD IMAGE)           A2508390
         DC    C'SIH2SYS '              * INPUT  SUPPORT FUNCTION    A2508400
         DC    FL2'80'                  * INPUT RECORD LENGTH        A2508410
         DC    A(INPBUF)                * INPUT BUFFER ADDRESS       A2508420
         TM    INPBUF,X'07'             * ERROR TYPE                 A2508430
         BC    12,LDREAD                * EXCEPTIONAL RETURN         A2508440
         LA    CRDCNT,1(CRDCNT,0)       INCREMENT CARD COUNT BY 1    A2508450
         EJECT                                                       A2508460
*********************************************************************A2508470
*                                                                   *A2508480
*                      EXAMINE CARD ROUTINE                         *A2508490
*                                                                   *A2508500
*                 NAME= LDSWT1                                      *A2508510
*                                                                   *A2508520
*  A TABLE LOOK-UP IS USED TO SELECT THE APPROPRIATE PROCESSING ROU- *A2508530
*  TINE FOR THE CARD THAT HAS JUST  BEEN  READ.  EACH ENTRY IN TABLE *A2508540
*  LTYPLD CONSISTS OF 8 BYTES.  THE FIRST CONTAINS 02 (12-2-9 PUNCH) *A2508550
*  THE NEXT THREE CONTAIN A CARD TYPE AND THE LAST FOUR  CONTAIN THE *A2508560
*  ADDRESS OF THE PROCESSING ROUTINE FOR THAT TYPE OF CARD.  TO SAVE *A2508570
```

```
*  TIME, THE CARD TYPES ARE LISTED IN ORDER OF DECREASING FREQUENCY.  * A2508580
*                                                                     * A2508590
*********************************************************************** A2508600
         SPACE                                                        A2508610
LDSWT1   LM    LDCONT,LDFINL,LDCST1          INITIALIZE REGS. FOR BXLE   A2508620
         L     LWK1RG,LTYPCD                 CARD TYPE (COLS.1-4)        A2508630
LDSWT2   CL    LWK1RG,LTYPLD(LDCONT)         SEARCH TABLE FOR THIS CARD  A2508640
         BC    8,LDSWT3                      *  TYPE AND BRANCH IF FOUND A2508650
         BXLE  LDCONT,LDSTEP,LDSWT2          *                           A2508660
         BAL   LEXIT1,LMSDG1  01            OWz TE MESSAGE 'ILLEGAL     A2508670
         DC    C' '                          * CARD TYPE IN LOADER       A2508680
         DC    AL3(MSDG11)                   * INPUT' AND GET NEXT CARD  A2508690
         BC    15,LDCARD                     *                           A2508700
LDSWT3   L     LDBRRG,LTYPLD+4(LDCONT)       FETCH ROUTINE ADDRESS       A2508710
LDSWT4   BCR   15,LDBRRG                    NO BRANCH IF ACTUAL C.SECT. A2508720
         CL    LWK1RG,LDEND0                 MUST BE SKIPPED. IS END CRD A2508730
         BC    7,LDCARD                      NO,IGNORE AND GET NEXT CARD A2508740
         OI    LDSWT4+1,X'F0'                YES,END OF SKIP TO NEXT C.S A2508750
         BC    15,LDCARD                     *                           A2508760
         EJECT                                                        A2508770
*********************************************************************** A2508780
*                                                                     * A2508790
*                      SLC CARD PROCESSING ROUTINE                    * A2508800
*                                                                     * A2508810
*                    NAME= LDSLC1                                     * A2508820
*                                                                     * A2508830
*  THIS ROUTINE SETS THE LOCATION  COUNTER TO THE ADDRESS  SPECIFIED  * A2508840
*  IN THE SLC CARD OR TO THE CURRENT  ADDRESS  OF  A VALID  SYMBOLIC  * A2508850
*  NAME, OR TO THEIR SUM IF BOTH ARE PRESENT.  CONTROL THEN RETURNS   * A2508860
*  TO THE CARD READING ROUTINE.                                       * A2508870
*                                                                     * A2508880
*  A SYMBOLIC  NAME  SPECIFIED ON THE CARD IS DETERMINED AS VALID IF  * A2508890
*  IT IS FOUND IN THE LOADER  DICTIONARY EITHER AS A CONTROL SECTION  * A2508900
*  OR AS A RELOCATED ENTRY.  IF NOT RELOCATED OR NOT IN THE DICTION-  * A2508910
*  ARY, THE SYMBOL IS INVALID AND AN APPROPRIATE MESSAGE IS WRITTEN.  * A2508920
*                                                                     * A2508930
*  IF THE CARD CONTAINS  NEITHER  ADDRESS  NOR  SYMBOL, A MESSAGE IS  * A2508940
*  PRINTED AND CONTROL RETURNS TO THE CARD READING ROUTINE.           * A2508950
*                                                                     * A2508960
*  THE  UPDATED  LOCATION  COUNTER  VALUE IS CHECKED TO SEE THAT THE  * A2508970
*  LOADER TABLE BLOCK IS NOT OVERLAID. IF IT IS, A MESSAGE IS WRIT-   * A2508980
*  TEN AND LOADING STOPS WITH A DUMP.                                 * A2508990
*                                                                     * A2509000
*********************************************************************** A2509010
         SPACE                                                        A2509020
LDSLC1   SR    LWK3RG,LWK3RG                 CLEAR WORKING REGISTER      A2509030
         OI    LDHXAD,X'20'                  SET FLAG NO HEX ADDR IN CRD A2509040
         CLI   LSLCAD,X'40'                  TEST IF HEX ADDR IN CARD    A2509050
         BC    8,LDSLC2                      NO - BRANCH                 A2509060
         LA    LEXIT1,LSLCAD                 CONVERT ADDRESS TO BINARY   A2509070
         LA    LEXIT2,6                      * AND STORE                 A2509080
         BAL   LEXIT3,LHEXB1                 *                           A2509090
         LR    LWK3RG,LEXIT2                 *                           A2509100
         NI    LDHXAD,X'DF'                  CLEAR FLG NO HEXADDR IN CRD A2509110
LDSLC2   CLI   LSLCNM,X'40'                  TEST IF SYMBOLIC ADDRESS    A2509120
```

```
         BC    6,LDSLC3                  YES - BRANCH                  A2509130
         TM    LDHXAD,X'20'              TEST IF HEX ADDR IN CARD      A2509140
         BC    12,LDSLC6                 YES - BRANCH                  A2509150
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'NEITH+R        A2509160
         DC    C' '                      * NAME NOR ADDRESS IN SLC     A2509170
         DC    AL3(MSDG1D)               * CARD' AND GET NEXT CARD     A2509180
         BC    15,LDCARD                 *                             A2509190
LDSLC3   LM    LDCONT,LDFINL,LDDICA      INITIALIZE DICT. LOOK-UP      A2509200
         ST    LWK3RG,LESADR-1           STORE CONVERTED HEXA ADDR     A2509210
LDSLC4   BXLE  LDCONT,LDSTEP,LDSLC8      LOOK UP SYMB. IN DICTIONARY   A2509220
         CLC   0(6,LDCONT),LSLCNM        * AND BRANCH IF NOT FOUND     A2509230
         BC    6,LDSLC4                  *                             A2509240
         TM    8(LDCONT),X'02'           IS SYMBOL VIRTUAL(EXTR)       A2509250
         BC    1,LDSLC8                  YES - BRANCH                  A2509260
         TM    8(LDCONT),X'01'           IS SYMBOL CONTROL SECTION     A2509270
         BC    12,LDSLC5                 YES - BRANCH                  A2509280
         TM    8(LDCONT),X'05'           IS SYMBOL RELOCATED ENTRY     A2509290
         BC    12,LDSLC8                 NO - BRANCH                   A2509300
LDSLC5   A     LWK3RG,8(LDCONT)          HEXAD ADDR + SYMBOLIC ADDR    A2509310
         SLL   LWK3RG,8                  CLEAR FIRST BYTE(DICT.TYPE)   A2509320
         SRL   LWK3RG,8                  *                             A2509330
LDSLC6   CL    LWK3RG,LOCCTR             NEW LOCCTR NOT LESS THN OLD   A2509340
         ST    LWK3RG,LOCCTR             STORE NEW LOCCTR              A2509350
         BC    10,LDSLC7                 YES - BRANCH                  A2509360
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'SLC HAS SET    A2509370
         DC    C' '                      * LOCCTR TO VALUE             A2509380
         DC    AL3(MSDG1G)               * LOADED' AND CONTINUE        A2509390
LDSLC7   CLC   LOCCTR,LDTOPP             DOES NEW LOCCTR OVERLAY       A2509400
         BC    4,LDCARD                  * LOADER TABLES-BR IF NO      A2509410
         BC    15,LDRES6                 YES,WRITE MESSAGE AND STOP    A2509420
LDSLC8   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'SYMBOL NOT     A2509430
         DC    C' '                      * DEFINED'                    A2509440
         DC    AL3(MSDG1E)               *                             A2509450
         TM    LDHXAD,X'20'              TEST IF HEX ADDR IN CARD      A2509460
         BC    1,LDCARD                  YES - BRANCH                  A2509470
         L     LWK3RG,LESADR-1           RELOAD CONVERTED HEXA ADDR    A2509480
         BC    15,LDSLC6                 *                             A2509490
         SPACE 2                                                       A2509500
*****************************************************************     A2509510
*                                                               *     A2509520
*                    ICS CARD PROCESSING ROUTINE                *     A2509530
*                                                               *     A2509540
*                  NAME= LDICS1                                 *     A2509550
*                                                               *     A2509560
*  THIS ROUTINE  ESTABLISHES A DICTIONARY ENTRY FOR THE CONTROL SEC-    A2509570
*  TION NAME ON THE ICS CARD IF NO ENTRY FOR THAT  CARD  EXISTS AND,    A2509580
*  IF NECESSARY, ADDS THE CARD-SPECIFIED CONTROL  SECTION  LENGTH TO    A2509590
*  THE LOCATION COUNTER.                                        *     A2509600
*                                                               *     A2509610
*****************************************************************     A2509620
         SPACE                                                         A2509630
LDICS1   LA    LEXIT1,LICSLG             LOAD C.S. LENGTH FROM CARD    A2509640
         LA    LEXIT2,4                  CONVERT LENGTH OF             A2509650
         BAL   LEXIT3,LHEXB1             * CONTROL SECTION AND         A2509660
         STH   LEXIT2,LSECLG             * STORE                       A2509670
```

```
         XC    LICSLG(4),LICSLG          CLEAR C.S. LENGTH IN CARD   A2509680
         LA    LEXIT2,LDICS2             *                           A2509690
         BAL   LEXIT1,LDMOR1             IS NAME IN DICTIONARY       A2509700
         TM    8(LDCONT),X'02'           YES-IS SYMBOL VIRTUAL(EXTR) A2509710
         BC    12,LDCARD                 NO - BRANCH TO LDCARD       A2509720
LDICS2   BAL   LEXIT3,LDRES1             YES- RESERVE STORAGE        A2509730
         BAL   LEXIT1,LDTCK1             CHECK ROOM IN TABLES        A2509740
         BC    15,LDCARD                 *                           A2509750
         EJECT                                                       A2509760
*************************************************************************A2509770
*                                                                     * A2509780
*                      ESD CARD PROCESSING ROUTINE                    * A2509790
*                                                                     * A2509800
*                    NAME= LDESD1                                     * A2509810
*                                                                     * A2509820
*  THIS  ROUTINE  DETERMINES  THE  TYPE OF SYMBOL (0, 1 OR 2) ON THE  * A2509830
*  CARD AND PROCEEDS AS FOLLOWS.                                      * A2509840
*                                                                     * A2509850
*  ESD0. THE ADDRESS IS CALCULATED AT WHICH THE ENTRY MAY BE  STORED  * A2509860
*        IN THE REFERENCE TABLE REFTAB.                               * A2509870
*                                                                     * A2509880
*        THE SYMBOL IS LOOKED FOR IN THE DICTIONARY AND IF IT IS NOT  * A2509890
*        FOUND,  A  'CONTROL  SECTION'  ENTRY IS CREATED IN BOTH THE  * A2509900
*        DICTIONARY AND REFTAB, AND THE LOCATION COUNTER IS UPDATED.  * A2509910
*                                                                     * A2509920
*        IF THE SYMBOL IS FOUND IN THE DICTIONARY LABELED AS 'EXTER-  * A2509930
*        NAL',  THE ENTRY IS CHANGED TO A 'CONTROL SECTION' ENTRY, A  * A2509940
*        'CONTROL SECTION' ENTRY IS CREATED IN REFTAB  AND THE LOCA-  * A2509950
*        TION COUNTER IS UPDATED.                                     * A2509960
*                                                                     * A2509970
*        IF THE SYMBOL IS FOUND IN THE DICTIONARY LABELED AS 'ENTRY'  * A2509980
*        A MESSAGE IS WRITTEN TO THAT EFFECT AND  PROCESSING  CONTI-  * A2509990
*        NUES AS FOR AN 'EXTERNAL' LABEL.                             * A2510000
*                                                                     * A2510010
*        IF THE SYMBOL IS FOUND IN THE DICTIONARY  LABELED  AS 'CON-  * A2510020
*        TROL SECTION', THE STORED C.S. LENGTH IS COMPARED WITH THAT  * A2510030
*        STATED IN THE CARD AND IF THEY DIFFER, A MESSAGE IS WRITTEN  * A2510040
*        TO THAT EFFECT.                                              * A2510050
*                                                                     * A2510060
*  ESD1. THE SYMBOL IS LOOKED FOR IN THE DICTIONARY AND IF IT IS NOT  * A2510070
*        FOUND, AN 'ENTRY' ENTRY IS CREATED IN THE DICTIONARY  ONLY.  * A2510080
*                                                                     * A2510090
*        IF THE SYMBOL IS FOUND IN THE DICTIONARY LABELED AS 'EXTER-  * A2510100
*        NAL', THE ENTRY IS CHANGED TO AN 'ENTRY' ENTRY.              * A2510110
*                                                                     * A2510120
*        IF THE SYMBOL IS FOUND IN THE DICTIONARY LABELED AS 'ENTRY'  * A2510130
*        OR 'CONTROL SECTION', A MESSAGE IS WRITTEN TO THAT  EFFECT.  * A2510140
*                                                                     * A2510150
*  ESD2. THE ADDRESS IS CALCULATED  AT  WHICH  THE  ELEMENT  MAY  BE  * A2510160
*        STORED IN REFTAB.                                            * A2510170
*                                                                     * A2510180
*        THE SYMBOL IS LOOKED FOR IN THE DICTIONARY AND IF IT IS NOT  * A2510190
*        FOUND,  AN 'EXTERNAL' ENTRY IS CREATED IN BOTH THE DICTION-  * A2510200
*        ARY AND REFTAB.                                              * A2510210
*                                                                     * A2510220
```

```
*        IF  FOUND,  THE  ENTRY  IN  THE DICTIONARY IS LEFT AS IT IS  * A2510230
*        (CONTROL SECTION, ENTRY  OR  EXTERNAL)  AND  AN  'EXTERNAL'  * A2510240
*        ENTRY IS CREATED IN REFTAB.                                   * A2510250
*                                                                      * A2510260
*  ANY TIME AN ENTRY IS CREATED IN EITHER TABLE,  THE TABLE SPACE IS   * A2510270
*  CHECKED AND ADJUSTED IF NECESSARY.                                  * A2510280
*                                                                      * A2510290
************************************************************************ A2510300
         SPACE 2                                                       A2510310
LDESD1   TM    LESTYP,X'03'            IS CARD TYPE ESD-0             A2510320
         BC    5,LDESDG                NO,BRANCH - YES,INITIALIZE     A2510330
         LM    LDCONT,LDFINL,LDOPTA    * OPTION  TABLE LOOK-UP        A2510340
LDESDF   BXLE  LDCONT,LDSTEP,LDESDG    BR. IF TABLE EXHAUSTED         A2510350
         CLC   0(6,LDCONT),LESNME      IS C.S. NAME IN TABLE          A2510360
         BC    7,LDESDF                NO,GO TO NEXT ELEMENT          A2510370
         NI    LDSWT4+1,X'0F'          YES,SET LDSWT ROUTINE TO       A2510380
         BC    15,LDCARD               * SKIP ONE CONTROL SECTION     A2510390
LDESDG   TM    LDSTAT,X'03'            HAS TXT CARD BEEN LOADED       A2510400
         BC    5,LDESD6                YES - BRANCH                   A2510410
LDESD2   NI    LDABSL,X'7F'            CLEAR ABS. LOADER FLAG         A2510420
         OI    LDINIT,X'04'            SET END CARD NOT FOUND FLAG    A2510430
         NI    LESTYP,X'03'            CLEAR NON SIGNIFICANT BITS     A2510440
         TM    LESTYP,X'01'            IS CARD TYPE ESD 1             A2510450
         BC    1,LDESD3                YES - BRANCH                   A2510460
LDESDA   LH    LWK1RG,LESDID           LOAD ESD ID OF ELEMENT         A2510470
         LA    LWK2RG,1(LWK1RG,0)      ADD 1 TO ID FOR NEXT ESD       A2510480
         STH   LWK2RG,LESDID           STORE NEXT ESD ID              A2510490
         LA    LEXIT2,LDESDC           COMPUTE ADDRESS OF ELEMENT     A2510500
         SLL   LWK1RG,3                * IN REFERENCE TABLE           A2510510
         A     LWK1RG,LDREFA           *                              A2510520
         C     LWK1RG,LDREFZ           IS ELEM.ADDR.PAST END REFTB    A2510530
         BC    12,LDESDB               YES - BRANCH                   A2510540
         ST    LWK1RG,LDREFZ           ELEM.ADDR.IS NEW END REFTB     A2510550
         BAL   LEXIT1,LDTCK1           CHECK TABLE SPACE              A2510560
         L     LWK1RG,LDREFZ           ADJUST ADDRESS OF ELEMENT      A2510570
LDESDB   MVC   4(4,LWK1RG),LESTYP      TYPE AND ASS.ADDR.TO REFTB     A2510580
         CLI   LESNME,X'40'            IS NAME BLANK(PRIVATE CODE)    A2510590
         BC    8,LDMOR2                YES - BRANCH                   A2510600
         BAL   LEXIT1,LDMOR1           IS NAME IN DICTIONARY          A2510610
         ST    LDCONT,0(LWK1RG,0)      YES-DICT ENTRY TO REFTAB       A2510620
         TM    LESTYP,X'02'            TYPE IN REFTAB IS EXTERNAL     A2510630
         BC    1,LDESDZ                YES - BRANCH                   A2510640
         TM    8(LDCONT),X'02'         TYPE.IN DICT. IS EXTERNAL      A2510650
         BC    1,LDESDE                YES - BRANCH                   A2510660
         TM    8(LDCONT),X'01'         TYPE.IN DICT. IS ENTRY         A2510670
         BC    1,LDESDD                * OR C.S.   YES - BRANCH       A2510680
         CLC   6(2,LDCONT),LSECLG      COMPARE LENGTHS                A2510690
         BC    8,LDESDZ                EQUAL - BRANCH                 A2510700
         BAL   LEXIT1,LMSDG1           WRITE MESSAGE 'CONTROL         A2510710
         DC    C' '                    * SECTION DEFINED WITH TWO     A2510720
         DC    AL3(MSDG18)             * LENGTHS' AND CONTINUE        A2510730
         BC    15,LDESDZ               *                              A2510740
LDESDC   ST    LDCONT,0(LWK1RG,0)      NEW DICT ENTRY TO REFTAB       A2510750
         MVC   8(4,LDCONT),LESTYP      TYP AND ORG FRM REF TO DICT    A2510760
         XC    6(2,LDCONT),6(LDCONT)   CLEAR LENGTH IN DICT.          A2510770
```

```
         TM    LESTYP,X'02'              TYPE IN REFTAB IS EXTERNAL  A2510780
         BC    1,LDESDZ                  YES - BRANCH                A2510790
LDESDE   BAL   LEXIT3,LDRES1             RESERV STORAGE FOR CSECTION A2510800
         BC    15,LDESDZ                 * AND CONTINUE              A2510810
LDESDD   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'USED AS      A2510820
         DC    C' '                      * ENTRY AND CONTROL SECTION A2510830
         DC    AL3(MSDG17)               * NAME' AND CONTINUE        A2510840
         BC    15,LDESDE                 *                           A2510850
LDESD3   LA    LEXIT2,LDESD4             PREPARE RETURN ADDRESS      A2510860
         BAL   LEXIT1,LDMOR1             IS NAME IN DICTIONARY       A2510870
         TM    8(LDCONT),X'02'           TYPE.IN DICT. IS EXTERNAL   A2510880
         BC    1,LDESD4                  YES - BRANCH                A2510890
         TM    8(LDCONT),X'01'           TYPE IN DICT. IS ENTRY      A2510900
         BC    1,LDESD5                  YES - BRANCH                A2510910
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'USED AS      A2510920
         DC    C' '                      * ENTRY AND CONTROL SECTION A2510930
         DC    AL3(MSDG17)               * NAME' AND CONTINUE        A2510940
         BC    15,LDESDX                 *                           A2510950
LDESD5   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'ENTRY POINT  A2510960
         DC    C' '                      * IS REPEATED'              A2510970
         DC    AL3(MSDG1J)               * AND CONTINUE              A2510980
         BC    15,LDESDX                 *                           A2510990
LDESD6   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'ESD CARD     A2511000
         DC    C' '                      * FOLLOWS TXT CARD'         A2511010
         DC    AL3(MSDG16)               * AND GET NEXT CARD         A2511020
         BC    15,LDCARD                 *                           A2511030
LDESD4   MVC   6(2,LDCONT),LSECID        STORE C.S. IDENT, TYPE AND  A2511040
         MVC   8(4,LDCONT),LESTYP         * ORIGIN IN DICTIONARY      A2511050
LDESDZ   BAL   LEXIT1,LDTCK1             CHECK ROOM IN TABLES        A2511060
LDESDX   LA    LWK2RG,16                 COMPUTE NUMBER OF           A2511070
         LH    LWK1RG,LESDCT             * BYTES ON CARD             A2511080
         SR    LWK1RG,LWK2RG             * MINUS 16                  A2511090
         BC    8,LDCARD                  IF ZERO GET NEXT CARD       A2511100
         MVC   LDXBUF+16(32),LDXBUF+32   SHIFT CARD IMAGE COLUMNS    A2511110
         STH   LWK1RG,LESDCT             * 33-64 TO COLS. 17-48      A2511120
         BC    15,LDESD2                 * AND GO TO PROCESS THE     A2511130
*                                        * NEXT ESD TERM ON CARD.    A2511140
         EJECT                                                       A2511150
*************************************************************************A2511160
*                                                                       * A2511170
*                     TXT CARD PROCESSING ROUTINE                       * A2511180
*                                                                       * A2511190
*                   NAME= LDTXT1                                        * A2511200
*                                                                       * A2511210
*  THE  FIRST  TXT  CARD  CALLS  SUBROUTINE 'LDSTAB' TO RELOCATE ALL    * A2511220
*  'ENTRY' ENTRIES IN THE DICTIONARY.                                   * A2511230
*                                                                       * A2511240
*  THE RELOCATED ADDRESS OF THE FIRST BYTE OF TEXT IS THEN  OBTAINED    * A2511250
*  AND THE TEXT IN THE CARD IS  STORED  STARTING  AT  THIS  ADDRESS.    * A2511260
*  THE ADDRESS OF  THE  FIRST  BYTE OF TEXT IN THE FIRST TXT CARD IS    * A2511270
*  ALSO STORED. WHEN THE PROGRAM IS EXECUTED, CONTROL IS TRANSFERRED    * A2511280
*  TO THIS ADDRESS IF NO OTHER TRANSFER ADDRESS IS GIVEN LATER IN AN    * A2511290
*  END OR LDT CARD.                                                     * A2511300
*                                                                       * A2511310
*************************************************************************A2511320
```

```
         SPACE                                                          A2511330
LDTXT1   TM    LDSTAT,X'03'          TXT,REP,RLD CARDS READ            A2511340
         BC    5,LDTXT2              SOME - BRANCH                     A2511350
         BAL   LEXIT3,LDSTAB         SET TXT READ FLAG AND RELO-       A2511360
         BC    15,LDTXT3             * CATE ENTRIES IN DICT.           A2511370
LDTXT2   BC    4,LDTXT3              TXT READ ONLY - BRANCH            A2511380
         BAL   LEXIT1,LMSDG1         WRITE MESSAGE 'TXT CARD           A2511390
         DC    C' '                  * FOLLOWS REP OR RLD CARD'        A2511400
         DC    AL3(MSDG12)           * AND CONTINUE                    A2511410
LDTXT3   LH    LWK2RG,LTXTCT         LOAD TEXT LENGTH                  A2511420
         LTR   LWK2RG,LWK2RG         * IS LENGTH=0                     A2511430
         BC    8,LDCARD              * YES,BRANCH-NO,                  A2511440
         MVC   LWZCSI(2),LESDID      STORE ESD ID AND ADDRESS          A2511450
         MVC   LWZORG(3),LTXTAD      * IN WORKING AREA FOR             A2511460
         LA    LEXIT2,LDTXT6         * RELOCATION ROUTINE              A2511470
         BAL   LEXIT1,LDRLC1         RELOCATE TEXT ADDRESS             A2511480
         TM    LDABSL,X'80'          ABSOLUTE LOADER FLAG ON           A2511490
         BC    1,LDTXT4              YES - BRANCH                      A2511500
         OI    8(LWK4RG),X'08'       SET FLG IN DICT'C.S.SYMB.'        A2511510
LDTXT4   CLI   LTXTCT+1,X'38'        BYTE COUNT EXCEEDS 56             A2511520
         BC    12,LDTXT5             NO - BRANCH                       A2511530
         BAL   LEXIT1,LMSDG1         WRITE MESSAGE 'BYTE COUNT         A2511540
         DC    C' '                  * ON TXT CARD EXCEEDS 56'         A2511550
         DC    AL3(MSDG14)           * AND CONTINUE                    A2511560
LDTXT5   LH    LWK2RG,LTXTCT         COMPUTE TEXT LENGTH               A2511570
         LR    LWK1RG,LWK2RG         * PLUS                            A2511580
         A     LWK2RG,LWZORG-1       * RELOCATED ADDR OF TEXT          A2511590
         S     LWK2RG,LDTOPP         START OF TABLES OVERLAID          A2511600
         BC    2,LDTXT7              YES - BRANCH                      A2511610
         BCTR  LWK1RG,0              PREPARE DATA (TEXT BYTE           A2511620
         STC   LWK1RG,LDMOV1+1       * COUNT AND TEXT LOADING          A2511630
         L     LWK1RG,LWZORG-1       * ADDRESS) AND STORE              A2511640
LDMOV1   MVC   0(1,LWK1RG),LBYTXT    * TEXT IN CORE STORAGE            A2511650
         TM    LDFTTR,X'40'          1ST TIME TRANSFER FLAG            A2511660
         BC    12,LDCARD             NO - BRANCH                       A2511670
         NI    LDFTTR,X'BF'          CLEAR FLAG                        A2511680
         MVC   LDENTA(3),LWZORG      SAVE FIRST ADDRESS LOADED         A2511690
         BC    15,LDCARD             GET NEXT CARD                     A2511700
LDTXT6   BAL   LEXIT1,LMSDG1         WRITE MESSAGE 'ADDRESS OUT-       A2511710
         DC    C' '                  * SIDE RANGE OR C.S               A2511720
         DC    AL3(MSDG13)           * ALREADY LOADED' AND GET         A2511730
         BC    15,LDCARD             * NEXT CARD                       A2511740
LDTXT7   BAL   LEXIT1,LMSDG1         WRITEMESSAGE 'TEXT OVERLAYS       A2511750
         DC    C' '                  * LOADER TABLES' AND GET          A2511760
         DC    AL3(MSDG15)           * NEXT CARD                       A2511770
         BC    15,LDCARD             *                                 A2511780
         SPACE 2                                                        A2511790
*****************************************************************     A2511800
*                                                               *     A2511810
*                    REP CARD PROCESSING ROUTINE                *     A2511820
*                                                               *     A2511830
*                   NAME= LDREP1                                *     A2511840
*                                                               *     A2511850
*  THIS ROUTINE CONVERTS THE REP CARD INTO THE FORMAT OF A TXT  CARD  * A2511860
*  AND GIVES CONTROL TO THE TXT CARD PROCESSING ROUTINE.        *     A2511870
```

```
*                                                                     * A2511880
*********************************************************************** A2511890
         SPACE                                                        A2511900
LDREP1   TM    LDSTAT,X'03'              TXT,REP,RLD CARDS READ       A2511910
         BC    5,LDREP2                  SOME-BRANCH, NONE-RELOCATE   A2511920
         BAL   LEXIT3,LDSTAB             * ENTRIES IN DICTIONARY      A2511930
LDREP2   OI    LDSTAT,X'02'              SET FLAG REP/RLD CARD READ   A2511940
         LA    LEXIT2,6(0,0)             CONVERT ADDRESS TO BINARY    A2511950
         LA    LEXIT1,LREPAD             * AND STORE                  A2511960
         BAL   LEXIT3,LHEXB1             *                            A2511970
         ST    LEXIT2,LTXTAD-1           *                            A2511980
         LA    LEXIT1,LREPID             CONVERT ESD ID TO BINARY     A2511990
         LA    LEXIT2,3(0,0)             * AND STORE                  A2512000
         BAL   LEXIT3,LHEXB1             *                            A2512010
         STH   LEXIT2,LESDID             *                            A2512020
         LA    LEXIT1,LRPBYT             FIRST BYTE OF TEXT           A2512030
         SR    LDCONT,LDCONT             CLEAR AND SET BXH COUNTER    A2512040
         LA    LDSTEP,2                  *                            A2512050
         LA    LDFINL,21                 *                            A2512060
         CLI   0(LEXIT1),X'40'           IS FIRST BYTE OF TEXT BLANK  A2512070
         BC    8,LDCARD                  YES-BRANCH                   A2512080
LDREP3   LA    LEXIT2,4(0,0)             CONVERT 4 BYTES OF TEXT      A2512090
         BAL   LEXIT3,LHEXB1             * TO HEXADECIMAL             A2512100
         STH   LEXIT2,LRPBYT(LDCONT)     * AND STORE                  A2512110
         BXH   LDCONT,LDSTEP,LDREP4      BRANCH IF 22BYTES CONVERTED  A2512120
         CLI   0(LEXIT1),X'6B'           IS FIRST BYTE A COMMA        A2512130
         BC    6,LDREP4                  NO - BRANCH                  A2512140
         LA    LEXIT1,1(LEXIT1,0)        GET NEXT FOUR BYTES          A2512150
         BC    15,LDREP3                 * AND RETURN                 A2512160
LDREP4   CLI   0(LEXIT1),X'40'           IS FIRST BYTE BLANK          A2512170
         BC    8,LDREPZ                  YES - BRANCH                 A2512180
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'BLANK OR      A2512190
         DC    C' '                      * COMMA MISSING IN REP CARD  A2512200
         DC    AL3(MSDG1B)               * AND GET NEXT CARD          A2512210
         BC    15,LDCARD                 *                            A2512220
LDREPZ   STH   LDCONT,LTXTCT             STORE BYTE COUNT             A2512230
         BC    15,LDTXT3                 GO TO TXT CARD ROUTINE       A2512240
         EJECT                                                        A2512250
*********************************************************************** A2512260
*                                                                     * A2512270
*                     RLD CARD PROCESSING ROUTINE                     * A2512280
*                                                                     * A2512290
*                    NAME= LDRLD1                                     * A2512300
*                                                                     * A2512310
*  THIS ROUTINE ESTABLISHES THE REAL VALUE OF EXPRESSIONS DEFINED BY  * A2512320
*                                                                     * A2512330
*                        ADSYMB   DC   A(SYMBOL+N)                    * A2512340
*                                                                     * A2512350
*  WHERE ALL THE REQUIRED ELEMENTS ARE PRESENT.   THE ADDRESS OF THE  * A2512360
*  EXPRESSION IS RELOCATED FIRST AND THEN THE VALUE OF THE  CONSTANT  * A2512370
*  IS CALCULATED.                                                     * A2512380
*                                                                     * A2512390
*  IF THE ELEMENT IN THE DICTIONARY IS STILL AN EXTERNAL, THE AVAIL-  * A2512400
*  ABLE DATA ARE STORED IN TABLE 'LIST' PENDING  FURTHER  TREATMENT.  * A2512410
*                                                                     * A2512420
```

```
*  NOTE-  TO AVOID  CONFUSION, THE TERM 'POSITION HEADER' IS USED TO  * A2512430
*  INDICATE THE ESID NUMBER OF THE EXPRESSION  AND THE TERM 'RELOCA-  * A2512440
*  TION HEADER' REFERS TO THE ESID NUMBER OF THE SYMBOL CONTAINED IN  * A2512450
*  THE EXPRESSION.                                                    * A2512460
*                                                                     * A2512470
********************************************************************** A2512480
LDRLD1   TM    LDSTAT,X'03'              TXT,REP-RLD CARDS READ      A2512490
         BC    5,LDRLD2                  SOME-BRANCH,NONE-RELOCATE   A2512500
         BAL   LEXIT3,LDSTAB             * ENTRIES IN DICTIONARY     A2512510
LDRLD2   OI    LDSTAT,X'03'              SET FLAG REP-RLD CARD READ  A2512520
         TM    LDABSL,X'80'              ABSOLUTE LOADER FLAG        A2512530
         BC    1,LDCARD                  YES-BRANCH                  A2512540
         SR    LDCONT,LDCONT             CLEAR AND INITIALIZE CNTR   A2512550
         LA    LDSTEP,4(0,0)             * (STEP=4)                  A2512560
         LH    LDFINL,LRLDCT             * (NBER OF BYTES IN CARD)   A2512570
         CLR   LDFINL,LDSTEP             BYTE COUNT IN CARD=0        A2512580
         BC    12,LDCARD                 YES-BRANCH                  A2512590
         BCTR  LDFINL,0                  DECREMENT LENGTH BY 1 AND   A2512600
         STC   LDFINL,LDRMV1+1           * STORE FOR LATER MOVE      A2512610
         STC   LDFINL,LDRMV2+1           * CHARACTER INSTRUCTIONS    A2512620
LDRLD3   AR    LDCONT,LDSTEP             *                           A2512630
         LH    LWK1RG,LRELHD             COMPUTE ADDRESS OF ENTRY IN A2512640
         SLL   LWK1RG,3                  * REFTAB (=START OF TABLE + A2512650
         A     LWK1RG,LDREFA             * RELOC HEADER X 8)TO GET   A2512660
         ST    LWK1RG,LWZLST             * ADDR.OF SYMB.IN DICT.AND  A2512670
         MVC   LWZLDC(4),0(LWK1RG)       * STORE IN DICT.WORK AREA   A2512680
LDRLD4   MVC   LWZFLG(1),LRLFLG          STORE CONTINUATION FLAG     A2512690
         MVC   LWZCSI(2),LPOSHD          STORE POS-HEADER AND SYMBOL A2512700
         MVC   LWZORG(3),LRLADR          ASSBLY ADDR.IN DIC.WRK.AREA A2512710
         LA    LEXIT2,LDRLD6             RELOCATE ADDRESS IN DICT.   A2512720
         BAL   LEXIT1,LDRLC1             * WORKING AREA              A2512730
         OI    8(LWK4RG),X'08'           SET'C.S.BEING LOADED'       A2512740
         MVC   LWZLAD(3),LWZORG          STORE RELOCATED ADDRESS     A2512750
         LA    LEXIT1,LDRLD5             CALC.REAL VALUE OF EXPRESS. A2512760
         BAL   LEXIT2,LDLST1             * AND STORE AT RELOC.ADDR.  A2512770
         L     LWK2RG,LDLSTZ             NOT PROCESSED-STORE ADDR.OF A2512780
         MVC   0(8,LWK2RG),LWZLDC        * DIC.ENTRY,FLAG AND RELOC. A2512790
         LA    LWK2RG,8(LWK2RG,0)        * ADDRESS IN TABLE LIST     A2512800
         ST    LWK2RG,LDLSTZ             UPDATE END OF LIST TABLE    A2512810
         BAL   LEXIT1,LDTCK1             CHECK TABLE SPACE           A2512820
LDRLD5   BXH   LDCONT,LDSTEP,LDCARD      BR IF END OF RLD CARD       A2512830
         TM    LWZFLG,X'01'              TEST CONTINUATION FLAG      A2512840
         BC    8,LDRMV2                  OFF-BRANCH                  A2512850
LDRMV1   MVC   LRLFLG,LRLFLG+4           ON-MOVE UP NEXT X BYTES BY  A2512860
         BC    15,LDRLD4                 * FOUR POSITIONS            A2512870
LDRMV2   MVC   LRELHD,LRELHD+8           MOVE UP NEXT X BYTES BY     A2512880
         BC    15,LDRLD3                 * EIGHT POSITIONS           A2512890
LDRLD6   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'ADDRESS NOT  A2512900
         DC    C' '                      * RELOCATABLE' AND SKIP TO  A2512910
         DC    AL3(MSDG1M)               * NEXT ENTRY                A2512920
         BC    15,LDRLD5                 *                           A2512930
         SPACE 2                                                     A2512940
********************************************************************** A2512950
*                                                                     * A2512960
*                    END CARD PROCESSING ROUTINE                      * A2512970
```

```
*                                                                     * A2512980
*                    NAME= LDEND1                                     * A2512990
*                                                                     * A2513000
*  THIS ROUTINE LOOKS FOR A PROGRAM EXECUTION ENTRY  POINT, AND GOES  * A2513010
*  THROUGH  TABLE LIST TO CALCULATE, WHERE POSSIBLE, THE EXPRESSIONS  * A2513020
*  IT CONTAINS.   PROCESSED  EXPRESSIONS  ARE REMOVED FROM THE TABLE  * A2513030
*  AND THE GAPS CLOSED UP.                                            * A2513040
*                                                                     * A2513050
*********************************************************************** A2513060
         SPACE                                                        A2513070
LDEND1   BAL   LEXIT3,LDSTAC              RELOCATE ENTRIES IN DICT.  A2513080
         TM    LDENTR,X'10'               TEST IF ENTRY POINT FOUND  A2513090
         BC    8,*+8                      *YES-BRANCH                A2513100
         BAL   LEXIT3,LDENT1              NO,THEN LOOK FOR ENTRY     A2513110
         LA    LEXIT4,RESENT              SAVE RETURN ADDRESS        A2513120
LDEND2   XC    LWZLST(4),LWZLST           CLEAR LIST WORKING AREA    A2513130
         LM    LDCONT,LDFINL,LDLSTA       INITIALIZE COUNTER         A2513140
         LA    LDFINL,1(LDFINL)           *                          A2513150
         LR    LMIDRG,LDCONT              TRANSFER LIST TABLE ELEMENT A2513160
LDEND3   MVC   LWZLDC(8),0(LDCONT)        * TO DICT. WORKING AREA    A2513170
         BXH   LDCONT,LDSTEP,LDEND4       BRANCH IF END OF LIST TABLE A2513180
         LA    LEXIT1,LDEND3              CALC.REAL VALUE OF EXPRESS A2513190
         BAL   LEXIT2,LDLST1              * AND STORE AT RELOC.ADDR. A2513200
         MVC   0(8,LMIDRG),LWZLDC         NOT PROCESSED-CLOSE GAP IN A2513210
         AR    LMIDRG,LDSTEP              * LIST TABLE AND PROCESS   A2513220
         BC    15,LDEND3                  * NEXT ELEMENT             A2513230
LDEND4   ST    LMIDRG,LDLSTZ              UP DATE END OF LIST TABLE  A2513240
         NI    LDINIT,X'78'               CLEAR SOME FLAGS           A2513250
         BCR   15,LEXIT4                  LOAD NEXT SECT.OR LDT   PSW A2513260
         EJECT                                                        A2513270
*********************************************************************** A2513280
*                                                                     * A2513290
*                    LDT CARD PROCESSING ROUTINE                      * A2513300
*                                                                     * A2513310
*                    NAME= LDLDT1                                     * A2513320
*                                                                     * A2513330
*  THIS ROUTINE TERMINATES THE LOADING PROCESS BY-                    * A2513340
*                                                                     * A2513350
*  1. LOOKING FOR AN ENTRY POINT INTO THE PROGRAM.                    * A2513360
*                                                                     * A2513370
*  2. LOOKING FOR ANY  REMAINING  EXTERNAL SYMBOLS IN THE DICTIONARY  * A2513380
*     AND WRITING  A MESSAGE FOR ANY IT FINDS.                        * A2513390
*                                                                     * A2513400
*  3. WRITING THE PROGRAM ENTRY POINT DETERMINED ABOVE IF THE LOADER  * A2513410
*     PRINTS THE LOADING CONTROL MESSAGES. OTHERWISE, IF ANY WARNING  * A2513420
*     OR ERROR CONDITION HAS BEEN DETECTED, THE LOADER ISSUES A MES-  * A2513430
*     SAGE ON THE 1052 PRINTER-KEYBOARD AND ENTERS THE WAIT STATE.    * A2513440
*                                                                     * A2513450
*  4. TRANSFERRING CONTROL TO THE SELF-LOADING PROGRAM GENERATOR  OR  * A2513460
*     TO THE PROGRAM LOADED.                                          * A2513470
*                                                                     * A2513480
*********************************************************************** A2513490
         SPACE                                                        A2513500
LDLDT1   BAL   LEXIT3,LDENT1              LOOK FOR ENTRY POINT       A2513510
         TM    LDINIT,X'04'               HAS END CARD BEEN READ     A2513520
```

```
         BC    12,LDLDT2                 YES-BRANCH                      A2513530
         BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'LDT CARD NOT     A2513540
         DC    C' '                      * PRECEDED BY END CARD          A2513550
         DC    AL3(MSDG19)               * AND CONTINUE                  A2513560
         BAL   LEXIT3,LDSTAC             RELOCATE ENTRIES IN DICT.       A2513570
         BAL   LEXIT4,LDEND2                                             A2513580
LDLDT2   LM    LDCONT,LDFINL,LDDICA      INITIALIZE COUNTER              A2513590
LDLDT3   BXLE  LDCONT,LDSTEP,LDLDT4      BRANCH IF END OF DICT.          A2513600
         TM    8(LDCONT),X'02'           IS ENTRY 'EXTERNAL'             A2513610
         BC    8,LDLDT3                  NO - GO TO NEXT ENTRY           A2513620
         MVC   PRTBUF(6),0(LDCONT)       PRINT MESSAGE 'EXTERNAL         A2513630
         XC    PRTBUF+6(2),PRTBUF+6      * SYMBOL HAS NO REAL            A2513640
         BAL   LEXIT1,LMSDG1             * DEFINITION'                   A2513650
         DC    C' '                      *                               A2513660
         DC    AL3(MSDG1A)               *                               A2513670
         BC    15,LDLDT3                 GO TO NEXT ENTRY                A2513680
LDLDT4   LA    LEXIT1,LDTPSW             CONVERT TRANSFER PSW            A2513690
         L     LEXIT2,AMSDGZ             HEXADECIMAL AND WRITE IT        A2513700
         BAL   LEXIT3,LBHEX1             *                               A2513710
         MVC   PRTBUF(8),MESTXT          CLEAR CARD IDENTIFICATION       A2513720
         BAL   LEXIT1,LPRINT             *                               A2513730
         DC    C'1'                      *                               A2513740
         DC    AL3(MSDG1Z)               *                               A2513750
         LH    LEXIT2,LODDEV             GET LOAD DEVICE ADDRESS         A2513760
         LTR   LEXIT2,LEXIT2             IS IT ZERO - YES,DEVICE IS      A2513770
         BC    8,LDLDT5                  * NOT A 2400 TU,BRANCH -NO,     A2513780
         BAL   LEXIT1,LDREW1             * REWIND MAGNETIC TAPE          A2513790
LDLDT5   TM    LDLIST,X'03'              HAS ANY WARNING,ERROR CON-      A2513800
         BC    4,LDSTOP                  * -DITION BEEN FOUND.YES,BR     A2513810
LDLDT6   BC    0,LDEDIT                  * NO,GO TO EDIT PRG. LOADED     A2513820
         CNOP  2,4                       * OR,                           A2513830
         SVC   3                         * LOAD TRANSFER PSW             A2513840
         DC    A(LDTPSW)                 *                               A2513850
         SPACE 5                                                         A2513860
*****************************************************************************A2513870
*                                                                           * A2513880
*                    ENTRY POINT RESEARCH SUBROUTINE                        * A2513890
*                                                                           * A2513900
*                    NAME= LDENT1                                           * A2513910
*                                                                           * A2513920
*                 CALLING SEQUENCE - BAL LEXIT3,LDENT1                      * A2513930
*                                                                           * A2513940
*  THIS SUBROUTINE IS ENTERED FROM THE LDT  AND  END CARD PROCESSING        * A2513950
*  ROUTINES.                                                                * A2513960
*                                                                           * A2513970
*****************************************************************************A2513980
         SPACE                                                           A2513990
LDENT1   MVC   LWZORG(3),LENDAD          FETCH ABSOLUTE ENTRY FROM       A2514000
         CLI   LENDAD,X'40'              END CARD AND TEST IF BLANK      A2514010
         BC    8,LDENT3                  YES - BRANCH                    A2514020
         MVC   LWZCSI(2),LESDID          FETCH ESD IDENT.                A2514030
         LR    LEXIT2,LEXIT3             RELOCATE ENTRY POINT ADDR.      A2514040
         BAL   LEXIT1,LDRLC1             EXIT TO CALLER IF INVALID       A2514050
         MVC   LDENTA(3),LWZORG          STORE RELOC. TRANSFER ADDR.     A2514060
LDENT2   NI    LDENTR,X'EF'              CLEAR FLAG 'END CARD ADDR.      A2514070
```

```
         BCR   15,LEXIT3                 * NOT SAVED' AND EXIT       A2514080
LDENT3   CLI   LESNME,X'40'              ENTRY POINT SYMBOL IS BLANK A2514090
         BCR   8,LEXIT3                  YES - EXIT BACK TO CALLER   A2514100
         LR    LEXIT2,LEXIT3             LOOK UP SYMBOL IN DICT. AND A2514110
         BAL   LEXIT1,LDMOR1             * EXIT IF NOT FOUND         A2514120
         MVC   LDENTA(3),9(LDCONT)       FETCH SYMBOL ORIGIN         A2514130
         BC    15,LDENT2                 *                           A2514140
         EJECT                                                       A2514150
*************************************************************************A2514160
*                                                                     * A2514170
*                        CHECK TABLES SUBROUTINE                      * A2514180
*                                                                     * A2514190
*                    NAME= LDTCK1                                     * A2514200
*                                                                     * A2514210
*                 CALLING SEQUENCE - BAL LEXIT1,LDTCK1                * A2514220
*                                                                     * A2514230
*  THIS ROUTINE CHECKS AND ADJUSTS THE LOADER TABLE BLOCK.            * A2514240
*                                                                     * A2514250
*  DURING  LOADER  INITIALIZATION  (SEE THAT ROUTINE), THE START AND  * A2514260
*  END OF THE DICTIONARY (LDDICA,LDDICZ) AND THE START OF THE REFER-  * A2514270
*  ENCE TABLE (LDREFA) ARE SET AT THE END OF THE TABLE BLOCK WHEREAS   * A2514280
*  THE  START  AND  END OF THE LIST TABLE (LDLSTA,LDLSTZ) ARE SET AT  * A2514290
*  THE START OF THE TABLE BLOCK.                                      * A2514300
*                                                                     * A2514310
*  THE LOADER ENTRY OR RESUME  ROUTINE SETS THE END OF THE REFERENCE  * A2514320
*  TABLE (LDREFZ) AT THE START SO AS TO CANCEL ANY ENTRIES THAT THIS  * A2514330
*  TABLE MAY CONTAIN AND CALLS UP THE PRESENT ROUTINE.                * A2514340
*                                                                     * A2514350
*  THE PRESENT ROUTINE IS ALSO CALLED UP EVERY TIME AN ENTRY IS MADE  * A2514360
*  IN ANY OF THE THREE TABLES IN THE TABLE BLOCK.                     * A2514370
*                                                                     * A2514380
*  THE GAP BETWEEN  THE FIRST TWO TABLES (LDELT1) IS CHECKED AND, IF  * A2514390
*  LESS THAN 80 BYTES, A MESSAGE IS WRITTEN AND LOADING STOPS  (WAIT  * A2514400
*  STATE).IF IT IS GREATER, THE GAP  BETWEEN  THE  LAST  TWO  TABLES  * A2514410
*  (LDELT2) IS TESTED AND,  IF LESS THAN 20 BYTES,  THE MIDDLE TABLE  * A2514420
*  IS SHIFTED TO THE LEFT BY A QUARTER OF THE EXISTING  GAP  BETWEEN  * A2514430
*  IT AND THE FIRST TABLE.  CONTROL THEN RETURNS TO THE CALLER.       * A2514440
*                                                                     * A2514450
*  NOTE-  ENTRIES  ARE  PLACED  CONSECUTIVELY  FROM LEFT TO RIGHT IN  * A2514460
*  TABLES LIST AND REFERENCE, BUT FROM RIGHT TO LEFT IN THE DICTION-  * A2514470
*  ARY.                                                               * A2514480
*                                                                     * A2514490
*************************************************************************A2514500
         SPACE                                                       A2514510
LDTCK1   L     LWK1RG,LDREFA             START OF REF. TABLE MINUS   A2514520
         S     LWK1RG,LDLSTZ             * END OF LIST TABLE IS LESS A2514530
         C     LWK1RG,LDELT1             * THAN REQU. GAP (80 BYTES) A2514540
         BC    4,LDTCKX                  YES - BRANCH                A2514550
         L     LWK2RG,LDDICZ             END OF DICT. MINUS END OF   A2514560
         S     LWK2RG,LDREFZ             * REF. TABLE IS LESS THAN   A2514570
         C     LWK2RG,LDELT2             * REQUIRED GAP (20 BYTES)   A2514580
         BC    10,LDTCKZ                 NO - BRANCH                 A2514590
         SRL   LWK1RG,4                  MOVE MIDDLE TABLE OF BLOCK  A2514600
         SLL   LWK1RG,2                  * (REF. TABLE) TO THE LEFT  A2514610
         LM    LDCONT,LDFINL,LDREFA      * BY ONE QUARTER OF THE GAP A2514620
```

```
         LR    LWK2RG,LDCONT              * BETWEEN THE FIRST TWO     A2514630
         SR    LWK2RG,LWK1RG              * TABLES AND ADJUST ADDRESS A2514640
         ST    LWK2RG,LDREFA              * OF START OF REFERENCE     A2514650
         BC    15,LDTCK3                  * TABLE                     A2514660
LDTCK2   AR    LWK2RG,LDSTEP              *                           A2514670
LDTCK3   MVC   0(8,LWK2RG),0(LDCONT)      *                           A2514680
         BXLE  LDCONT,LDSTEP,LDTCK2       *                           A2514690
         ST    LWK2RG,LDREFZ              STORE NEW END OF REF. TABLE A2514700
LDTCKZ   BCR   15,LEXIT1                  RETURN TO CALLER            A2514710
LDTCKX   BAL   LEXIT1,LMSDG1              WRITE MESSAGE 'INSUFFICIENT A2514720
         DC    C'1'                       * SPACE AVAILABLE FOR       A2514730
         DC    AL3(MSDG21)                * LOADER TABLES' AND STOP   A2514740
         BC    15,LDSTOP                  *                           A2514750
         SPACE 5                                                      A2514760
*************************************************************************A2514770
*                                                                     * A2514780
*                    ADDRESS RELOCATION SUBROUTINE                    * A2514790
*                                                                     * A2514800
*                   NAME= LDRLC1                                      * A2514810
*                                                                     * A2514820
*                CALLING SEQUENCE - LA  LEXIT2,------                 * A2514830
*                                   BAL LEXIT1,LDRLC1                 * A2514840
*                EXITS - LEXIT1 IF ADDRESS IS RELOCATED               * A2514850
*                        LEXIT2 IF ADDRESS IS NOT RELOCATED           * A2514860
*                                                                     * A2514870
*  THIS ROUTINE RELOCATES ASSEMBLY ADDRESSES. IF THE ABSOLUTE LOADER  * A2514880
*  FLAG IS ON, IT JUST CHECKS THAT  THE  LOCATION  COUNTER IS WITHIN  * A2514890
*  THE BOUNDS OF THE STORAGE AREA RESERVED FOR LOADING.               * A2514900
*                                                                     * A2514910
*  OTHERWISE IT LOOKS UP THE REFERENCE  TABLE AT THE ADDRESS  DETER-  * A2514920
*  MINED BY THE ESID NUMBER.  IF THE ELEMENT  AT  THIS  ADDRESS IS A  * A2514930
*  CONTROL SECTION, IT NEXT EXAMINES THE DICTIONARY ELEMENT GIVEN BY  * A2514940
*  THE REFTAB ELEMENT. IF THE CONTROL SECTION TYPE IN THE DICTIONARY  * A2514950
*  IS 'BEING LOADED',  THE ADDRESS IN THE WORKING  AREA IS RELOCATED  * A2514960
*  BY  ADDING  TO  IT THE RELOCATED  ADDRESS OF THE CONTROL  SECTION  * A2514970
*  FOUND IN THE DICTIONARY AND THEN SUBTRACTING THE ASSEMBLY ADDRESS  * A2514980
*  OF THE CONTROL SECTION FOUND IN REFTAB.   CONTROL THEN RETURNS TO  * A2514990
*  THE CALLER.                                                        * A2515000
*                                                                     * A2515010
*  IN ALL OTHER CASES RELOCATION IS NOT POSSIBLE AND CONTROL RETURNS  * A2515020
*  TO THE CALLER AT ANOTHER ENTRY THAN WHEN RELOCATION IS EFFECTED.   * A2515030
*                                                                     * A2515040
*************************************************************************A2515050
         SPACE                                                        A2515060
LDRLC1   NI    LWZTYP,X'00'               CLEAR LWZTYP                A2515070
         TM    LDABSL,X'80'               ABSOLUTE LOADER FLAG        A2515080
         BC    1,LDRLCA                   YES - BRANCH                A2515090
LDRLCC   NI    LDFLAG,X'F7'               CLEAR 'ESD ID = 0' FLAG     A2515100
         LH    LWK1RG,LWZCSI              PICK UP ESD ID              A2515110
         LTR   LWK1RG,LWK1RG              IS ESD ID = 0               A2515120
         BC    8,LDRLC2                   YES - BRANCH                A2515130
         BCTR  LWK1RG,0                   COMPUTE ADDR. OF ELEMENT IN A2515140
         SLL   LWK1RG,3                   * REF. TABLE (ESD ID X 8)   A2515150
         BC    15,*+8                     * AND SKIP NEXT INSTRUCTION A2515160
LDRLC2   OI    LDFLAG,X'08'               SET FLAG 'ESID ID = 0'      A2515170
```

```
         A     LWK1RG,LDREFA              ADDR. FIRST ELEM. IN REFTAB A2515180
         LM    LWK2RG,LWK3RG,LDREFA+4     INITIALIZE REFTAB SEARCH    A2515190
LDRLC3   BXH   LWK1RG,LWK2RG,0(LEXIT2)    END OF TABLE REACHED        A2515200
         MVC   LWZREF(8),0(LWK1RG)        READ OUT DBLE WD FRM REFTAB A2515210
         TM    LWZTIP,X'03'               TYPE IN REF TABLE IS C.S.   A2515220
         BC    8,LDRLC5                   YES - BRANCH                A2515230
LDRLC4   TM    LDFLAG,X'08'               IS FLAG 'ESD ID = 0' ON     A2515240
         BC    1,LDRLC3                   YES - GO TO NEXT ELEMENT    A2515250
         BCR   15,LEXIT2                  EXIT (NOT RELOCATED)        A2515260
LDRLC5   L     LWK4RG,LWZREF              GET ADDR. OF C.S. IN DICT.  A2515270
         TM    8(LWK4RG),X'04'            DICT.SHOWS'C.S.ALREADY      A2515280
         BC    10,LDRLC6                  * LOADED'. YES-BRANCH       A2515290
         TM    LDINIT,X'04'               TEST IF END ADDR RELOCATION A2515300
         BCR   1,LEXIT2                   NO-EXIT(NOT RELOCATED)      A2515310
LDRLC6   CLC   LWZORG(3),LWZADR           ADDR.LOWER THAN START OF CS A2515320
         BC    4,LDRLC4                   YES-BRANCH                  A2515330
         LH    LWK5RG,6(LWK4RG,0)         C.S.LENGTH FROM DICT.AND    A2515340
         A     LWK5RG,LWZADR-1            * ASSEMBLY ADDR FROM REFTAB A2515350
         CL    LWK5RG,LWZORG-1            IS ADDR.BEYOND END OF C.S.  A2515360
         BC    12,LDRLC4                  YES-BRANCH                  A2515370
         L     LWK5RG,8(LWK4RG,0)         COMPUTE RELOCATED ADDRESS   A2515380
         SLL   LWK5RG,8                   * (LOAD RELOCATED ADDR.OF   A2515390
         SRL   LWK5RG,8                   * C.S.,CLEAR FIRST BYTE,ADD A2515400
         A     LWK5RG,LWZORG-1            * ASSEMBLY ADDR.TO BE       A2515410
         S     LWK5RG,LWZADR-1            * RELOCATED LESS ASSEMBLY   A2515420
         ST    LWK5RG,LWZORG-1            * ADDRESS OF CONTROL SECT.  A2515430
         BCR   15,LEXIT1                  EXIT(RELOCATED)             A2515440
LDRLCA   L     LWK5RG,LWZORG-1            CHECK THAT ADDRESS IS LOWER A2515450
         CL    LWK5RG,LDTOPP              * THAN START OF TABLE BLOCK A2515460
         BCR   10,LEXIT2                  NO-EXIT(NOT RELOCATED)      A2515470
         CL    LWK5RG,LOCCTR              ADDRESS LOWER THAN LOCCTR   A2515480
         BCR   10,LEXIT1                  NO - EXIT(RELOCATED)        A2515490
         BCR   15,LEXIT2                  YES - EXIT (NOT RELOCATED)  A2515500
         EJECT                                                        A2515510
*************************************************************************A2515520
*                                                                      * A2515530
*              LOOK-UP SYMBOL IN DICTIONARY SUBROUTINE                 * A2515540
*                                                                      * A2515550
*                  NAME= LDMOR1                                        * A2515560
*                                                                      * A2515570
*               CALLING SEQUENCE - LA  LEXIT2,------                   * A2515580
*                                  BAL LEXIT1,LDMOR1                   * A2515590
*               EXITS - LEXIT1 IF SYMBOL IS FOUND                      * A2515600
*                       LEXIT2 IF SYMBOL IS NOT FOUND                  * A2515610
*                                                                      * A2515620
*  THIS ROUTINE LOOKS FOR A SYMBOL IN THE DICTIONARY.  IF THE SYMBOL   * A2515630
*  IS FOUND,  CONTROL RETURNS TO CALLER,  BUT IF NOT, A NEW ENTRY IS   * A2515640
*  MADE IN THE DICTIONARY FOR THIS SYMBOL.                             * A2515650
*                                                                      * A2515660
*  THIS ROUTINE IS ENTERED AT LDMOR2 WHEN A SYMBOL MUST  BE  ENTERED   * A2515670
*  IN THE DICTIONARY AND NO LOOK-UP IS REQUIRED.                       * A2515680
*                                                                      * A2515690
*************************************************************************A2515700
         SPACE                                                        A2515710
LDMOR1   LM    LDCONT,LDFINL,LDDICA       INITIALIZE DICT. LOOK-UP    A2515720
```

```
LDCOMP   BXLE  LDCONT,LDSTEP,LNOTFD        BR. IF DICT. EXHAUSTED      A2515730
         CLC   0(6,LDCONT),LESNME          ARE SYMBOLS IDENTICAL       A2515740
         BC    7,LDCOMP                    NO - GET NEXT SYMBOL        A2515750
         BCR   15,LEXIT1                   EXIT (SYMBOL FOUND)         A2515760
LDMOR2   LM    LDCONT,LDFINL,LDDICA        INITIALIZE DICT. LOOK-UP    A2515770
         BXH   LDCONT,LDSTEP,*             GO TO END OF DICTIONARY     A2515780
LNOTFD   MVC   0(6,LDCONT),LESNME          STORE SYMBOL IN DICT.       A2515790
         AR    LDSTEP,LDCONT               GO TO NEXT ELEMENT POSITION A2515800
         ST    LDSTEP,LDDICZ               UPDATE ADDR.OF END OF DICT. A2515810
         BCR   15,LEXIT2                   EXIT(SYMBOL NOT FOUND)      A2515820
         EJECT                                                         A2515830
*************************************************************************A2515840
*                                                                      * A2515850
*                     RESERVE STORAGE SUBROUTINE                       * A2515860
*                                                                      * A2515870
*                    NAME= LDRES1                                      * A2515880
*                                                                      * A2515890
*                 CALLING SEQUENCE - BAL LEXIT3,LDRES1                 * A2515900
*                                                                      * A2515910
*  THIS ROUTINE ADJUSTS THE LOCATION COUNTER TO A DOUBLE WORD BOUND-   * A2515920
*  ARY, STORES IT IN THE DICTIONARY AS THE  STARTING  ADDRESS OF THE   * A2515930
*  SECTION TOGETHER WITH THE LENGTH OF THE SECTION UNDER THE CONTROL    * A2515940
*  SECTION SYMBOL.  THE LOCATION  COUNTER IS THEN INCREMENTED BY THE   * A2515950
*  LENGTH OF THE SECTION  AND  CHECKED  THAT IT DOES  NOT EXCEED THE   * A2515960
*  AVAILABLE  STORAGE.  IF IT DOES, A MESSAGE IS WRITTEN AND LOADING   * A2515970
*  STOPS (WAIT STATE). OTHERWISE A MESSAGE  GIVES THE NAME AND START   * A2515980
*  ADDRESS OF THE CONTROL SECTION, IF THE LOADER PRINTS THE  LOADING   * A2515990
*  CONTROL MESSAGES, AND CONTROL IS RETURNED TO THE CALLER.            * A2516000
*                                                                      * A2516010
*************************************************************************A2516020
         SPACE                                                         A2516030
LDRES1   L     LWK1RG,LOCCTR               CHECK VALUE OF LOCATION     A2516040
LDRES2   TM    LOCCTR+3,X'07'              * COUNTER AND INCREMENT     A2516050
         BC    8,LDRES3                    * BY STEPS OF 1 UNTIL IT    A2516060
         LA    LWK1RG,1(LWK1RG,0)          * IS A MULTIPLE OF 8        A2516070
         ST    LWK1RG,LOCCTR               *                           A2516080
         BC    15,LDRES2                   *                           A2516090
LDRES3   MVC   6(2,LDCONT),LSECLG          STORE CONTR.SECT.LENGTH     A2516100
         ST    LWK1RG,8(LDCONT,0)          * ORIGIN AND TYPE IN        A2516110
         NI    8(LDCONT),X'00'             * DICTIONARY                A2516120
         AH    LWK1RG,LSECLG               UPDATE LOCCTR BY LENGTH OF  A2516130
         C     LWK1RG,LDTOPP               * SEGMENT AND CHECK IF IT   A2516140
         ST    LWK1RG,LOCCTR               * OVERLAPS THE TABLE BLOCK  A2516150
         BC    10,LDRES6                   YES-BRANCH                  A2516160
LDRES4   MVC   PRTBUF(6),0(LDCONT)         TRANSFER CONTROL SECTION    A2516170
         XC    PRTBUF+6(2),PRTBUF+6        * NAME TO PRINT BUFFER      A2516180
         LA    LEXIT1,9(LDCONT,0)          CONVERT LOADING ADDRESS     A2516190
LDRES5   L     LEXIT2,AHSDGK               * TO HEXADECIMAL            A2516200
         ST    LEXIT3,SAVREG               *                           A2516210
         BAL   LEXIT3,LBHEX1               *                           A2516220
         BAL   LEXIT1,LPRINT               WRITE MESSAGE 'XXXXXX CON-  A2516230
         DC    C' '                        * TROL SECTION LOADED AT    A2516240
         DC    AL3(MSDG1K)                 * YYYYYY'                   A2516250
         L     LEXIT3,SAVREG               *                           A2516260
         BCR   15,LEXIT3                   EXIT TO CALLER              A2516270
```

```
LDRES6   BAL   LEXIT1,LMSDG1              WRITE MESSAGE 'INSUFFICIENT A2516280
         DC    C'1'                       * STORAGE AVAILABLE FOR     A2516290
         DC    AL3(MSDG23)                * THIS PROGRAM'             A2516300
         BC    15,LDSTOP                  EXIT TO DUMP AND STOP       A2516310
         EJECT                                                        A2516320
*************************************************************************A2516330
*                                                                      * A2516340
*               RELOCATE ENTRIES IN DICT. SUBROUTINE                   * A2516350
*                                                                      * A2516360
*                  NAME= LDSTA-                                        * A2516370
*                                                                      * A2516380
*               CALLING SEQUENCE - BAL LEXIT3,LDSTA-                   * A2516390
*                                                                      * A2516400
*  THIS ROUTINE EXAMINES ALL THE ELEMENTS IN THE DICTIONARY TO RELO-   * A2516410
*  CATE ANY OUTSTANDING  ENTRIES  WHERE POSSIBLE.  IF THE ROUTINE IS   * A2516420
*  ENTERED FROM THE END CARD  PROCESSING  ROUTINE, THE STATUS OF THE   * A2516430
*  CONTROL SECTION BEING LOADED IS CHANGED TO 'ALREADY LOADED'.        * A2516440
*                                                                      * A2516450
*  IF THE  ENTRIES  CANNOT BE RELOCATED,  A MESSAGE IS WRITTEN,  THE   * A2516460
*  ASSEMBLY ADDRESS IS CLEARED AND THE TYPE IS SET TO 'ENTRY RELOCA-   * A2516470
*  TED' TO AVOID  TREATING  THIS  ENTRY  AGAIN  THE  NEXT TIME  THIS   * A2516480
*  ROUTINE IS CALLED.                                                  * A2516490
*                                                                      * A2516500
*************************************************************************A2516510
         SPACE                                                        A2516520
LDSTAC   NI    LDSTAT,X'F3'               CLEAR FLAGS'TXT,REP,RLD     A2516530
         BC    15,*+8                     * READ','END CRD NOT FOUND' A2516540
LDSTAB   OI    LDSTAT,X'0D'               SET ABOVE FLAGS             A2516550
         LM    LDCONT,LDFINL,LDDICA       INITIALIZE DICTIONARY SCAN  A2516560
LDSTA1   BXLE  LDCONT,LDSTEP,0(LEXIT3)    EXIT IF DICT. IS EXHAUSTED  A2516570
         TM    8(LDCONT),X'01'            ELEMENT IS ENTRY            A2516580
         BC    8,LDSTA4                   NO - BRANCH                 A2516590
         TM    8(LDCONT),X'05'            ENTRY IS RELOCATED          A2516600
         BC    1,LDSTA1                   YES - BRANCH                A2516610
         MVC   LWZDIC(6),6(LDCONT)        MOVE ELEMENT TO DIC.WK.AREA A2516620
         LA    LEXIT2,LDSTA3              RELOCATE ENTRY AND SET      A2516630
         BAL   LEXIT1,LDRLC1              * 'ENTRY RELOCATED' IN      A2516640
LDSTA2   OI    LWZTYP,X'05'               * DICTIONARY WORK AREA      A2516650
         MVC   6(6,LDCONT),LWZDIC         MOVE ELEMENT BACK TO DICT.  A2516660
         BC    15,LDSTA1                  GO TO NEXT ELEMENT          A2516670
LDSTA3   XC    LWZORG(3),LWZORG           CLEAR ASSEMBLY ADDRESS      A2516680
         MVC   PRTBUF(6),0(LDCONT)        STORE DICT. SYMBOL IN       A2516690
         XC    PRTBUF+6(2),PRTBUF+6       * PRINT BUFFER              A2516700
         ST    LEXIT3,SAVREG              SAVE RETURN ADDRESS         A2516710
         BAL   LEXIT1,LMSDG1              WRITE MESSAGE 'ENTRY POINT  A2516720
         DC    C' '                       * NOT RELOCATABLE'          A2516730
         DC    AL3(MSDG1L)                *                           A2516740
         L     LEXIT3,SAVREG              RESTORE RETURN ADDRESS      A2516750
         BC    15,LDSTA2                  CONTINUE                    A2516760
LDSTA4   TM    LDFLAG,X'08'               FLAG 'TKT,REP,RLD READ' ON  A2516770
         BC    1,LDSTA1                   YES - GET NEXT ELEMENT      A2516780
         TM    8(LDCONT),X'08'            DIC.SHOWS'C.S.BEING LOADED' A2516790
         BC    12,LDSTA1                  NO - GET NEXT ELEMENT       A2516800
         XI    8(LDCONT),X'0C'            SET 'C.S. LOADED' IN DICT.  A2516810
         BC    15,LDSTA1                  GO TO NEXT ELEMENT          A2516820
```

```
         EJECT                                                                  A2516830
*************************************************************************** A2516840
*                                                                         * A2516850
*            CALCULATION OF RLD EXPRESSION VALUE SUBROUTINE               * A2516860
*                                                                         * A2516870
*                     NAME= LDLST1                                        * A2516880
*                                                                         * A2516890
*                CALLING SEQUENCE - LA  LEXIT2,------                     * A2516900
*                                   BAL LEXIT1,LDLST1                     * A2516910
*                EXITS - LEXIT1 IF ELEMENT IS PROCESSED                   * A2516920
*                        LEXIT2 IF ELEMENT IS NOT PROCESSED               * A2516930
*                                                                         * A2516940
*  THIS ROUTINE CALCULATES THE REAL VALUE OF THE CONSTANT (SYMBOL+N)       * A2516950
*  TO BE LOADED AT ADDRESS ADSYMB.                                        * A2516960
*                                                                         * A2516970
*  AN ANALYSIS OF THE DICTIONARY AND REFERENCE TABLE ELEMENTS RELAT-       * A2516980
*  ED TO THE SYMBOL RESULTS AS FOLLOWS-                                   * A2516990
*                                                                         * A2517000
*  1. DICTIONARY ELEMENT IS EXTERNAL - NO CALCULATION POSSIBLE.           * A2517010
*                                                                         * A2517020
*  2. DICTIONARY ELEMENT IS C.S. OR ENTRY AND REFTAB ELEMENT IS C.S.      * A2517030
*     (ESID = 1, I.E. SYMBOL  BELONGS TO SECTION) - THE ASSEMBLY AD-      * A2517040
*     RESS OF THE CONTROL SECTION FOUND IN REFTAB IS SUBTRACTED FROM      * A2517050
*     THE  RELOCATED  ADDRESS  IN  THE  DICTIONARY AND THE RESULT IS      * A2517060
*     ADDED TO OR SUBTRACTED FROM THE INITIAL VALUE OF THE CONSTANT.      * A2517070
*                                                                         * A2517080
*  3. DICTIONARY ELEMENT IS C.S. OR ENTRY AND REFTAB  ELEMENT IS EX-      * A2517090
*     TERNAL (ESID = 2 TO 15) - THE RELOCATED ADDRESS OF THE C.S. OR      * A2517100
*     ENTRY IN DICTIONARY IS ADDED TO OR SUBTRACTED FROM THE INITIAL      * A2517110
*     VALUE OF THE CONSTANT.                                              * A2517120
*                                                                         * A2517130
*  4. DICTIONARY  ELEMENT IS C.S. OR ENTRY  AND  THERE  IS NO REFTAB      * A2517140
*     ELEMENT.  THIS MEANS  THAT  THE ROUTINE HAS BEEN CALLED BY THE      * A2517150
*     END CARD PROCESSING  ROUTINE AND THAT THE  REFTAB  ELEMENT HAS      * A2517160
*     BEEN TRANSFERRED TO TABLE 'LIST' - TREATMENT AS FOR (3) ABOVE.      * A2517170
*                                                                         * A2517180
*************************************************************************** A2517190
         SPACE                                                              A2517200
LDLST1   LM    LWK1RG,LWK2RG,LWZLST      SYMB.ADDR.IN REFTAB AND DIC       A2517210
         TM    8(LWK2RG),X'02'           DICT. ELEMENT IS EXTERNAL         A2517220
         BCR   1,LEXIT2                  YES - EXIT (NOT PROCESSED)        A2517230
         MVC   LWZDIC(6),6(LWK2RG)       RELOCATED ADDR TO WORK AREA       A2517240
         NI    LWZTYP,X'00'              CLEAR TYPE IN DIC.WORK AREA       A2517250
         L     LWK2RG,LWZORG-1           RELOCATED ADDR OF SYMBOL          A2517260
         LTR   LWK1RG,LWK1RG             END CARD BEING PROCESSED          A2517270
         BC    8,LDLST2                  YES-BRANCH                        A2517280
         TM    4(LWK1RG),X'02'           TYPE IN REFTAB IS EXTERNAL        A2517290
         BC    1,LDLST2                  YES-BRANCH                        A2517300
         S     LWK2RG,4(LWK1RG,0)        SUBTRACT C.S.ASSEMBLY ADDR        A2517310
LDLST2   L     LWK1RG,LWZLAD-1           ADDR.OF CONST.TO BE CALC.         A2517320
         NI    LDMVC1+1,X'00'            CLEAR LENGTH IN  MOVE             A2517330
         NI    LDMVC2+1,X'00'            * INSTRUCTIONS                    A2517340
         LH    LWK4RG,LDMVC2+4           SAVE ADDR. OF WORKING AREA        A2517350
         TM    LWZFLG,X'0C'              CONSTANT LENGTH = 4 BYTES         A2517360
         BC    1,LDLST3                  YES - BRANCH                      A2517370
```

```
         XC    LWZLCT(4),LWZLCT          CLEAR WORKING AREA          A2517380
         TM    LWZFLG,X'08'              CONSTANT LENGTH=3 BYTES     A2517390
         BC    1,LDLST4                  YES - BRANCH                A2517400
         TM    LWZFLG,X'04'              CONSTANT LENGTH = 2 BYTES   A2517410
         BC    1,LDLST5                  YES - BRANCH                A2517420
         LA    LWK6RG,3(LWK4RG,0)        MOVE WILL START AT LWZLCT+3 A2517430
         BC    15,LDLST6                 *                           A2517440
LDLST3   OI    LDMVC1+1,X'03'            INSERT LENGTH '3' IN MVC    A2517450
         OI    LDMVC2+1,X'03'            *                           A2517460
         BC    15,LDMVC1                 GO TO MOVE                  A2517470
LDLST4   MVI   LDMVC1+1,X'02'            INSERT LENGTH '2' IN MVC    A2517480
         MVI   LDMVC2+1,X'02'            *                           A2517490
         LA    LWK6RG,1(LWK4RG,0)        MOVE WILL START AT LWZLCT+1 A2517500
         BC    15,LDLST6                 *                           A2517510
LDLST5   MVI   LDMVC1+1,X'01'            INSERT LENGTH '1' IN MUC    A2517520
         MVI   LDMVC2+1,X'01'            *                           A2517530
         LA    LWK6RG,2(LWK4RG,0)        MOVE WILL START AT LWZLCT+2 A2517540
LDLST6   STH   LWK6RG,LDMVC1+2           INSERT LWZLCT+N IN MOVE     A2517550
         STH   LWK6RG,LDMVC2+4           * INSTRUCTION               A2517560
LDMVC1   MVC   LWZLCT(1),0(LWK1RG)       CONST.INIT.VAL.TO LWZLCT+N  A2517570
         L     LWK5RG,LWZLCT             * AND THEN TO WORK REGISTER A2517580
         TM    LWZFLG,X'02'              TEST COMPLEMENT FLAG        A2517590
         BC    1,LDLST7                  ADD OR SUBTRACT RELOCATION  A2517600
         AR    LWK5RG,LWK2RG             * QUANTITY AND PLACE        A2517610
         BC    15,*+6                    * RESULT IN WORKING AREA    A2517620
LDLST7   SR    LWK5RG,LWK2RG             *                           A2517630
         ST    LWK5RG,LWZLCT             *                           A2517640
LDMVC2   MVC   0(1,LWK1RG),LWZLCT        PLACE CONST.AT CORRECT ADDR A2517650
         STH   LWK4RG,LDMVC1+2           RESTORE WORKING AREA ADDR.  A2517660
         STH   LWK4RG,LDMVC2+4           * TO MOVE INSTRUCTIONS      A2517670
         BCR   15,LEXIT1                 EXIT1=LOAD CONST PROCESSED  A2517680
         EJECT                                                       A2517690
*************************************************************************A2517700
*                                                                       * A2517710
*            HEXADECIMAL TO BINARY CONVERSION SUBROUTINE                * A2517720
*                                                                       * A2517730
*                  NAME= LHEXB1                                         * A2517740
*                                                                       * A2517750
*       CALLING SEQUENCE - L   LEXIT1,START OF FIELD ADDRESS            * A2517760
*                          L   LEXIT2,LENGTH OF FIELD (BYTES)           * A2517770
*                          BAL LEXIT3,LHEXB1                            * A2517780
*                                                                       * A2517790
*  THE  CHARACTER  VALIDITY  CHECK  ALSO  DETERMINES WHETHER A VALID    * A2517800
*  CHARACTER IS ALPHABETIC (A-F) OR NUMERIC (1-9)                       * A2517810
*                                                                       * A2517820
*  IN  THE  FIRST  CASE, SUBTRACTING 'B7' AMOUNTS TO ADDING 9 TO THE    * A2517830
*  NUMERIC AND  SUBTRACTING 192 FROM THE ZONE, THUS LEAVING THE COR-    * A2517840
*  RECT BINARY VALUE OF THE  HEXADECIMAL  CHARACTER  IN  THE  SECOND    * A2517850
*  HALF-BYTE.                                                           * A2517860
*                                                                       * A2517870
*  IN THE SECOND CASE, ONE NEED ONLY CLEAR THE ZONE.                    * A2517880
*                                                                       * A2517890
*************************************************************************A2517900
         SPACE                                                       A2517910
LHEXB1   SR    LWK3RG,LWK3RG             WORKING REGISTERS           A2517920
```

```
         LR    LWK1RG,LWK3RG             *                              A2517930
         LR    LWK2RG,LWK3RG             *                              A2517940
         IC    LWK1RG,LCSTB7             INSERT 183 (DECIMAL)           A2517950
LHEXB2   TM    0(LEXIT1),X'F0'           BYTE VALUE SMALLER THAN F0     A2517960
         BC    1,LHEXB3                  NO-BRANCH                      A2517970
         CLI   0(LEXIT1),X'C0'           BYTE VALUE GREATER THAN C0     A2517980
         BC    12,LHEXB5                 NO-BRANCH                      A2517990
         CLI   0(LEXIT1),X'C6'           BYTE VALUE GREATER THAN C6     A2518000
         BC    2,LHEXB5                  YES-BRANCH                     A2518010
         IC    LWK2RG,0(LEXIT1,0)        BYTE TO WORKING REGISTER       A2518020
         SR    LWK2RG,LWK1RG             CONVERT TO ALPHA               A2518030
LHEXB4   SLL   LWK3RG,4                  MULTIPLY PREVIOUS RESULT BY    A2518040
         AR    LWK3RG,LWK2RG             * 16 AND ADD PRESENT RESULT    A2518050
         LA    LEXIT1,1(LEXIT1,0)        ADDRESS OF NEXT BYTE           A2518060
         BCT   LEXIT2,LHEXB2             BRANCH IF ANY BYTES LEFT       A2518070
         LR    LEXIT2,LWK3RG             TRANSFER FINAL RESULT          A2518080
         BCR   15,LEXIT3                 RETURN TO CALLER               A2518090
LHEXB3   CLI   0(LEXIT1),X'F9'           BYTE VALUE GREATER THAN F9     A2518100
         BC    2,LHEXB5                  YES-BRANCH                     A2518110
         NI    0(LEXIT1),X'0F'           CLEAR ZONE                     A2518120
         IC    LWK2RG,0(LEXIT1,0)        BYTE TO WORKING REGISTER       A2518130
         BC    15,LHEXB4                 * AND CONTINUE                 A2518140
LHEXB5   BAL   LEXIT1,LMSDG1             WRITE MESSAGE 'CHARACTER IN    A2518150
         DC    C' '                      * CARD NOT HEXADECIMAL'        A2518160
         DC    AL3(MSDG1H)               *                              A2518170
         BC    15,LDCARD                 GET NEXT CARD                  A2518180
         EJECT                                                          A2518190
*********************************************************************** A2518200
*                                                                     * A2518210
*            BINARY TO HEXADECIMAL CONVERSION SUBROUTINE              * A2518220
*                                                                     * A2518230
*                  NAME= LBHEX1                                       * A2518240
*                                                                     * A2518250
*   CALLING SEQUENCE - LA  LEXIT1,ADDRESS OF BINARY FIELD             * A2518260
*                      L   LEXIT2, BITS  0-23 ADDR. OF RESULT FIELD   * A2518270
*                                  BITS 24-31 2 X(BYTE COUNT- 1)      * A2518280
*                      BAL LEXIT3,LBHEX1                                A2518290
*                                                                     * A2518300
*  EACH BINARY CHARACTER, OCCUPYING ONE HALF-BYTE, IS TRANSLATED VIA  * A2518310
*  A TABLE TO A FULL-BYTE HEXADECIMAL CHARACTER.                      * A2518320
*                                                                     * A2518330
*********************************************************************** A2518340
         SPACE                                                          A2518350
LBHEX1   SR    LWK4RG,LWK4RG             CLEAR WORKING REGISTER         A2518360
         STC   LEXIT2,LBHEX2+3           HEXADECIMAL BYTE COUNT         A2518370
         SRL   LEXIT2,8                  CLEAR BYTE COUNT IN REGISTE    A2518380
         LR    LWK1RG,LEXIT2             ADDR OF DESTINATION FIELD      A2518390
         LA    LWK2RG,2                  LOAD'2' IN REGISTER (STEP)     A2518400
LBHEX2   LA    LWK3RG,0(LEXIT2,0)        END ADDR.OF DESTIN.FIELD       A2518410
LBHEX3   LA    LWK5RG,15                 LOAD MASK 'OF'                 A2518420
         IC    LWK4RG,0(LEXIT1,0)        LOAD ONE BYTE                  A2518430
         NR    LWK5RG,LWK4RG             2ND HALF BYTE IN REG.5         A2518440
         SRL   LWK4RG,4                  CLEAR 2ND HALF BYTE IN REG.    A2518450
         IC    LWK4RG,BHXTAB(LWK4RG)     CONVERT 1ST HALF-BYTE          A2518460
         STC   LWK4RG,0(LWK1RG,0)        * AND STORE                    A2518470
```

```
         IC    LWK5RG,BHXTAB(LWK5RG)       CONVERT 2ND HALF-BYTE       A2518480
         STC   LWK5RG,1(LWK1RG,0)          * AND STORE                 A2518490
         LA    LEXIT1,1(LEXIT1,0)          UPDATE ADDR OF BINARY OPND  A2518500
         BXLE  LWK1RG,LWK2RG,LBHEX3        RETURN IF ANY BYTES LEFT    A2518510
         LR    LEXIT2,LWK1RG               1ST BYTE OF FOLLOWING FIELD A2518520
         BCR   15,LEXIT3                   RETURN TO CALLER            A2518530
BHXTAB   DC    C'0123456789ABCDEF'         BIN-HEX CONVERSION TABLE    A2518540
         EJECT                                                         A2518550
*********************************************************************** A2518560
*                                                                     * A2518570
*                            CONSTANTS AREA                           * A2518580
*                                                                     * A2518590
*********************************************************************** A2518600
         SPACE                                                         A2518610
         DS    0D                          INITIAL PSW (AFTER LOADING) A2518620
LDTPSW   DC    X'FF'                       *I/O,EXT.INTERRUPTS ENABLED A2518630
         DC    X'05'                       *PROBLEM STATE              A2518640
         DC    X'00000F'                   *PROGR. INTERRUPTS ENABLED  A2518650
LDENTA   DC    XL3'0'                      BRANCH ADDR.(AFTER LOADING) A2518660
*                                                                      A2518670
LTYPLD   DC    X'02'                       CARD TYPES AND SPECIFIC     A2518680
         DC    C'TXT'                      * ROUTINE ADDRESSES         A2518690
         DC    A(LDTXT1)                   *                           A2518700
         DC    X'02'                       *                           A2518710
         DC    C'ESD'                      *                           A2518720
         DC    A(LDESD1)                   *                           A2518730
         DC    X'02'                       *                           A2518740
         DC    C'RLD'                      *                           A2518750
         DC    A(LDRLD1)                   *                           A2518760
LDEND0   DC    X'02'                       *                           A2518770
         DC    C'END'                      *                           A2518780
         DC    A(LDEND1)                   *                           A2518790
         DC    X'02'                       *                           A2518800
         DC    C'ICS'                      *                           A2518810
         DC    A(LDICS1)                   *                           A2518820
         DC    X'02'                       *                           A2518830
         DC    C'SLC'                      *                           A2518840
         DC    A(LDSLC1)                   *                           A2518850
         DC    X'02'                       *                           A2518860
         DC    C'REP'                      *                           A2518870
         DC    A(LDREP1)                   *                           A2518880
         DC    X'02'                       *                           A2518890
         DC    C'LDT'                      *                           A2518900
         DC    A(LDLDT1)                   *                           A2518910
LDCST1   DC    F'0'                        START OF TABLE LTYPLD       A2518920
         DC    F'8'                        LENGTH OF LTYPLD ELEMENT    A2518930
         DC    F'64'                       SIZE OF LTYPLD TABLE        A2518940
*                                                                      A2518950
LDOPTA   DC    A(0)                        ADDR OF START OF OPTION TBL A2518960
         DC    F'-8'                       LNGTH OF OPTION TBL ELEMENT A2518970
LDOPTZ   DC    A(0)                        ADDR OF END OF OPTION TABLE A2518980
LDDICA   DC    F'0'                        ADDR OF START OF DICT.      A2518990
         DC    F'-12'                      LNGTH OF DICTIONARY ELEMENT A2519000
LDDICZ   DC    F'0'                        ADDR. OF END OF DICTIONARY  A2519010
LDLSTA   DC    F'0'                        ADDR OF START OF LIST TABLE A2519020
```

```
         DC    F'8'                    LNGTH OF LIST TABLE ELEMENT A2519030
LDLSTZ   DC    F'0'                    ADDR OF END OF LIST TABLE   A2519040
LDREFA   DC    F'0'                    ADDR OF START OF REF. TABLE A2519050
         DC    F'8'                    LNGTH OF REF. TABLE ELEMENT A2519060
LDREFZ   DC    F'0'                    ADDR OF END OF REF. TABLE   A2519070
TBSIZE   DC    F'104'                  LOADER TABLE BLOCK SIZE     A2519080
LDELT1   DC    F'80'                   GAP BETWEEN REF AND LIST    A2519090
LDELT2   DC    F'20'                   GAP BETWEEN REF AND DICT    A2519100
LDSWS    DC    X'00'                   CONTROL FLAGS               A2519110
LDSTAT   EQU   LDSWS                   BITS 6,7 LOADING STATUS     A2519120
*                                      * 00  LOADING STAGE A       A2519130
*                                      * 10     -      -  B       A2519140
*                                      * 11     -      -  C       A2519150
LDINIT   EQU   LDSWS                   BIT 5 END CARD FOUND (IF 0) A2519160
LDFLAG   EQU   LDSWS                   BIT 4 GENERAL PURPOSE FLAG  A2519170
LDENTR   EQU   LDSWS                   BIT 3 END CARD ADDR SAVED   A2519180
LDHXAD   EQU   LDSWS                   BIT 2 NO HEXADEC. ADDRESS   A2519190
*                                      *     IN SLC CARD           A2519200
LDFTTR   EQU   LDSWS                   BIT 1 1ST TIME TRANSFER     A2519210
LDABSL   EQU   LDSWS                   BIT 0 ABSOLUTE LOADER FLAG  A2519220
LCSTB7   DC    X'B7'                   183 DECIMAL                 A2519230
*                                                                  A2519240
LWZDIC   DC    XL6'0'                  DICTIONARY WORKING AREA     A2519250
LWZTYP   EQU   LWZDIC+2                TYPE                        A2519260
LWZORG   EQU   LWZDIC+3                ORIGIN                      A2519270
LWZCSI   EQU   LWZDIC                  CONTROL SECTION ID OR       A2519280
LWZCSL   EQU   LWZDIC                  CONTROL SECTION LENGTH      A2519290
*                                                                  A2519300
LDAREA   DC    A(24)                   ADDRESS OF PROGRAM TO BE    A2519310
         DC    F'56'                   * REGENERATED.              A2519320
LOCCTR   DS    F                       LOCATION COUNTER            A2519330
LDTOPP   DS    F                       ADDR. OF LOADING TBLE BLOCK A2519340
LWZREF   DS    2F                      REFERENCE WORKING AREA      A2519350
LWZTIP   EQU   LWZREF+4                TYPE                        A2519360
LWZADR   EQU   LWZREF+5                ADDRESS (AS ASSEMBLED)      A2519370
LWZLST   DS    4F                      RELOCATION LIST WORK AREA   A2519380
LWZLDC   EQU   LWZLST+4                ENTRY TO DICTIONARY         A2519390
LWZFLG   EQU   LWZLST+8                RLD FLAGS (COMPL/CONTIN.)   A2519400
LWZLAD   EQU   LWZLST+9                ADDRESS                     A2519410
LWZLCT   EQU   LWZLST+12               RELOCATED ADDRESS           A2519420
         SPACE                                                     A2519430
*****************************************************************  A2519440
*                                                               *  A2519450
*                          EQUIVALENTS                          *  A2519460
*                                                               *  A2519470
*****************************************************************  A2519480
         SPACE                                                     A2519490
LTYPCD   EQU   LDXBUF                  LDXBUF DEFINED IN LDREAD    A2519500
LESTYP   EQU   LDXBUF+24               ESD - ESD TYPE (0,1 OR 2)   A2519510
LESNME   EQU   LDXBUF+16               ESD - NAME                  A2519520
LESADR   EQU   LDXBUF+25               ESD - ADDRESS               A2519530
LSECID   EQU   LDXBUF+30               ESD - ES ID OF C.S.         A2519540
LSECLG   EQU   LDXBUF+30               ESD - LENGTH OF C.S.        A2519550
LESDCT   EQU   LDXBUF+10               ESD - NBR OF BYTES IN CARD  A2519560
LESDID   EQU   LDXBUF+14               ESD - E.S IDENTIFICATION    A2519570
```

```
LTXTAD   EQU   LDXBUF+5                  TXT - ASSEMBLY ADDRESS       A2519580
LTXTCT   EQU   LDXBUF+10                 TXT - NBR OF BYTES IN TEXT   A2519590
LBYTXT   EQU   LDXBUF+16                 TXT - 1ST TEXT BYTE IN CARD  A2519600
LRLDCT   EQU   LDXBUF+10                 RLD - NBR OF BYTES IN CARD   A2519610
LRELHD   EQU   LDXBUF+16                 RLD - RELOCATION HEADER      A2519620
LPOSHD   EQU   LDXBUF+18                 RLD - POSITION HEADER        A2519630
LRLFLG   EQU   LDXBUF+20                 RLD - COMPL/CONTIN. FLAGS    A2519640
LRLADR   EQU   LDXBUF+21                 RLD - ASSEMBLY ADDR OF SYMB  A2519650
LENDAD   EQU   LDXBUF+5                  END - TRANSFER ADDRESS       A2519660
LREPAD   EQU   LDXBUF+6                  REP - HEXADEC.ADDRESS        A2519670
LREPID   EQU   LDXBUF+13                 REP - ES ID OF C.S.          A2519680
LRPBYT   EQU   LDXBUF+16                 REP - IST REP BYTE IN CARD   A2519690
LICSNM   EQU   LDXBUF+16                 ICS - NAME OF C.S.           A2519700
LICSLG   EQU   LDXBUF+24                 ICS - LENGTH                 A2519710
LSLCAD   EQU   LDXBUF+6                  SLC - HEXADEC ADDRESS        A2519720
LSLCNM   EQU   LDXBUF+16                 SLC - SYMBOLIC ADDRESS       A2519730
SAVREG   EQU   TBSIZE                    SAVE ADDRESS REGISTER        A2519740
         SPACE                                                        A2519750
*************************************************************************A2519760
*                                                                     * A2519770
*                        REGISTER ASSIGNMENT                          * A2519780
*                                                                     * A2519790
*************************************************************************A2519800
         SPACE                                                        A2519810
LBASRG   EQU   15                       BASE REGISTER                A2519820
LBAZRG   EQU   14                       BASE REGISTER                A2519830
CRDCNT   EQU   13                       CARD COUNTING REGISTER       A2519840
LDCONT   EQU   7                        MUST BE ODD                  A2519850
LDSTEP   EQU   8                        LDCONT+1                     A2519860
LDFINL   EQU   9                        LDCONT+2                     A2519870
LDBRRG   EQU   12                       LINK REGISTER                A2519880
LMIDRG   EQU   LDBRRG                   *                            A2519890
LWK1RG   EQU   1                        WORKING REG.1-ODD            A2519900
LWK2RG   EQU   2                           -     - 2-WK1RG+1         A2519910
LWK3RG   EQU   3                           -     - 3-  -  +2         A2519920
LWK4RG   EQU   4                           -     - 4-  -  +3         A2519930
LWK5RG   EQU   5                           -     - 5-  -  +4         A2519940
LWK6RG   EQU   6                           -     - 6-  -  +5         A2519950
LEXIT1   EQU   10                       RETURN REGISTER 1            A2519960
LEXIT2   EQU   11                          -     -    2              A2519970
LEXIT3   EQU   LDBRRG                      -     -    3              A2519980
LEXIT4   EQU   LWK3RG                      -     -    4              A2519990
LINKR1   EQU   LEXIT1                   PARAMETER LIST REGISTER 1    A2520000
LINKR2   EQU   LEXIT2                   PARAMETER LIST REGISTER 2    A2520010
         EJECT                                                        A2520020
*************************************************************************A2520030
*                                                                     * A2520040
*                        CONSOLE  MESSAGE                             * A2520050
*                                                                     * A2520060
*  THIS MESSAGE IS PRINTED ON THE IBM 1052 PRINTER-KEYBOARD.  IT  IS  * A2520070
*  ISSUED AT THE END OF THE LOADING PROCESS  IF ANY WARNING OR ERROR   * A2520080
*  CONDITION HAVE ARISEN AND IF THE LOADER HAS NOT PRINTED  THE  AC-  * A2520090
*  CORDING MESSAGES.                                                  * A2520100
*                                                                     * A2520110
*************************************************************************A2520120
```

```
         SPACE                                                          A2520130
MSDG01   DC    FL1'36'                    MESSAGE LENGTH               A2520140
         DC    AL3(*+3)                     -    ADDRESS               A2520150
MSDGCX   DC    CL3' A2'                     -    CODE                  A2520160
         DC    CL3'11W'                     -    SERIAL NUMBER         A2520170
         DC    CL16' LOADING ERROR,C'       -    TEXT                  A2520180
         DC    CL14'ANNOT CONTINUE'                                    A2520190
         DC    X'15'                                                   A2520200
         SPACE 2                                                       A2520210
*****************************************************************     A2520220
*                                                               *     A2520230
*                      LOADING CONTROL MESSAGES                 *     A2520240
*                                                               *     A2520250
*  THESE MESSAGES ARE PRINTED ON AN IBM PRINTER OR WRITTEN ON A TAPE  * A2520260
*  UNIT  ONLY  IF THE USER WISHES  FOR THEM.                    *     A2520270
*                                                               *     A2520280
*****************************************************************     A2520290
         SPACE 2                                                       A2520300
MESHDR   DC    C' RL '                    (=RELOCATING LOADER)         A2520310
         SPACE 2                                                       A2520320
*****************************************************************     A2520330
*                                                               *     A2520340
*                   1. INFORMATIVE  MESSAGES                    *     A2520350
*                                                               *     A2520360
*  THE FOLLOWING MESSAGES ARE USED TO FORM A STORAGE MAP OR INDEX OF  * A2520370
*  THE CURRENT ADDRESSES OF ALL CONTROL SECTION NAMES  AND  TO  MAKE  * A2520380
*  THE INITIAL PSW  KNOWN.                                      *     A2520390
*                                                               *     A2520400
*****************************************************************     A2520410
         SPACE                                                         A2520420
MSDG1K   DC    C'00I'                   MSDG1K SERIAL NUMBER          A2520430
         DC    FL1'26'                    -    LENGTH                A2520440
         DC    CL16'CONTROL SECTION '     -    TEXT                  A2520450
         DC    CL10'LOADED AT '                                       A2520460
MSDGKK   EQU   MESTXT+26                                              A2520470
         DS    0F                                                     A2520480
AMSDGK   DC    AL3(MSDGKK)              HEXADEC PART OF MESSAGE 1K    A2520490
         DC    X'04'                                                  A2520500
MSDG1Z   DC    C'01I'                   MSDG1Z SERIAL NUMBER          A2520510
         DC    FL1'14'                                                A2520520
         DC    CL14'*INITIAL  PSW '                                   A2520530
MSDGZZ   EQU   MESTXT+14                                              A2520540
         DS    0F                                                     A2520550
AMSDGZ   DC    AL3(MSDGZZ)                                            A2520560
         DC    X'0E'                                                  A2520570
         SPACE                                                        A2520580
*****************************************************************     A2520590
*                                                               *     A2520600
*                    2.   WARNING  MESSAGES                     *     A2520610
*                                                               *     A2520620
*****************************************************************     A2520630
         SPACE                                                        A2520640
MSDG11   DC    C'02I'                   MSDG11 SERIAL NUMBER          A2520650
         DC    FL1'28'                                                A2520660
         DC    CL16'ILLEGAL CARD IN '                                 A2520670
```

```
         DC    CL12'LOADER INPUT'                                      A2520680
MSDG12   DC    C'03I'          MSDG12 SERIAL NUMBER                    A2520690
         DC    FL1'27'                                                 A2520700
         DC    CL16'TXT FOLLOWS REP '                                  A2520710
         DC    CL11'OR RLD CARD'                                       A2520720
MSDG13   DC    C'04I'          MSDG13 SERIAL NUMBER                    A2520730
         DC    FL1'43'                                                 A2520740
         DC    CL16'ADDRESS OUTSIDE '                                  A2520750
         DC    CL16'C.S. OR C.S. ALR'                                  A2520760
         DC    CL11'EADY LOADED'                                       A2520770
MSDG14   DC    C'05I'          MSDG14 SERIAL NUMBER                    A2520780
         DC    FL1'36'                                                 A2520790
         DC    CL16'TXT CARD CONTAIN'                                  A2520800
         DC    CL16'S MORE THAN 56 B'                                  A2520810
         DC    CL4'YTES'                                               A2520820
MSDG15   DC    C'06I'          MSDG15 SERIAL NUMBER                    A2520830
         DC    FL1'27'                                                 A2520840
         DC    CL16'TEXT OVERLAYS LO'                                  A2520850
         DC    CL11'ADER TABLES'                                       A2520860
MSDG16   DC    C'07I'          MSDG16 SERIAL NUMBER                    A2520870
         DC    FL1'25'                                                 A2520880
         DC    CL16'ESD CARD FOLLOWS'                                  A2520890
         DC    CL9' TXT CARD'                                          A2520900
MSDG17   DC    C'08I'          MSDG17 SERIAL NUMBER                    A2520910
         DC    FL1'38'                                                 A2520920
         DC    CL16'USED AS ENTRY AN'                                  A2520930
         DC    CL16'D CONTROL SECTIO'                                  A2520940
         DC    CL6'N NAME'                                             A2520950
MSDG18   DC    C'09I'          MSDG18 SERIAL NUMBER                    A2520960
         DC    FL1'38'                                                 A2520970
         DC    CL16'CONTROL SECTION '                                  A2520980
         DC    CL16'DEFINED WITH 2 L'                                  A2520990
         DC    CL6'ENGTHS'                                             A2521000
MSDG19   DC    C'10I'          MSDG19 SERIAL NUMBER                    A2521010
         DC    FL1'33'                                                 A2521020
         DC    CL16'LDT CARD NOT PRE'                                  A2521030
         DC    CL16'CEDED BY END CAR'                                  A2521040
         DC    CL1'D'                                                  A2521050
MSDG1A   DC    C'11I'          MSDG1A SERIAL NUMBER                    A2521060
         DC    FL1'38'                                                 A2521070
         DC    CL16'EXTERNAL SYMBOL '                                  A2521080
         DC    CL16'HAS NO REAL DEFI'                                  A2521090
         DC    CL6'NITION'                                             A2521100
MSDG1B   DC    C'12I'          MSDG1B SERIAL NUMBER                    A2521110
         DC    FL1'34'                                                 A2521120
         DC    CL16'BLANK OR COMMA M'                                  A2521130
         DC    CL16'ISSING IN REP CA'                                  A2521140
         DC    CL2'RD'                                                 A2521150
MSDG1E   DC    C'13I'          MSDG1E SERIAL NUMBER                    A2521160
         DC    FL1'43'                                                 A2521170
         DC    CL16'ADDRESS OF SYMBO'                                  A2521180
         DC    CL16'L IN SLC CARD NO'                                  A2521190
         DC    CL11'T RELOCATED'                                       A2521200
MSDG1D   DC    C'14I'          MSDG1D SERIAL NUMBER                    A2521210
         DC    FL1'36'                                                 A2521220
```

```
         DC    CL16'NEITHER NAME,NOR'                                  A2521230
         DC    CL16' ADDRESS IN SLC '                                  A2521240
         DC    CL4'CARD'                                               A2521250
MSDG1G   DC    C'15I'                  MSDG1G SERIAL NUMBER            A2521260
         DC    FL1'45'                                                 A2521270
         DC    CL16'SLC HAS SET LOC.'                                  A2521280
         DC    CL16'CNTR. TO VALUE A'                                  A2521290
         DC    CL13'LREADY LOADED'                                     A2521300
MSDG1H   DC    C'16I'                  MSDG1H SERIAL NUMBER            A2521310
         DC    FL1'33'                                                 A2521320
         DC    CL16'CHARACTER IN CAR'                                  A2521330
         DC    CL16'D NOT HEXADECIMA'                                  A2521340
         DC    CL1'L'                                                  A2521350
MSDG1J   DC    C'17I'                  MSDG1J SERIAL NUMBER            A2521360
         DC    FL1'23'                                                 A2521370
         DC    CL16'ENTRY POINT IS R'                                  A2521380
         DC    CL7'EPEATED'                                            A2521390
MSDG1L   DC    C'18I'                  MSDG1L SERIAL NUMBER            A2521400
         DC    FL1'27'                                                 A2521410
         DC    CL16'ENTRY POINT NOT '                                  A2521420
         DC    CL11'RELOCATABLE'                                       A2521430
MSDG1M   DC    C'19I'                  MSDG1M SERIAL NUMBER            A2521440
         DC    FL1'23'                                                 A2521450
         DC    CL16'ADDRESS NOT RELO'                                  A2521460
         DC    CL7'CATABLE'                                            A2521470
MSDG1N   DC    C'20I'                  MSDG1N SERIAL NUMBER            A2521480
         DC    FL1'25'                                                 A2521490
         DC    CL16'EOF BEFORE END O'                                  A2521500
         DC    CL9'F LOADING'                                          A2521510
         SPACE                                                         A2521520
*************************************************************************A2521530
*                                                                      * A2521540
*                    3.     ERROR  MESSAGES                            * A2521550
*                                                                      * A2521560
*************************************************************************A2521570
         SPACE                                                         A2521580
MSDG21   DC    C'21W'                  MSDG21 SERIAL NUMBER            A2521590
         DC    FL1'36'                                                 A2521600
         DC    CL16'INSUFFICIENT SPA'                                  A2521610
         DC    CL16'CE FOR LOADER TA'                                  A2521620
         DC    CL4'BLES'                                               A2521630
MSDG23   DC    C'22W'                  MSDG23 SERIAL NUMBER            A2521640
         DC    FL1'42'                                                 A2521650
         DC    CL16'INSUFFICIENT STO'                                  A2521660
         DC    CL16'RAGE AVAILABLE F'                                  A2521670
         DC    CL10'OR PROGRAM'                                        A2521680
MSDG0A   DC    C'23W'                  MSDG0A SERIAL NUMBER            A2521690
         DC    FL1'13'                                                 A2521700
         DC    CL13'PROGRAM ERROR'                                     A2521710
*                                      LOCATION LENTRY=ENTRY POINT     A2521720
         END   LENTRY                  * TO THE RELOCATING LOADER.     A2521730
         AOPIN CROSSREF                                                A2700010
A2UB     TITLE ' UPDATE V-1,L-1 '                                      A2700020
UPDATE   START 0                                                       A2700030
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700040
```

```
* UPDATE                                                                * A2700050
*                                                                       * A2700060
* FUNCTIONS    - MAKE THE CORRECTIONS DEFINED BY THE 'UPDTCORR' CARDS  * A2700070
*                TO THE OLD 'UPDTOLD' TAPE,AND WRITE A NEW 'UPDTNEW '  * A2700080
*                TAPE                                                  * A2700090
*                                                                       * A2700100
* OPTIONS      - LIST OF THE UPDATING OPERATIONS (/ DEVSUP PRINTER)    * A2700110
*                                                                       * A2700120
*              - PARTIAL DUPLICATION OF THE NEW TAPE (DUPLFILE)        * A2700130
*                                                                       * A2700140
* COMMON SIMULATOR  UTILITY PROGRAMS USED BY UPDATING                  * A2700150
*              - ABSOLUTE LOADER                                       * A2700160
*              - CONTROL PROGRAM                                       * A2700170
*              - I/O PACKAGE                                           * A2700180
*              - INITIALIZER                                           * A2700190
*              - RELOCATABLE LOADER                                    * A2700200
*                                                                       * A2700210
* SYSTEM       - 32K MAIN STORAGE (16K IF'UPDATE' IS INITIALIZED)      * A2700220
*              - STANDARD INSTRUCTIONS SET                             * A2700230
*              - 9-TRACK MAGNETIC TAPES                                * A2700240
*              - CARD READER                                           * A2700250
*              - CARD PUNCHER (OPTION FOR UPDTNEW,DUPLFILE)            * A2700260
*              - PRINTER (OPTION FOR UPDATING LIST)                    * A2700270
*              - PRINTER-KEYBOARD                                      * A2700280
*                                                                       * A2700290
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700300
         EJECT                                                            A2700310
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700320
*                                                                       * A2700330
* EXAMPLE OF OLD TAPE TO BE UPDATED (UPDTOLD)                           * A2700340
*                                                                       * A2700350
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700360
*COL1              72 73    80 *                                        * A2700370
*  I                I I      I *                                        * A2700380
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700390
*                              *                      *                 * A2700400
*   ONE RECORD       A73B0001 *                       *                 * A2700410
*   N   RECORDS      A73B ETC * ONE BINARY MODULE     *                 * A2700420
*   ONE RECORD       A73B0119 *    ( B COL 76 )       *                 * A2700430
* * * * * * * * * * * * * * * * * * * * * * * * * * * *                 * A2700440
*   ONE RECORD       A71B0001 *                       * ONE FILE        * A2700450
*   N   RECORDS      A71B ETC * ONE BINARY MODULE     *                 * A2700460
*   ONE RECORD       A71B0097 *                       *                 * A2700470
* * * * * * * * * * * * * * * * * * * * * * * * * * * *                 * A2700480
*                              *                      *                 * A2700490
*    TAPE-MARK                 *                      *                 * A2700500
*                              *                      *                 * A2700510
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700520
*                              *                      *                 * A2700530
*   ONE RECORD       A7300010 *                       *                 * A2700540
*   N   RECORDS      A73  ETC * ONE SYMBOLIC MODULE   *                 * A2700550
*   ONE RECORD       A7318590 *  (NUMERIC COL 76)     *                 * A2700560
* * * * * * * * * * * * * * * * * * * * * * * * * * * *                 * A2700570
*   ONE RECORD       A7100010 *                       * ONE FILE        * A2700580
*   N   RECORDS      A71  ETC * ONE SYMBOLIC MODULE   *                 * A2700590
```

```
*   ONE RECORD        A7158700 *                     *              * A2700600
* * * * * * * * * * * * * * * * * * * * * * * * * * *              * A2700610
*                              *                     *              * A2700620
*    TAPE-MARK                 *                     *              * A2700630
*                              *                     *              * A2700640
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700650
*                              *                     *              * A2700660
*                              *                     *              * A2700670
*   ONE RECORD        A7200000 *                     *              * A2700680
*   N   RECORDS       A72  ETC * ONE SYMBOLIC MODULE  *              * A2700690
*   ONE RECORD        A7201020 *                     * ONE FILE     * A2700700
* * * * * * * * * * * * * * * * * * * * * * * * * * *              * A2700710
*                              *                     *              * A2700720
*   ONE RECORD        A7A99999 * ONE SYMBOLIC MODULE  *              * A2700730
* * * * * * * * * * * * * * * * * * * * * * * * * * *              * A2700740
*                              *                     *              * A2700750
*    TAPE-MARK                 *                     *              * A2700760
*                              *                     *              * A2700770
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700780
*                              *                     *              * A2700790
*    TAPE-MARK                 *                     * DATA END     * A2700800
*                              *                     *              * A2700810
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700820
        EJECT                                                          A2700830
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2700840
*                                                                      * A2700850
* CORRECTION CARDS (UPDTCORR) USED TO UPDATE AN OLD TAPE (UPDTOLD)     * A2700860
*                                                                      * A2700870
*  1-AT FILE LEVEL                                                     * A2700880
*      / UPDATE CARD TO DESIGNATE AN OLD FILE                          * A2700890
*          OR - TO BE COPIED                                           * A2700900
*          OR - BEFORE WHICH A NEW FILE MUST BE INSERTED               * A2700910
*          OR - OF WHICH GIVEN MODULES MUST BE CORRECTED               * A2700920
*         (IF A / UPDATE CARD IS NOT PRESENT FOR AN OLD FILE,THIS      * A2700930
*          FILE WILL BE DELETED)                                       * A2700940
*                                                                      * A2700950
*  2-AT COMPLETE MODULE LEVEL, OR AT RECORDS IN A MODULE LEVEL         * A2700960
*      12-2-9 RIS CARD FOR                                             * A2700970
*              - REPLACE                                               * A2700980
*              - INSERT                                                * A2700990
*              - SUPPRESS                                              * A2701000
*              - NUMBERING                                             * A2701010
*                                                                      * A2701020
*  3-MODIFICATION CARDS,FOLLOWING AN RIS CARD USED FOR                 * A2701030
*              - AN INSERTION                                          * A2701040
*              - OR AN REPLACEMENT                                     * A2701050
*        THESE CARDS REPRESENT THE NEW RECORDS TO BE WRITTEN INTO      * A2701060
*          THE NEW TAPE (UPDTNEW).                                     * A2701070
*                                                                      * A2701080
*        THEY MUST CONFORM TO THE FOLLOWING STANDARDS                  * A2701090
*                                                                      * A2701100
*             A-MODULE IDENTIFICATION COL 73 THROUGH 75 OR 76.         * A2701110
*                -ALL MODIFICATION CARDS FOLLOWING AN RIS CARD MUST    * A2701120
*                 HAVE THE SAME IDENTIFICATION,WHICH IS THAT OF THE    * A2701130
*                 NEW MODULE TO BE WRITTEN                             * A2701140
```

```
*                -A MODIF CARD FOLLOWING A RIS CARD INVOLVING RECORDS* A2701150
*                 (AND NOT A WHOLE MODULE) MUST HAVE THE SAME IDENTI-* A2701160
*                 FICATION AS IN COL 8 THROUGH 10 OR 11 OF THE RIS   * A2701170
*                 CARD                                               * A2701180
*                                                                    * A2701190
*             B-NUMBERING COLUMNS 76 OR 77 THROUGH 80.               * A2701200
*                -ALL MODIFICATION CARDS MUST BE NUMBERED.           * A2701210
*                -NO CHARACTER OTHER THAN THOSE FROM 0 THROUGH 9 ARE * A2701220
*                 ACCEPTED IN THIS FIELD.                            * A2701230
*                -THE NUMBERING OF A MODIFICATION CARD SET FOLLOWING * A2701240
*                 AN RIS CARD MUST BE IN ASCENDING ORDER             * A2701250
*                -THE NUMBER IS A SERIAL NUMBER FOR USER.            * A2701260
*                 THE CORRECTION CARDS WILL BE NUMBERED BY THE       * A2701270
*                 PROGRAM BEFORE WRITING ON THE NEW TAPE.            * A2701280
*                                                                    * A2701290
*             C-NUMBER OF THE MODIFICATION CARDS.                    * A2701300
*                -THIS NUMBER CAN BE DIFFERENT FROM THE NUMBER OF THE* A2701310
*                 OLD RECORDS WHICH HAVE BEEN REPLACED.              * A2701320
*                                                                    * A2701330
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701340
        EJECT                                                         A2701350
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701360
* EXAMPLE OF UPDATING AT FILE LEVEL                                  * A2701370
*   UPDTOLD CONSISTING OF * ONE FILE , MODULES A73B,A71B,TM.         * A2701380
*                         * ONE FILE , MODULES A73 ,A71 ,TM.         * A2701390
*                         * ONE FILE , MODULES A72 ,A7A ,TM.         * A2701400
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701410
*                         *                       *                  * A2701420
*      FUNCTIONS          * / UPDATE CARDS        * OTHER CARDS (RIS) * A2701430
*                         * 1      8 13 15 18 24  *     FOLLOWING     * A2701440
*                         * I      I  I I  I   I  * A / UPDATE  CARD  * A2701450
*                         *                       *                  * A2701460
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701470
*                         *                       *                  * A2701480
* COPY AN OLD FILE        * / UPDATE    A73B      *                  * A2701490
* (A73B,A71B,TM)          *                       *                  * A2701500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701510
*                         *                       *                  * A2701520
* DELETE AN OLD FILE      *                       *                  * A2701530
* (A73,A71,TM)            *                       *                  * A2701540
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701550
*                         *                       *                  * A2701560
* INSERT A NEW FILE       * / UPDATE    A72     I * RIS CARD(INSERT A MOD-* A2701570
* (YYY,XXX,TM)            *                       * ULE AT BEGINNING OF A * A2701580
* BEFORE AN OLD FILE      *                       * FILE)              * A2701590
* (A72,A7A,TM)            *                       *+MODIFICATION CARDS  * A2701600
*                         *                       * (1 SET 'RIS + MODIFI- * A2701610
*                         *                       * CATIONS'FOR EACH MOD- * A2701620
*                         *                       * ULE TO BE INSERTED)  * A2701630
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701640
*                         *                       *                  * A2701650
* CORRECT SOME MODULES    *                       * RIS CARD(THIS CARD   * A2701660
* (A7A FOR EXAMPLE)       *                       * DEFINES THE MODULE TO * A2701670
* IN AN OLD FILE          *                       * BE CORRECTED AND THE  * A2701680
* (A72,A7A,TM)            *                       * TYPE OF CORRECTION TO * A2701690
```

```
*OR                    *                      * BE MADE)              * A2701700
* 1ST OPTION-LIST ALL  * / UPDATE  A A72      *+MODIFICATIONS CARDS   * A2701710
* MODULES OF THE FILE  *                      * (IN CASE OF A INSERT  * A2701720
*                      *                      * OR REPLACE REQUEST)   * A2701730
* * * * * * * * * * * * * * * * OR  * * * * *                         * A2701740
*OR                    *                      *(1 SET 'RIS+MODIFICA-  * A2701750
* 2ND OPTION-LIST COR- * / UPDATE    A72      * TIONS'FOR EACH MODULE * A2701760
* RECTED MODULES ONLY  *                      * TO BE CORRECTED)      * A2701770
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701780
* INSERT A NEW FILE    * / UPDATE             *                       * A2701790
* (VVV,WWW,XXX,TM)     *                      * SEE 'INSERT A NEW FILE* A2701800
* AT END OF THE NEW    *                      * BEFORE AN OLD FILE    * A2701810
* TAPE                 *                      *                       * A2701820
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701830
*                                                                       * A2701840
* NOTE  ANY IDENTIFICATION IN COL 15 THROUGH 17 OR 18 OF THE / UPDATE * A2701850
*       CARD IS THAT OF THE FIRST MODULE OF THE OLD FILE                * A2701860
*                                              *****                    * A2701870
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701880
        EJECT                                                           A2701890
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701900
*                                                                       * A2701910
* EXAMPLE OF UPDATING AT MODULES LEVEL IN ANY GIVEN OLD FILE.           * A2701920
*      (MODULES ZZZB,YYY ,DDD ,EEEB,GGG ,TM.)                          * A2701930
*                                                                       * A2701940
* A / UPDATE CARD HAS ESTABLISHED THE LINK BETWEEN THE CORRECTION       * A2701950
*      CARDS       AND THE OLD FILE .                                   * A2701960
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2701970
*                    * ONE RIS CARD (12 2 9 COL 1)    * MODIFICATION * A2701980
*                    * FOR EACH CORRECTED MODULE      *     CARDS    * A2701990
*      FUNCTIONS     *                                * FOLLOWING A  * A2702000
*                    *1  4 8   41 44 52     59   67 69*   RIS  CARD  * A2702010
*                    *I  I I    I I  I      I     I I *              * A2702020
*                    *                                *              * A2702030
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702040
*                    *                                *              * A2702050
* INSERT A NEW MODULE * RIS      I C         X     X X * CARDS        * A2702060
*      (AAA)         *                                * IDENTIFIED   * A2702070
* AT BEGINNING OF  A *         (...X   SEE  NOTE)     * BY   AAA     * A2702080
* FILE               *                                * COL 73...75  * A2702090
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702100
*                    *                                *              * A2702110
*REPLACE AN OLD MODULE* RIS ZZZB R C         X     X X * CARDS        * A2702120
*      (ZZZB)        *                                * IDENTIFIED   * A2702130
* BY A NEW MODULE    *                                * BY   BBB     * A2702140
*      (BBB)         *                                * COL 73...75  * A2702150
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702160
*                    *                                *              * A2702170
* DELETE AN OLD MODULE* RIS YYY  S C                  *              * A2702180
*      (YYY)         *                                *              * A2702190
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702200
*                    *                                *              * A2702210
* INSERT A NEW MODULE * RIS YYY  I C         X     X X * CARDS        * A2702220
*      (CCCB)        *                                * IDENTIFIED   * A2702230
* AFTER AN OLD MODULE *                               * BY   CCCB    * A2702240
```

```
*      (YYY)          *                                  * COL 73...76  * A2702250
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702260
*                     *                                  *              * A2702270
* COPY  AN OLD MODULE *                                  *              * A2702280
*      (DDD)          *                                  *              * A2702290
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702300
*                     *                                  *              * A2702310
* NUMBER AN OLD FILE  * RIS EEEB N C         X     X X *              * A2702320
*      (EEEB)         *                                  *              * A2702330
* WITHOUT CHANGING    *                                  *              * A2702340
*  ITS IDENTIFICATION *                                  *              * A2702350
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702360
*                     *                                  *              * A2702370
* NUMBER AN OLD FILE  * RIS GGG  N C  FFFB   X     X X *              * A2702380
*      (GGG)          *                                  *              * A2702390
* AND CHANGE ITS      *                                  *              * A2702400
*    IDENTIFICATION TO*                                  *              * A2702410
*      (FFFB)         *                                  *              * A2702420
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702430
        EJECT                                                             A2702440
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702450
*                                                                         * A2702460
* NOTE  DUPLICATION OF A NEW MODULE ON'DUPLFILE'                          * A2702470
*                                                                         * A2702480
*     PUNCH D INTO COL 69 OF THE RIS CARD WHICH CAUSED THE MODULE TO      * A2702490
*     BE CORRECTED.                                                       * A2702500
*                                                                         * A2702510
*     ONLY THE MODULES DESIGNATED BY RIS CARDS (OTHER RIS CARDS THAN      * A2702520
*     SUPPRESSION OF A COMPLETE MODULE REQUESTING) CAN BE DUPLICATED      * A2702530
*                                                                         * A2702540
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702550
* NOTE  NUMBERING OF COL 76 OR 77 THROUGH 80 OF THE NEW RECORDS OF A      * A2702560
*     CORRECTED MODULE.                                                   * A2702570
*                                                                         * A2702580
*     SPECIFICATIONS IN AN RIS CARD 'BY MODULE' (C INTO COL 44)           * A2702590
*          COL ---59 = INITIAL NUMBER REQUESTED FOR THE FIRST NEW         * A2702600
*                      RECORD OF THE CORRECTED MODULE                     * A2702610
*          COL ---67 = NUMBERING STEP                                     * A2702620
*                                                                         * A2702630
*     THE PROGRAM ASSIGNS THE VALUE 10 TO ONE OR THE OTHER OF THESE 2     * A2702640
*     FIELDS LEFT BLANK BY THE USER. A STEP OF ZERO IS ALSO REPLACED      * A2702650
*     BY THE VALUE 10. ANY OTHER NUMERICAL VALUE WILL BE ACCEPTED.        * A2702660
*                                                                         * A2702670
*  ***************************************************************      * A2702680
*  * EXAMPLES(MODULE FFFB)          *    PUNCH IN RIS CARD      *      * A2702690
*  *         TO OBTAIN              * INITIAL NUMBER      STEP  *      * A2702700
*  *                                *    COL---59      COL---67 *      * A2702710
*  ***************************************************************      * A2702720
*  * FFFB0010,FFFB0020,FFFB0030 ETC*    (BLANKS)      (BLANKS) *      * A2702730
*  *                               *OR        10            10 *      * A2702740
*  * FFFB0000,FFFB0010,FFFB0020 ETC*          0      (BLANKS) *      * A2702750
*  * FFFB0001,FFFB0002,FFFB0003 ETC*          1             1 *      * A2702760
*  * FFFB0002,FFFB0005,FFFB0010 ETC*          2             5 *      * A2702770
*  * FFFB0100,FFFB0102,FFFB0104 ETC*        100             2 *      * A2702780
*  ***************************************************************      * A2702790
```

```
*                                                                     * A2702800
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702810
        EJECT                                                         A2702820
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702830
* EXAMPLE OF UPDATING AT THE RECORD LEVEL IN AN OLD MODULE           * A2702840
*      (MODULE NNN00010,NNN00020 THROUGH NNN00070,NNN00080)           * A2702850
*                                                                     * A2702860
* A / UPDATE CARD HAS BEEN ESTABLISHED TO DESIGNATE THE OLD FILE OF   * A2702870
*      WHICH THE MODULE TO BE CORRECTED IS A PART( OTHER MODULES IN   * A2702880
*      THE SAME FILE MAY HAVE BEEN CORRECTED)                         * A2702890
*                                                                     * A2702900
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702910
*                 *                                     *             * A2702920
*                 *ONE RIS CARD(12 2 9 COL 1)FOR EACH   *MODIFICATION* A2702930
*                 *    CORRECTION   REQUESTED           *    CARDS    * A2702940
*    FUNCTIONS    *1  4 8       24       41   59    67 69* FOLLOWING  * A2702950
*                 *I  I I        I        I    I     I I*AN RIS CARD * A2702960
*                 *                                     *             * A2702970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2702980
*                 *                                     *             * A2702990
* INSERT 2 RECORDS *                                     * 2 CARDS     * A2703000
*(NNN00001 THR.  2)* RIS NNN              I    X     X X* IDENTIFIED  * A2703010
* AT BEGINNING OF  *                                    *  BY NNN     * A2703020
* A MODULE (NNN)   *            (..X  SEE NOTE)         * COL 73--75  * A2703030
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703040
*                 *                                     *             * A2703050
* COPY OLD RECORDS *                                     *             * A2703060
*(NNN00010 THR. 30)*                                    *             * A2703070
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703080
*                 *                                     *             * A2703090
* REPLACE THE     *                                     * 4 CARDS     * A2703100
* OLD RECORDS     * RIS NNN00040 NNN00050 R   40    10  * IDENTIFIED  * A2703110
*(NNN00040 THR. 50)*                                    *  BY NNN     * A2703120
*BY THE NEW RECORDS*                                    *COL 73--75   * A2703130
*(NNN00040 THR. 70)*                                    *             * A2703140
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703150
*                 *                                     *             * A2703160
* NUMBER THE      *                                     *             * A2703170
* OLD RECORDS     * RIS NNN00060 NNN00070 N   80    10  *             * A2703180
*(NNN00060 THR. 70)*                                    *             * A2703190
*FROM   80 TO   90 *                                    *             * A2703200
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703210
*                 *                                     *             * A2703220
* DELETE THE OLD  *                                     *             * A2703230
* RECORD (NNN00080)* RIS NNN00080 NNN00080 S            *             * A2703240
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703250
*                 *                                     *             * A2703260
* INSERT THE NEW  *                                     * 1 CARD      * A2703270
* RECORD (NNN00100)* RIS NNN00080          I  100     X  * IDENTIFIED  * A2703280
*AFTER   OLD RECORD*                                    *   BY NNN    * A2703290
* (NNN00080)      *                                     *COL 73--75   * A2703300
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703310
        EJECT                                                         A2703320
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703330
*                                                                     * A2703340
```

```
* NOTE  DUPLICATION OF A NEW MODULE ON 'DUPLFILE'                        * A2703350
*                                                                       * A2703360
*     PUNCH D INTO COL 69 OF THE FIRST RIS CARD WHICH CONCERNS THE      * A2703370
*     MODULE TO BE DUPLICATED.                                          * A2703380
*                                                                       * A2703390
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703400
*                                                                       * A2703410
* NOTE  NUMBERING COL 76 OR 77 THROUGH 80 OF THE NEW RECORDS WRITTEN    * A2703420
*     DURING CORRECTION .  THE OLD RECORDS COPIED ,WHICH ARE NOT        * A2703430
*     DESIGNED BY AN RIS CARD, ARE NOT AFFECTED BY THIS NOTE.           * A2703440
*                                                                       * A2703450
*     SPECIFICATIONS IN AN RIS CARD 'BY RECORD' (NOT C IN COL 44)       * A2703460
*          COL ---59 = INITIAL NUMBER REQUESTED FOR THE FIRST NEW       * A2703470
*                      RECORD WRITTEN DURING THE CORRECTION             * A2703480
*          COL ---67 = NUMBERING STEP                                   * A2703490
*                                                                       * A2703500
*     THE PROGRAM ASSIGNS THE VALUE 1 TO ONE OR THE OTHER OF THESE 2    * A2703510
*     FIELDS LEFT BLANK BY THE USER. A STEP OF ZERO IS ALSO REPLACED    * A2703520
*     BY THE VALUE  1 .ANY OTHER NUMERICAL VALUE WILL BE ACCEPTED,IF    * A2703530
*     IT ALLOWS THE INCREASING NUMBERING OF THE NEW RECORDS OF THE      * A2703540
*     SAME MODULE.                                                      * A2703550
*                                                                       * A2703560
*  ************************************************************         * A2703570
*  * EXAMPLES (MODULE NNN)          *      PUNCH IN   RIS CARD  *       * A2703580
*  *          TO OBTAIN             * INITIAL NUMBER     STEP  *       * A2703590
*  *                                *    COL---59      COL---67 *       * A2703600
*  ************************************************************         * A2703610
*  * NNN00001,NNN00002,NNN00003 ETC*    (BLANKS)      (BLANKS) *       * A2703620
*  *                               *OR         1             1 *       * A2703630
*  * NNN00100,NNN00101,NNN00102 ETC*         100      (BLANKS) *       * A2703640
*  * NNN00001,NNN00010,NNN00020 ETC*    (BLANKS)           10 *       * A2703650
*  ************************************************************         * A2703660
*                                                                       * A2703670
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703680
        EJECT                                                             A2703690
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703700
*                                                                       * A2703710
* LISTING SUPPLIED TO PRINTER                                           * A2703720
*                                                                       * A2703730
*   1-IT CONTAINS                                                       * A2703740
*                                                                       * A2703750
*      - THE LIST OF THE NEW RECORDS (WITH THEIR IDENTIFICATION ZONES,  * A2703760
*        AND ON THE SAME LINE THE IDENTIFICATION ZONE OF THE CORRES-    * A2703770
*        PONDING MODIF CARDS AND OLD RECORDS).                          * A2703780
*                                                                       * A2703790
*      - THE IMAGES OF / UPDATE AND RIS CARDS.                          * A2703800
*                                                                       * A2703810
*      - ANY ERROR OR INFORMATIVE MESSAGES.                             * A2703820
*                                                                       * A2703830
*   2-THE MODULES BELONGING TO COPIED OR DELETED FILES ARE NOT LISTED.  * A2703840
*                                                                       * A2703850
*   3-WHEN A PART OF A FILE WILL BE  CORRECTED,THE USER,                * A2703860
*      - CAN REQUEST THE LIST OF ALL THE MODULES OF THIS FILE,          * A2703870
*      - OR ONLY THE LIST OF THE CORRECTED OR INSERTED MODULES.         * A2703880
*                                                                       * A2703890
```

```
*   4-THE LIST CAN BE DELETED IF THE 'DEVSUP PRINTER'CARD IS NOT      * A2703900
*       PRESENT                                                       * A2703910
*                                                                     * A2703920
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703930
        EJECT                                                          A2703940
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2703950
*                                                                     * A2703960
* CALLING PROCEDURE OF UPDATING PROGRAM                               * A2703970
*                                                                     * A2703980
* PHASE-1-PREPARATION OF PARAMETER CARDS                              * A2703990
*                                                                     * A2704000
* / DEV360 ADDR=X'009'1052'          PRINTER-KEYBOARD                 * A2704010
* / DEV360 ADDR=X'00C'2540',COLBIN   CARD READER                      * A2704020
* / DEV360 ADDR=X'00D'2540'          CARD PUNCHER                     * A2704030
* / DEV360 ADDR=X'00E'1403',132BAR   PRINTER                          * A2704040
* / DEV360 ADDR=X'ADD'2400',9TRACK   FOR EACH TAPE UNIT USED          * A2704050
*                                                                     * A2704060
* / DEVSUP  SIMXSYS=X'ADD',TYPE,I    UPDATE PROGRAM                   * A2704070
* / DEVSUP  SIMXOUT=X'ADD',TYPE,O    LOADER MESSAGES                  * A2704080
*                                                                     * A2704090
* / DEVSUP  UPDTOLD=X'ADD',2400,I    OLD TAPE TO BE UPDATED           * A2704100
* / DEVSUP UPDTCORR=X'ADD',TYPE,I    CORRECTIONS                      * A2704110
* / DEVSUP  UPDTNEW=X'ADD',TYPE,O    NEW TAPE AFTER UPDATING          * A2704120
* / DEVSUP  PRINTER=X'ADD',TYPE,O    LIST OF UPDATED RESULTS          * A2704130
* / DEVSUP DUPLFILE=X'ADD',TYPE,O    PARTIAL DUPLICATION OF UPDTNEW   * A2704140
*                                                                     * A2704150
* / CALL UPDATE,LIST,PUNCH           CALLING OF UPDATE PROGRAM        * A2704160
*                                                                     * A2704170
*                                                                     * A2704180
* PHASE-2-READYING OF I/O UNITES                                      * A2704190
*                                                  ** ONE TAPE CAN    * A2704200
*   UTILITY PROGRAMS  TAPE(FILE PROTECT)OR CARD READER * CONTAIN ALL  * A2704210
*   UPDATE PROGRAM    TAPE(FILE PROTECT)OR CARD READER * OR PART OF   * A2704220
*   UPDTOLD           TAPE(FILE PROTECT)                * THESE 3 ITEMS* A2704230
*                                                      **             * A2704240
*   UPDTNEW           TAPE UNIT OR CARD PUNCH                          * A2704250
*   DUPLFILE          TAPE UNIT OR CARD PUNCH   OR NOT USED            * A2704260
*   PRINTER           TAPE UNIT OR PRINTER OR NOT USED                 * A2704270
*                                                                     * A2704280
*   PARAMETER CARDS   TAPE UNIT OR CARD READER                        * A2704290
*                                                                     * A2704300
*                                                                     * A2704310
*     NOTE  DEPENDING ON THE OPTION ,THE CARD READER CAN CONTAIN ALL  * A2704320
*           OR PART OF THE FOLLOWING DECKS                            * A2704330
*                                                                     * A2704340
*               - UTILITY PROGRAMS (FROM IPL THROUGH LDT)             * A2704350
*               - PARAMETER CARDS  (DEV360 THROUGH CALL)              * A2704360
*               - UPDATE PROGRAM ,I.E.                                * A2704370
*                       . / PROGNAME UPDATE,TBSIZ=F'500'              * A2704380
*                       . UPDATE PROGRAM PROPER                       * A2704390
*                       . LDT CARD                                    * A2704400
*               - UPDTCORR (/ UPDATE CARD...ETC)                      * A2704410
*                                                                     * A2704420
*           IN THAT ORDER                                             * A2704430
*                                                                     * A2704440
```

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2704450
        EJECT                                                           A2704460
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2704470
*                                                                       * A2704480
* PHASE-3-LOADING OF UTILITY PROGRAMS                                   * A2704490
*                                                                       * A2704500
*   - SET THE LOAD-UNIT SWITCHES TO THE ADDRESS OF THE UNIT USED FOR    * A2704510
*     THE UTILITY PROGRAMS                                              * A2704520
*                                                                       * A2704530
*   - PRESS THE LOAD KEY                                                * A2704540
*                                                                       * A2704550
*   - THE WAIT STATE WILL BE ENTERED                                    * A2704560
*                                                                       * A2704570
*                                                                       * A2704580
* PHASE-4-CALLING OF UPDATE PROGRAM                                     * A2704590
*                                                                       * A2704600
*   - DEPRESS THE REQUEST KEY ON THE PRINTER-KEYBOARD                   * A2704610
*                                                                       * A2704620
*   - THE FOLLOWING MESSAGE WILL BE PRINTED                             * A2704630
*       AX00A ASSIGN CONTROL CARD INPUT DEVICE                          * A2704640
*                                                                       * A2704650
*   - DEPRESS THE REQUEST KEY ON THE PRINTER-KEYBOARD                   * A2704660
*                                                                       * A2704670
*   - ENTER THE FOLLOWING COMMAND   TYPE,X'ADD' (EXAMPLE 2540,X'00C')   * A2704680
*       GIVING THE ADDRESS OF THE DEVICE FROM WHICH PARAMETER CARDS     * A2704690
*       WILL BE READ                                                    * A2704700
*                                                                       * A2704710
*   - DEPRESS SIMULTANEOUSLY THE 'ALTERNATE CODING' AND NUMERIC'5'KEYS* A2704720
*       ON THE PRINTER-KEYBOARD                                         * A2704730
*                                                                       * A2704740
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2704750
        EJECT                                                           A2704760
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2704770
*                                                                       * A2704780
* PROGRAM INITIALIZATION ROUTINE                                        * A2704790
*                                                                       * A2704800
* ENTRY AT'INIT'(START OF JOB).                                         * A2704810
*                                                                       * A2704820
* FUNCTIONS   - PROGRAM CHECK ROUTINE INITIALIZATION (PCHECK) .         * A2704830
*             - CANCEL THE SVC18 IN ROUTINE 'OPRINT'                    * A2704840
*                  IF THE DEVSUP-PRINTER CARD IS  ABSENT.               * A2704850
*             - CANCEL THE SVC18 IN ROUTINE 'ODUPL'                     * A2704860
*                  IF THE DEVSUP-DUPLFILE CARD IS  ABSENT.              * A2704870
*             - INITIALIZE  THE SWITCHES RELATIVE TO ANY GENER.FUNCT.   * A2704880
*             - READ 1ST CARD FROM UPDTCORR.                            * A2704890
*             - READ 1ST OLD RECORD FROM UPDTOLD.                       * A2704900
*                                                                       * A2704910
* BRANCH TO   -ERSTPD(ERROR-STOP)IF THE FIRST CARD IS NOT A / UPDATE    * A2704920
*                  CARD,OR IF THE FIRST OLD RECORD IS A TAPE MARK.      * A2704930
*             -FLTGA (FILE TREATMENT) IN ALL OTHER CASES .              * A2704940
*                                                                       * A2704950
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2704960
INIT     BALR  BASERG,0                  BASES  LOADING                 A2704970
         USING *,BASERG                  *                              A2704980
         USING *+4000,BASER2             *                              A2704990
```

```
INIT1    LA    BASER2,4000(0,BASERG)        *                              A2705000
         SVC   9                            ENABLE                         A2705010
         CNOP  2,8                                                         A2705020
         SVC   6                            PROGRAM-CHECK ROUTINE          A2705030
         DC    A(PCHECK)                    * INITIALIZATION               A2705040
PRPSW    DS    D                            *                              A2705050
         MVC   DVNAME(8),OPRIN2+2           PRINTER DEVICE IS DEFINED      A2705060
         BAL   LINKF,DEVTST                 *BY A DEVSUP CARD              A2705070
         BC    15,INITD                     NO                             A2705080
         MVC   OPRIN2(2),INITJ              YES-RESTORE SVC 18 INSTRUCT    A2705090
         BC    15,INITE                     *                              A2705100
INITD    MVC   OPRIN2(2),INITH              REPL.SVC18 BY BCR 15,LINKF     A2705110
INITE    MVC   DVNAME(8),ODUPL2+2           DUPLFILE IS DEFINED BY         A2705120
         BAL   LINKF,DEVTST                 *A DEVSUP CARD                 A2705130
         BC    15,INITG                     NO                             A2705140
         MVC   ODUPL2(2),INITJ              YES-RESTORE SVC18 INSTRUCT.    A2705150
         BC    15,INITF                     *                              A2705160
INITG    MVC   ODUPL2(2),INITH              REPL.SVC18 BY BCR 15,LINKF     A2705170
INITF    NI    SWD,X'00'                    NOT ANY DATA END               A2705180
         BAL   LINKC,RDCRD                  READ 1ST CARD                  A2705190
         BC    15,INITA                     DATA END        ERROR          A2705200
         BC    15,INITB                     FILE END (/ UPDATE)            A2705210
         BC    15,INITA                     CS   END  (RIS) ERROR          A2705220
         BC    15,INITA                     NOT CS END(RIS) ERROR          A2705230
INITA    LA    WORK,ERME02                  (MODIF CARD)    ERROR          A2705240
         BAL   LINKD,PRME1                  PRINT ERROR MESSAGE            A2705250
         BC    15,ERSTPD                    * AND STOP                     A2705260
INITB    NI    DPLSW,X'00'                  SWITCHES INITIALIZATION        A2705270
         NI    SWDUMP,X'00'                 *                              A2705280
         NI    SWPRER,X'00'                 *                              A2705290
         NI    SWERR,X'00'                  *                              A2705300
* ITB1   BAL   LINKC,RDOLD          READ 1ST OLD REC  (DELETED) V1L1 A2705310
INITB1   BAL   LINKC,RDOLD          READ 1ST OLD RECORD         V1L1 A2705311
         BC    15,INITC                     DATA END(7F)2   ERROR          A2705320
*        BC    15,INITC             FILE END (7F) 1-  (DELETED) V1L1 A2705330
         BC    15,INITB1            FILE END (7F) 1-READ AGAIN  V1L1 A2705331
         BC    15,FLTGA                     CS END-                        A2705340
         BC    15,FLTGA                     NOT CS END                     A2705350
INITC    LA    WORK,ERME05                  ERROR                          A2705360
         BAL   LINKD,PRME1                  PRINT ERROR MESSAGE            A2705370
         BC    15,ERSTPD                    * AND STOP                     A2705380
INITH    BCR   15,LINKF                                                    A2705390
INITJ    SVC   18                                                          A2705400
         EJECT                                                             A2705410
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2705420
*                                                                         * A2705430
* FILE TREATMENT ROUTINE                                                  * A2705440
*                                                                         * A2705450
* ENTRY AT 'FLTGA'                                                        * A2705460
*          - THE FIRST TIME,FROM ROUTINE 'INIT'                           * A2705470
*          -THEN FROM ROUTINE 'CSTGC' (END OF FILE TREATMENT) .           * A2705480
*                                                                         * A2705490
* DATA       -A / UPDATE CARD HAS BEEN READ (OR CARDS EXHAUSTED) .        * A2705500
*            -THE 1ST RECORD OF ANY OLD FILE HAS BEEN READ (OR THE        * A2705510
*                 LAST TAPE MARK OF UPDTOLD).                             * A2705520
```

```
*                                                                     * A2705530
* FUNCTIONS   -INITIALIZE THE  SWITCHES RELATIVE TO A FILE.           * A2705540
*             -SELECT THE ROUTINE TO BE USED (AND CALL IT BY BAL LINK)* A2705550
*              1-NLSTOP(NORMAL END OF JOB)IF THE INPUT DATA           * A2705560
*                   ARE COMPLETLY PROCESSED (UPDTOLD AND UPDTCORR) .  * A2705570
*              2-SKLDM (DELETE 1 OLD FILE)IF THE PRESENT OLD FILE IS  * A2705580
*                   NOT MENTIONED IN THE READ / UPDATE CARD .         * A2705590
*              3-SKLDN (COPY 1 OLD FILE)IF THIS FILE IS MENTIONED IN  * A2705600
*               THE READ / UPDATE CARD,AND IF THIS CARD IS NOT FOLLOWED A2705610
*                   BY ANY RIS CARD.                                  * A2705620
*              4-SKCRDM(SKIP UPDTCORR UP TO THE NEXT / UPDATE CARD)IF * A2705630
*               THE 1ST CARD AFTER / UPDATE CARD IS BY ERROR          * A2705640
*                   A MODIFICATION CARD .                             * A2705650
*              5-CSTGB (CS.TREATMENT)IF THE OLD FILE IS MENTIONED IN  * A2705660
*                   THE / UPDATE CARD,AND THIS CARD IS FOLLOWED BY    * A2705670
*                   A RIS CARD .                                      * A2705680
*                                                                     * A2705690
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2705700
FLTGA     XC    NBRLIN(4),NBRLIN          CLEAR PRINTED LINES COUNT   A2705710
          XC    LIN1ST(1),LIN1ST          1ST LINE OF CS NOT PRINTED  A2705720
          MVI   NWZONE,X'40'              CLEAR NWZONE                A2705730
          MVC   NWZONE+1(79),NWZONE       *                           A2705740
          MVI   PRZONE,X'40'              CLEAR PRZONE                A2705750
          MVC   PRZONE+1(119),PRZONE      *                           A2705760
          MVI   ERTBG,X'40'               CLEAR ERTBG                 A2705770
          MVC   ERTBG+1(17),ERTBG         *                           A2705780
          NI    SWERR,X'FE'               NO ERROR DETECTED IN FILE   A2705790
          MVC   SWF(1),SWD                IS ALSO FILE END IF IT IS   A2705800
          NI    SWF,X'FB'                 *DATA END                   A2705810
          TM    SWD,X'03'                 ARE BOTH INPUT DATA END     A2705820
          BC    1,NLSTOP                  YES-BRANCH                  A2705830
          TM    SWDCE,X'02'               CARDS DATA END              A2705840
          BC    1,FLTGB                   YES-BRANCH                  A2705850
          TM    SWDOE,X'01'               OLD DATA END                A2705860
          BC    1,FLTGD                   YES-BRANCH                  A2705870
          CLC   GLCSD(CS),OLCSD           SAME CS IN READ / UPDATE AS A2705880
          BC    8,FLTGE                   *IN 1ST OLD RECORD OF FILE  A2705890
FLTGB     MVC   MESS30+51(CS),OLCSD       CS OF READ OLD RECORD       A2705900
          LA    WORK,MESS30               *STORED TO MESSAGE          A2705910
          BAL   LINKD,PRME3               PRINT MESSAGE               A2705920
          BAL   LINKB,SKLDM               DELETE AN OLD FILE          A2705930
          BC    15,FLTGA                  TO NEXT FILE                A2705940
FLTGD     LA    WORK,MESS32               PRINT MESSAGE               A2705950
          BAL   LINKD,PRME3               *                           A2705960
FLTGE     BAL   LINKC,RDCRD               READ 1 CARD                 A2705970
          BC    15,FLTGH                  DATA END                    A2705980
          BC    15,FLTGH                  FILE END(/ UPDATE)          A2705990
          BC    15,FLTGF                  CS END (RIS)                A2706000
          BC    15,FLTGF                  NOT CS END(RIS)             A2706010
          MVC   PRZONE+8(40),GPZONE       MODIF CARD - PRINT THE      A2706020
          BAL   LINKE,PRLINA              *PRECEDING / UPDATE CARD    A2706030
          LA    WORK,ERME04               PRINT ERROR MESSAGE         A2706040
          BAL   LINKD,PRME1               *                           A2706050
          BAL   LINKE,PRLINA              SKIP 1 LINE                 A2706060
          BAL   LINKB,SKCRDM              SKIP MODIFS UP TO / UPDATE  A2706070
```

```
FLTGG    TM    SWDOE,X'01'              OLD DATA END                A2706080
         BC    1,FLTGA                  YES-TO NEXT FILE            A2706090
         LA    WORK,MESS33              PRINT MESSAGE               A2706100
         BAL   LINKD,PRME3              *                           A2706110
         BAL   LINKB,SKLDN              COPY AN OLD FILE ON UPDTNEW A2706120
         BC    15,FLTGA                 TO NEXT FILE                A2706130
FLTGH    MVC   PRZONE+8(40),GPZONE      PRINT THE PRECEDING /UPDATE A2706140
         BAL   LINKE,PRLINA             *CARD                       A2706150
         BC    15,FLTGG                                             A2706160
FLTGF    BAL   LINKE,PREDF              PRECEDING /UPDATE TO MESS29 A2706170
         MVC   PRZONE+8(40),GPZONE      PRINT THE PRECEDING /UPDATE A2706180
         BAL   LINKE,PRLINA             *                           A2706190
         LA    WORK,MESS29              PRINT MESSAGE               A2706200
         BAL   LINKD,PRME3              *                           A2706210
         CLI   GPFCT,X'C9'              IS FILE INSERT REQUEST      A2706220
         BC    7,FLTGI                   NO                         A2706230
         LA    WORK,MESS22              PRINT MESSAGE               A2706240
         BAL   LINKD,PRME3              *                           A2706250
FLTGI    BAL   LINKE,PRLINA             SKIP 1 LINE                 A2706260
         BC    15,CSTGB                 TO CS TREATMENT             A2706270
         EJECT                                                      A2706280
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2706290
*                                                                   * A2706300
* CS TREATMENT ROUTINE  (CS=CONTROL SECTION = MODULE )              * A2706310
*                                                                   * A2706320
* ENTRY AT CSTGB                                                    * A2706330
*     -FROM ROUTINE 'FLTGA'(FILE TREATMENT)                         * A2706340
*                    -THE FOLLOWING CARDS AND RECORDS HAVE BEEN READ#* A2706350
*                    -1ST RIS CARD AFTER / UPDATE CARD              * A2706360
*                    -1ST OLD RECORD OF 1ST CS OF FILE,OR THE LAST  * A2706370
*                         TAPE MARK OF UPDTOLD (OLD DATA END) .     * A2706380
* ENTRY AT CSTGA                                                    * A2706390
*       -FROM THIS ROUTINE,AFTER TREATMENT OF ANY CS IN THE SAME FILE * A2706400
*                    -THE FOLLOWING CARDS AND RECORDS HAVE BEEN READ#* A2706410
*                    -ANY RIS CARD,OR / UPDATE CARD(CARDS FILE END)  * A2706420
*                         UNLESS THERE ARE NO MORE CARD (DATA END)   * A2706430
*                    -1ST OLD RECORD OF ANY CS IN FILE,OR THE TAPE   * A2706440
*                         MARK FOLLOWING THE LAST CS OF A FILE       * A2706450
*                         (OLD FILE END).                           * A2706460
*                                                                   * A2706470
* FUNCTIONS                                                         * A2706480
*       -INITIALIZE THE SWITCHES RELATIVE TO A CS .                 * A2706490
*       -SET SWITCH 'OLD FILE END' ON,IF THE PRECEDING / UPDATE CARD  * A2706500
*          HAS REQUESTED THE 'FILE INSERT FUNCTION'.                * A2706510
*       -SELECT THE ROUTINE TO BE USED (AND CALL IT BY'BAL LINK')     * A2706520
*             1-SKCRDA(SKIP UPDTCORR UP TO NEXT RIS CARD (NEXT CS))  * A2706530
*                   IF THE REQUESTED FUNCTION CANNOT BE PERFORMED .  * A2706540
*             2-SKLDA (COPY 1 OLD CS),IF THE READ OLD RECORD IS NOT  * A2706550
*                   CONCERNED BY THE READ RIS CARD.                  * A2706560
*             3-REPLA(REPLACE 1 OLD CS) * DEPENDING ON THE FUNCTION  * A2706570
*             4-SUPPR(DELETE  1 OLD CS) * REQUESTED BY THE READ RIS  * A2706580
*             5-COUNT(NUMBER  1 OLD CS) * CARD,FOR THE PRESENT CS.   * A2706590
*             6-INSEA(INSERT  1 NEW CS) * (RIS MODE PER CS ) .       * A2706600
*                                                                   * A2706610
* EXIT   TO                                                         * A2706620
```

```
*       -CSTGA (NEXT CS TREATMENT)IN ALL  ABOVE CASES .                 * A2706630
*       -RCTGB (RECORDS TREATMENT) IF THE READ RIS CARD REQUESTS A      * A2706640
*              FUNCTION PER RECORD IN THE CS WHOSE THE 1ST RECORD       * A2706650
*              HAS BEEN READ  .                                         * A2706660
*       -CSTGC (END OF FILE TREATMENT)IF ALL CS HAVE BEEN TREATED IN    * A2706670
*              THE CURRENT FILE.                                        * A2706680
*                                                                       * A2706690
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2706700
CSTGA    NI    LIN1ST,X'00'          1ST LINE OF CS NOT PRINTED        A2706710
CSTGB    NI    SWCS,X'00'            INITIALIZE SWCS                   A2706720
         XC    NWBIN(4),NWBIN        ZERO TO NWBIN                     A2706730
         CLI   GPFCT,X'C9'           IS FILE INSERT REQUEST            A2706740
         BC    7,CSTGS               NO                                A2706750
         OI    SWFOE,X'01'           YES-OLD FILE END                  A2706760
CSTGS    TM    SWF,X'03'             INPUT FILES END                   A2706770
         BC    1,CSTGC               YES-                              A2706780
         TM    SWFCE,X'02'           CARD FILE END                     A2706790
         BC    1,CSTGQ               YES-BRANCH                        A2706800
         TM    RLVALD,X'01'          IS INVALID READ RIS               A2706810
         BC    1,CSTGD               YES-BRANCH (ERROR)                A2706820
         CLI   RLMOD,X'C3'           CORRECTION MODE 'PER CS'          A2706830
         BC    7,CSTGE               NO-BRANCH                         A2706840
         CLI   RLCNT,X'C9'           READ RIS-INSERT REQUEST           A2706850
         BC    7,CSTGR               NO                                A2706860
         CLC   RLIDT1(8),BLANK       INSERT AT FILE BEGINNING          A2706870
         BC    8,CSTGG               YES                               A2706880
         BC    15,CSTGI              INSERT IN FILE                    A2706890
CSTGR    CLC   RLCSD1(CS),OLCSD      SAME CS IN READ RIS AS IN         A2706900
         BC    7,CSTGH               *READ OLD RECORD-NO BRANCH        A2706910
         CLI   RLCNT,X'D9'           READ RIS-REPLACE REQUEST          A2706920
         BC    8,CSTGO               YES                               A2706930
         CLI   RLCNT,X'D5'           READ RIS-NUMBERING REQUEST        A2706940
         BC    8,CSTGO1              YES                               A2706950
         BAL   LINKA,SUPPR           SUPPRESS 1 OLD CS                 A2706960
         BC    15,CSTGA              TO NEXT CS                        A2706970
CSTGO    BAL   LINKA,REPLA           REPLACE 1 OLD CS                  A2706980
         BC    15,CSTGA              TO NEXT CS                        A2706990
CSTGO1   BAL   LINKA,COUNT           NUMBER  1 OLD CS                  A2707000
         BC    15,CSTGA              TO NEXT CS                        A2707010
CSTGD    LA    WORK,ERME06           PRINT ERROR MESSAGE               A2707020
CSTGD1   BAL   LINKD,PRME1           *AND                              A2707030
         BC    15,CSTGP              *CONTINUE                         A2707040
CSTGE    TM    SWFOE,X'01'           IS OLD FILE END                   A2707050
         BC    1,CSTGF               YES-BRANCH                        A2707060
         CLC   RLCSD1(CS),OLCSD      EXPECTED OLD RECORD FOUND         A2707070
         BC    7,CSTGQ               NO                                A2707080
         BC    15,RCTGB              CORRECTION 'PER RECORD'           A2707090
CSTGF    LA    WORK,ERME07           PRINT ERROR MESSAGE               A2707100
         BC    15,CSTGD1             *AND CONTINUE                     A2707110
CSTGG    TM    SWFO1,X'04'           1ST RECORD OF A FILE              A2707120
         BC    1,CSTGH               NO-BRANCH (ERROR)                 A2707130
CSTGG1   BAL   LINKA,INSEA           INSERT 1ST CS OF UPDTNEW          A2707140
         BC    15,CSTGA              TO NEXT CS                        A2707150
CSTGH    LA    WORK,ERME08           PRINT ERROR MESSAGE               A2707160
         BC    15,CSTGD1             *AND CONTINUE                     A2707170
```

```
CSTGI    CLC   RLCSD1(CS),OPCSD          EXPECTED CSID IN PRECEDING  A2707180
         BC    7,CSTGJ                   *OLD RECORD - NO            A2707190
         BC    15,CSTGG1                 INSERT CS                   A2707200
CSTGJ    TM    SWFOE,X'01'               IS OLD FILE END             A2707210
         BC    1,CSTGK                   YES-BRANCH (ERROR)          A2707220
         BC    15,CSTGQ                  NO-                         A2707230
CSTGK    LA    WORK,ERME07               PRINT ERROR MESSAGE         A2707240
         BC    15,CSTGD1                 *AND CONTINUE               A2707250
CSTGM    TM    SWFOE,X'01'               IS OLD FILE END             A2707260
         BC    1,CSTGN                   YES-BRANCH (ERROR)          A2707270
CSTGQ    BAL   LINKB,SKLDA               COPY AN OLD CS              A2707280
         BC    15,CSTGA                  TO NEXT CS                  A2707290
CSTGN    LA    WORK,ERME07               PRINT ERROR MESSAGE         A2707300
         BC    15,CSTGD1                 *AND CONTINUE               A2707310
CSTGP    BAL   LINKB,SKCRDA              SKIP MODIF CARDS UP TO      A2707320
         BC    15,CSTGA                  *NEXT CS                    A2707330
         EJECT                                                       A2707340
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2707350
*                                                                    * A2707360
* END OF FILE TREATMENT                                              * A2707370
*                                                                    * A2707380
* 1ST ENTRY AT CSTGC,FROM THE 'CS TREATMENT' ROUTINE .               * A2707390
*           -PRINT A MESSAGE IF ANY ERROR HAS OCCURED DURING THE     * A2707400
*              TREATMENT OF THE PRECEDING FILE .                     * A2707410
*           -PRINT A MESSAGE FOR EACH ERROR CODE (I,O,N OR D)PRINTED * A2707420
*              IN THE PRECEDING LIST.                                * A2707430
*           -BRANCH TO CSTGC5 .                                      * A2707440
*                                                                    * A2707450
* 2ND ENTRY AT CSTGC5,FROM THE 'COPY AN OLD FILE' ROUTINE .          * A2707460
*           -WRITE A TAPE MARK ONTO  'UPDTNEW' .                     * A2707470
*           -WRITE A TAPE MARK ONTO 'DUPLFILE' , IF ANY MODULE HAS   * A2707480
*         BEEN WRITTEN ON DUPLFILE .                                 * A2707490
*           -BRANCH TO CSTGC6.                                       * A2707500
*                                                                    * A2707510
* 3RD ENTRY AT CSTGC6,FROM THE 'DELETE AN OLD FILE' ROUTINE .        * A2707520
*           -READ THE FIRST RECORD OF THE NEXT OLD FILE.             * A2707530
*           -PRINT THE MESSAGE 'FILE TREATMENT FINISHED'.            * A2707540
*                                                                    * A2707550
*  EXIT  TO 'FILE TREATMENT ROUTINE' AT FLTGA                        * A2707560
*                                                                    * A2707570
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2707580
CSTGC    BAL   LINKE,PRLINA              SKIP 1 LINE                 A2707590
         TM    SWERR,X'01'               ERROR DETECTED FOR THIS     A2707600
         BC    8,CSTGC4                  *FILE - NO                  A2707610
         LA    WORK,ERME26               PRINT ERROR MESSAGE         A2707620
         BAL   LINKD,PRME1               *                           A2707630
         BAL   LINKE,PRLINA              SKIP 1 LINE                 A2707640
*  PRINT A MESSAGE FOR EACH ERROR CODE ALREADY PRINTED                A2707650
CSTGC4   LA    WORK,ERTBG+16             1ST(RIGHT)ERROR CODE ADDRES A2707660
         SR    WORKA,WORKA                                           A2707670
         LA    WORKA,COD(WORKA)          ERROR CODE LENGTH           A2707680
         LNR   WORKA,WORKA               NEGATIVE LENGTH             A2707690
         LA    WORKB,ERTBG               LAST(LEFT)ERROR CODE ADDRES A2707700
         LA    WORKD,MSGTBL              1ST(LEFT)ERROR MESSAGE ADDR A2707710
CSTGC2   CLC   BLANK(COD),0(WORK)        CODE NOT PRINTED            A2707720
```

```
         BC    8,CSTGC3                  YES                              A2707730
         MVC   PRZONE+6(83),0(WORKD)     PRINT THE MESSAGE WHICH IS       A2707740
         BAL   LINKE,PRLINA              *DEFINED BY THE CODE             A2707750
         OI    SWERR,X'02'               ERROR DETECTED AT JOB LEVEL      A2707760
CSTGC3   LA    WORKD,83(WORKD)           NEXT MESSAGE ADDRESS             A2707770
         BXH   WORK,WORKA,CSTGC2         TO NEXT CODE (RIGHT TO LEFT      A2707780
*      SEARCH 1ST RECORD OF NEXT FILE                                     A2707790
CSTGC5   BAL   LINKF,OWTMRK              WRITE TAPE MARK ON UPDTNEW       A2707800
         TM    DPLSW,X'01'               DUPLICATING PERFORMED            A2707810
         BC    8,CSTGC6                  NO -BRANCH                       A2707820
         BAL   LINKF,ODTMRK              WRITE TAPE MARK ON DUPLFILE      A2707830
CSTGC6   NI    DPLSW,X'FE'               DUPLICATING NO PERFORMED         A2707840
         TM    SWOLD,X'03'               IS OLD DATA END                  A2707850
         BC    1,CSTGC7                  YES BRANCH                       A2707860
         TM    SWOLD,X'03'               THE LAST FUNCTION WAS A          A2707870
         BC    8,CSTGC7                  *FILE INSERTION-YES              A2707880
         BAL   LINKC,RDOLD               READ 1 OLD RECORD                A2707890
         BC    15,CSTGC7                 DATA END (7F)2                   A2707900
         BC    15,CSTGC7                 (FILE END (7F)1)                 A2707910
         BC    15,CSTGC7                 CS END (OLD RECORD)              A2707920
CSTGC7   BAL   LINKE,PRLINA              NOT CS END(OLD R.)SKIP LINE      A2707930
         LA    WORK,MESS27               PRINT MESSAGE                    A2707940
         BAL   LINKD,PRME3               *                                A2707950
         BC    15,FLTGA                  TO NEXT FILE                     A2707960
         EJECT                                                            A2707970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2707980
*                                                                         * A2707990
* NORMAL END OF JOB  ROUTINE                                              * A2708000
*                                                                         * A2708010
* ENTRY AT NLSTOP ,FROM THE 'FILE TREATMENT' ROUTINE .                    * A2708020
*                                                                         * A2708030
* FUNCTIONS   -WRITE A TAPE MARK ON 'UPDTNEW' .                           * A2708040
*             -WRITE A TAPE MARK ON'DUPLFILE'IF ANY FILE HAS BEEN         * A2708050
*                   DUPLICATED.                                           * A2708060
*             -WRITE A TAPE MARK ON THE'PRINTER'DEVICE IF THIS DEVICE     * A2708070
*                   ASSIGNED IN DEVSUP CARD IS A MAGNETIC TAPE.           * A2708080
*                                                                         * A2708090
*             -PRINT THE MESSAGE 'END OF UPDATING'.                       * A2708100
*             -PRINT THE MESSAGE 'POSSIBLE UPDATING ERROR' IF ANY         * A2708110
*                ERROR HAS OCCURED DURING THIS UPDATING.                  * A2708120
*                                                                         * A2708130
*  EXIT  TO   -JOBEND ( REWIND ROUTINE).                                 * A2708140
*                                                                         * A2708150
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2708160
NLSTOP   BAL   LINKF,OWTMRK              WRITE LAST T.M. ON UPDTNEW       A2708170
         TM    DPLSW,X'02'               DUPLICATING PERFORMED            A2708180
         BC    8,NLST1                   NO -BRANCH                       A2708190
         BAL   LINKF,ODTMRK              WRITE LAST T.M. ON DUPLFILE      A2708200
NLST1    LA    WORK,MESS34               PRINT MESSAGE                    A2708210
         BAL   LINKD,PRME3               *                                A2708220
         LA    WORK,MESS34               PRINT A MESSAGE                  A2708230
         BAL   LINKE,EDCSL2              *ON PRINTER-KEYBOARD             A2708240
         TM    SWERR,X'02'               ERROR DETECTED                   A2708250
         BC    8,NLST2                   NO                               A2708260
         LA    WORK,ERME37               PRINT ERROR MESSAGE              A2708270
```

```
         BAL   LINKD,PRME1                 *                          A2708280
         LA    WORK,ERME37                 ERR-MESSAGE TO             A2708290
         BAL   LINKE,EDCSL1                *PRINTER-KEYBOARD          A2708300
NLST2    BAL   LINKF,OPRSKP                SKIP TO NEXT PAGE          A2708310
         MVC   DVNAME(8),OPRIN2+2          SEARCH TYPE OF PRINTER     A2708320
         BAL   LINKF,DEVTST                * DEVICE                   A2708330
         BC    15,NLST4                                               A2708340
NLST4    CLC   DVTYP(4),TAPTYP             PRINTER IS REALLY A PRINTER A2708350
         BC    7,NLST3                     *DEVICE,IS NOT TAPE OPTION A2708360
         BAL   LINKF,OPTMRK                WRITE T.M. ON 'PRINTER'    A2708370
NLST3    BC    15,JOBEND                   TO 'REWIND AND STOP'ROUTINE A2708380
         EJECT                                                        A2708390
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2708400
*                                                                     * A2708410
* EXCEPTIONAL END OF JOB ROUTINE                                      * A2708420
*                                                                     * A2708430
* 1ST ENTRY AT ERSTPA,FROM 'CONSOLE MESSAGE WRITING'(EXCEPTION.RETURN)* A2708440
*                                                                     * A2708450
* 2ND ENTRY AT ERSTPC OR ERSTPH                                       * A2708460
*                FROM THE I/O ROUTINES (EXCEPTIONAL RETURN) .         * A2708470
*              (01-UNKNOWN SYMBOL )                                   * A2708480
*              (02-DEVICE MALFUNCTION)                                * A2708490
*              (03-STICKER DETECTED ON  A  TAPE) .                    * A2708500
*                                                                     * A2708510
* 3RD ENTRY AT ERSTPD,FROM 'INITIALIZATION ROUTINE'                   * A2708520
*              (THE 1ST CARD IS NOT A / UPDATE CARD)                  * A2708530
*              (THE 1ST OLD RECORD IS A TAPE MARK)                    * A2708540
*                                                                     * A2708550
* 4TH ENTRY AT ERSTPE,FROM THE 'PROGRAM CHECK'ROUTINE .               * A2708560
*                                                                     * A2708570
* FUNCTIONS   -WRITE ERROR MESSAGES (PRINTER AND PRINTER-KEYBOARD) .  * A2708580
*                                                                     * A2708590
*  EXIT  TO   -JOBEND (REWIND ROUTINE) .                             * A2708600
*                                                                     * A2708610
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2708620
ERSTPD   BCR   0,0                                                    A2708630
ERSTPC   LA    WORK,ERME28                 WRITE ERROR MESSAGE        A2708640
         BAL   LINKE,EDCSL1                *                          A2708650
ERSTPB   TM    SWPRER,X'01'                WAS PRINTER ERROR          A2708660
         BC    1,ERSTPF                    *YES-                      A2708670
         BAL   LINKE,PRLINA                SKIP 1 LINE                A2708680
         LA    WORK,ERME28                 PRINT ERROR MESSAGE        A2708690
         BAL   LINKD,PRME1                 *                          A2708700
         BAL   LINKF,OPRSKP                SKIP TO NEXT PAGE          A2708710
ERSTPF   TM    SWDUMP,X'01'                DUMP REQUESTED BY PROGRAM  A2708720
         BC    8,ERSTOP                    NO                         A2708730
*        DUMP  AND STOP                                               A2708740
         BC    15,ERSTOP                                              A2708750
         CNOP  2,4                                                    A2708760
         SVC   12                          DUMP                       A2708770
         DC    A(0)                                                   A2708780
         DC    F'16384'                                               A2708790
         DC    C'EDITDUMP'                                            A2708800
ERSTOP   BC    15,JOBEND                   TO 'REWIND AND STOP'ROUTINE A2708810
ERSTPH   OI    SWPRER,X'01'                PRINTER ERROR              A2708820
```

```
         BC    15,ERSTPC                                                  A2708830
ERSTPE   OI    SWDUMP,X'01'                                               A2708840
         BC    15,ERSTPD                                                  A2708850
ERSTPA   OI    SWDUMP,X'01'                                               A2708860
         BC    15,ERSTPB                                                  A2708870
         EJECT                                                            A2708880
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2708890
*                                                                         * A2708900
* REWIND ROUTINE                                                         * A2708910
*                                                                         * A2708920
* ENTRY AT JOBEND,FROM THE 'NORMAL END OF JOB ' ROUTINE                  * A2708930
*                  OR THE'EXCEPTIONAL END OF JOB'ROUTINE .               * A2708940
*                                                                         * A2708950
* FUNCTIONS   -REWIND EACH DEVICE,DEFINED  AS TAPE IN DEVSUP CARDS       * A2708960
*              (UPDTOLD,UPDTNEW,UPDTCORR,DUPLFILE,PRINTER) .              * A2708970
*                                                                         * A2708980
* STOP (ENTER    WAIT STATE )                                            * A2708990
*                                                                         * A2709000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2709010
JOBEND   SVC   9                          ENABLE                         A2709020
         SR    WORKD,WORKD                LOAD '5' IN WORKD              A2709030
         LA    WORKD,5(WORKD)             *                              A2709040
JOBEN5   MVC   DVNAME(8),IRECD+2          WORKD=5,SEARCH DEVICE TYPE     A2709050
         BC    15,JOBEN8                  *AND ADDRESS OF 'UPDTOLD'      A2709060
JOBEN4   MVC   DVNAME(8),OPRIN2+2         WORKD=4,SEARCH DEVICE TYPE     A2709070
         BC    15,JOBEN8                  *AND ADDRESS OF 'PRINTER'      A2709080
JOBEN3   MVC   DVNAME(8),OWRIT2+2         WORKD=3,SEARCH DEVICE TYPE     A2709090
         BC    15,JOBEN8                  *AND ADDRESS OF 'UPDTNEW'      A2709100
JOBEN2   MVC   DVNAME(8),ODUPL2+2         WORKD=2,SEARCH DEVICE TYPE     A2709110
         BC    15,JOBEN8                  *AND ADDRESS OF 'DUPLFILE'     A2709120
JOBEN1   MVC   DVNAME(8),ICARD+2          WORKD=1,IDEM FOR 'UPDTCORR'    A2709130
JOBEN8   BAL   LINKF,DEVTST               SEARCH DEVICE TYPE AND ADD.    A2709140
         BC    15,JOBEN6                  NOT FOUND                      A2709150
JOBEN6   BCTR  WORKD,0                    FOUND-WORKD=WORKD-1            A2709160
         CLC   DVTYP(4),TAPTYP            THE DEVICE IS A TAPE DEVICE    A2709170
         BC    7,JOBEN7                   NO                             A2709180
         MVC   REWADD(2),DVADD            STORE THIS DEV.ADD TO ADDR.    A2709190
         BAL   LINKF,REWIND               *AREA OF SVC13,AND REWIND      A2709200
JOBEN7   CH    WORKD,KB4                  WORKD =                        A2709210
         BC    8,JOBEN4                   *=4 REWIND UPDTOLD             A2709220
         CH    WORKD,KB2                  *                              A2709230
         BC    2,JOBEN3                   *=3 REWIND UPDTNEW             A2709240
         BC    6,JOBEN2                   *=2 REWIND DUPLFILE            A2709250
         LTR   WORKD,WORKD                *                              A2709260
         BC    7,JOBEN1                   *=1 REWIND UPDTCORR            A2709270
STOP     SVC   19                         *=0 STOP                       A2709280
         BC    15,*-2                                                    A2709290
         EJECT                                                           A2709300
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2709310
*                                                                         * A2709320
* RECORDS TREATMENT ROUTINE                                              * A2709330
*                                                                         * A2709340
* ENTRY AT RCTGB                                                         * A2709350
*     -1- FROM 'CS TREATMENT ROUTINE'WHEN A CORRECTION'PER RECORD' IS     * A2709360
*              REQUESTED .                                                * A2709370
```

```
*     -2- FROM THIS ROUTINE WHEN A CORRECTION HAS BEEN PERFORMED IN   * A2709380
*               THE CURRENT MODULE .                                  * A2709390
*                                                                     * A2709400
* DATA    - THE FOLLOWING CARDS AND RECORDS HAVE BEEN READ #          * A2709410
*     - A RIS CARD RELATED TO THE CURRENT MODULE (OR A RIS CARD RELA- * A2709420
*               TED TO  THE NEXT MODULE, OR A / UPDATE CARD, OR NO    * A2709430
*               MORE CARD ) .                                         * A2709440
*     - 1 OLD RECORD FROM THE CURRENT MODULE (OR THE 1ST RECORD OR THE* A2709450
*               NEXT MODULE,OR A TAPE MARK) .                         * A2709460
*                                                                     * A2709470
* FUNCTIONS                                                           * A2709480
*              -SELECT THE ROUTINE TO BE USED (AND CALL IT BY BAL LINK)* A2709490
*              1-SKCRDA(SKIP UPDTCORR UP TO NEXT RIS CARD(NEXT CS) )  * A2709500
*                  IF THE REQUESTED FUNCTION CANNOT BE PERFORMED .    * A2709510
*              2-SKLDB (COPY 1 OLD RECORD) IF THE READ OLD RECORD IS  * A2709520
*                  NOT THAT SPECIFIED IN THE READ RIS CARD .          * A2709530
*              3-SKLDC(COPY THE REMAINDER OF THE CURRENT MODULE) ,    * A2709540
*                  IF THE LAST RIS CARD RELATED TO THIS MODULE HAS    * A2709550
*                  BEEN PERFORMED .                                   * A2709560
*              4-REPLA (REPLACE N OLD RECORDS) * DEPENDING ON THE     * A2709570
*              5-SUPPR (DELETE  N OLD RECORDS) * FUNCTION REQUESTED BY* A2709580
*              6-COUNT (NUMBER  N OLD RECORDS) * THE READ RIS CARD .  * A2709590
*              7-INSEB (INSERT  N NEW RECORDS) *                      * A2709600
*                                                                     * A2709610
* EXIT   TO                                                           * A2709620
*     -RCTGB (TREAT THE NEXT RECORDS SET) IN ALL  ABOVE CASES .       * A2709630
*     -CSTGA (CS TREATMENT ROUTINE) IF THE CURRENT MODULE IS          * A2709640
*           COMPLETLY PROCESSED .                                     * A2709650
*                                                                     * A2709660
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2709670
RCTGB    TM    SWCS,X'03'              ARE BOTH INPUT CS END         A2709680
         BC    1,CSTGA                 YES-TO NEXT CS                A2709690
         TM    SWCSCE,X'02'            CARD CS END ONLY              A2709700
         BC    1,RCTGN                 YES-BRANCH                    A2709710
         TM    RLVALD,X'01'            IS INVALID RIS READ           A2709720
         BC    1,RCTGD                 YES-BRANCH (ERROR)            A2709730
         CLI   RLCNT,X'C9'             READ RIS - INSERT REQUEST     A2709740
         BC    7,RCTGP                 NO                            A2709750
         CLC   RLNUM1(NUM),BLANK       INSERT AT BEGINNING OF CS     A2709760
         BC    8,RCTGE                 YES                           A2709770
         BC    15,RCTGG                NO- INSERT IN CS              A2709780
RCTGP    TM    SWCSOE,X'01'            CURRENT OLD CS END            A2709790
         BC    1,RCTGI                 YES                           A2709800
         CLC   RLBIN1(4),OLBIN         IS EXPECTED OLD RECORD        A2709810
         BC    7,RCTGH                 NO                            A2709820
         CLI   RLCNT,X'D9'             READ RIS - REPLACE REQUEST    A2709830
         BC    8,RCTGK                 YES                           A2709840
         CLI   RLCNT,X'D5'             READ RIS - NUMBERING REQU.    A2709850
         BC    8,RCTGK1                YES                           A2709860
         BAL   LINKA,SUPPR             SUPPRESS OLD RECORDS          A2709870
         BC    15,RCTGB                TO NEXT SET                   A2709880
RCTGK    BAL   LINKA,REPLA             REPLACE OLD RECORDS           A2709890
         BC    15,RCTGB                TO NEXT SET                   A2709900
RCTGK1   BAL   LINKA,COUNT             NUMBER OLD RECORDS            A2709910
         BC    15,RCTGB                TO NEXT SET                   A2709920
```

```
RCTGN    BAL   LINKB,SKLDC                COPY THE CS REMAINDER       A2709930
         BC    15,RCTGB                   TO NEXT SET                 A2709940
RCTGD    LA    WORK,ERME06                PRINT ERROR MESSAGE         A2709950
RCTGD1   BAL   LINKD,PRME1                *AND                        A2709960
         BC    15,RCTGL                   *CONTINUE                   A2709970
RCTGE    TM    SWCSO1,X'04'               IS 1ST RECORD OF OLD CS     A2709980
         BC    1,RCTGF                    NO-BRANCH (ERROR)           A2709990
RCTGE1   BAL   LINKA,INSEB                INSERT 1ST SET OF NEW CS    A2710000
         BC    15,RCTGB                   TO NEXT SET                 A2710010
RCTGF    LA    WORK,ERME09                PRINT ERROR MESSAGE         A2710020
         BC    15,RCTGD1                  *AND CONTINUE               A2710030
RCTGG    CLC   RLBIN1(4),OPBIN            EXPECTED OLD RECORD IS READ A2710040
         BC    7,RCTGH                    NO-BRANCH                   A2710050
         BC    15,RCTGE1                  INSERT 1 SET IN NEW CS      A2710060
RCTGH    TM    SWCSOE,X'01'               IS OLD CS END               A2710070
         BC    8,RCTGM                    YES-BRANCH                  A2710080
         LA    WORK,ERME09                PRINT ERROR MESSAGE         A2710090
         BC    15,RCTGD1                  *AND CONTINUE               A2710100
RCTGI    LA    WORK,ERME10                PRINT ERROR MESSAGE         A2710110
         BC    15,RCTGD1                  *AND CONTINUE               A2710120
RCTGL    BAL   LINKB,SKCRDA               SKIP MODIF CARDS UP TO      A2710130
         BC    15,RCTGB                   *THE NEXT CS                A2710140
RCTGM    BAL   LINKB,SKLDB                TREAT 1 OLD RECORD          A2710150
         BC    15,RCTGB                   TO NEXT SET                 A2710160
         EJECT                                                        A2710170
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710180
*                                                                     * A2710190
* REPLACE ROUTINE INITIALIZATION  - CALLED  BY BAL LINKA,REPLA        * A2710200
*             -PRINT THE READ RIS CARD-IMAGE .                        * A2710210
*             -PRINT 'REPLACE REQUEST' .                              * A2710220
*             -INITIALIZE SWITCHES FOR THE'RISN' ROUTINE .            * A2710230
*             -BRANCH TO RISNC (REPLACE,INSERT,SUPPRESS,NUMBER,ROUTINE* A2710240
*                                                                     * A2710250
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710260
REPLA    NI    SWSET,X'00'                RESET SWSET                 A2710270
         NI    SWIDT,X'FE'                OLD CS ID NON CHANGED       A2710280
         MVC   PRZONE+9(79),RLZONE+1      PRINT RIS CARD              A2710290
         BAL   LINKE,PRLINA               *                           A2710300
         LA    WORK,MESS23                PRINT 'REPLACE  REQUEST'    A2710310
         BAL   LINKD,PRME3                *                           A2710320
         BAL   LINKE,PRLINA               SKIP 1 LINE                 A2710330
         MVI   SWRISN,X'08'               SWITCH-REPLACE REQUEST      A2710340
         BC    15,RISNC                                               A2710350
         EJECT                                                        A2710360
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710370
*                                                                     * A2710380
* SUPPRESS ROUTINE INITIALIZATION - CALLED  BY BAL LINKA,SUPPR        * A2710390
*             -PRINT THE READ RIS CARD-IMAGE .                        * A2710400
*             -PRINT 'SUPPRESS REQUEST' .                             * A2710410
*             -INITIALIZE SWITCHES FOR THE 'RISN' ROUTINE .           * A2710420
*             -SAVE READ RIS CARD IN PRECEDING RIS CARD AREA .        * A2710430
*             -CREATE A DUMMY MODIFICAT. CARD IN READING BUFFER .     * A2710440
*             -BRANCH TO RISNC(REPLACE,INSERT,SUPPRESS,NUMBER,ROUTINE)* A2710450
*                                                                     * A2710460
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710470
```

```
SUPPR    NI    SWSET,X'00'                BIT2,LAST MODIF FOUND       A2710480
         OI    SWSET,X'28'                BIT4,1ST MODIF TREATED      A2710490
         NI    SWIDT,X'FE'                OLD C5 ID NON CHANGED       A2710500
         MVC   PRZONE+9(79),RLZONE+1      PRINT RIS CARD              A2710510
         BAL   LINKE,PRLINA               *                           A2710520
         LA    WORK,MESS24                PRINT 'SUPPRESS REQUEST'    A2710530
         BAL   LINKD,PRME3                *                           A2710540
         BAL   LINKE,PRLINA               SKIP 1 LINE                 A2710550
         MVI   SWRISN,X'02'               SWITCH-SUPPRESS REQUEST     A2710560
SUPPR1   BAL   LINKE,PRCRIS               READ RIS TO PRECEDING RIS   A2710570
         MVI   MLZONE,X'40'               CREATE                      A2710580
         MVC   MLZONE+1(79),MLZONE        * A DUMMY                   A2710590
         MVC   MLCSD(8),BLANK             * MODIF CARD                A2710600
         MVC   MLNUM(8),ZEROF             *                           A2710610
         MVC   MLBIN(4),ZERO              *                           A2710620
         MVC   MLNBR(4),ZERO              *                           A2710630
         MVC   MLVALD(1),ZERO             *                           A2710640
         BC    15,RISNC                                               A2710650
         EJECT                                                        A2710660
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710670
*                                                                     * A2710680
* INSERT ROUTINE INITIALIZATION                                       * A2710690
*                 INSERT A MODULE  - CALLED  BY BAL LINKA,INSEA       * A2710700
*                 INSERT N RECORDS - CALLED  BY BAL LINKA,INSEB       * A2710710
*             -PRINT THE READ RIS CARD-IMAGE .                        * A2710720
*             -PRINT 'INSERT REQUEST'.                                * A2710730
*             -INITIALIZE SWITCHES FOR THE 'RISN' ROUTINE .           * A2710740
*             -BRANCH TO RISNC(REPLACE,INSERT,SUPPRESS,NUMBER,ROUTINE)* A2710750
*                                                                     * A2710760
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710770
INSEA    OI    SWCS,X'05'                 OLD C5 END                  A2710780
INSEB    NI    SWSET,X'00'                OLD SET END                 A2710790
         OI    SWSET,X'15'                *                           A2710800
         NI    SWIDT,X'FE'                OLD C5 ID NON CHANGED       A2710810
         MVC   PRZONE+9(79),RLZONE+1      PRINT RIS CARD              A2710820
         BAL   LINKE,PRLINA               *                           A2710830
         LA    WORK,MESS22                PRINT 'INSERT   REQUEST'    A2710840
         BAL   LINKD,PRME3                *                           A2710850
         BAL   LINKE,PRLINA               SKIP 1 LINE                 A2710860
         MVI   SWRISN,X'04'                                           A2710870
         BC    15,RISNC                                               A2710880
         EJECT                                                        A2710890
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2710900
*                                                                     * A2710910
* NUMBERING ROUTINE INITIALIZATION - CALLED  BY BAL LINKA,COUNT       * A2710920
*             -PRINT THE READ RIS CARD-IMAGE .                        * A2710930
*             -PRINT 'NUMBERING REQUEST'.                             * A2710940
*             -INITIALIZE SWITCHES FOR THE 'RISN' ROUTINE .           * A2710950
*             -SAVE READ RIS CARD IN PRECEDING RIS CARD AREA .        * A2710960
*             -CREATE A DUMMY MODIFICAT. CARD IN READING BUFFER .     * A2710970
*             -BRANCH TO RISNC(REPLACE,INSERT,SUPPRESS,NUMBER,ROUTINE)* A2710980
*                                                                     * A2710990
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2711000
COUNT    NI    SWSET,X'00'                BIT2,LAST MODIF FOUND       A2711010
         OI    SWSET,X'28'                BIT4,1ST MODIF TREATED      A2711020
```

```
         NI    SWIDT,X'FE'                 OLD CS ID NON CHANGED       A2711030
         CLI   RLMOD,X'C3'                 MODE'BY CS'IN READ RIS CARD A2711040
         BC    7,COUNT1                    NO                          A2711050
         CLC   RLCSDI(8),BLANK             NEW CS SPECIFIED IN READ    A2711060
         BC    8,COUNT1                    *RIS CARD -NO               A2711070
         OI    SWIDT,X'01'                 CHANGE CS ID                A2711080
COUNT1   MVC   PRZONE+9(79),RLZONE+1       PRINT RIS CARD              A2711090
         BAL   LINKE,PRLINA                *                           A2711100
         LA    WORK,MESS25                 PRINT 'COUNT REQUEST'       A2711110
         BAL   LINKD,PRME3                 *                           A2711120
         BAL   LINKE,PRLINA                SKIP 1 LINE                 A2711130
         MVI   SWRISN,X'01'                                            A2711140
         BC    15,SUPPR1                                               A2711150
         EJECT                                                         A2711160
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2711170
*                                                                       A2711180
* RISN ROUTINE (REPLACE,INSERT,SUPPRESS,NUMBER A MODULE OR RECORDS)     A2711190
*                                                                       A2711200
* ENTRY AT RISN,FROM ANY INITIALIZATION SUB ROUTINE OF THIS ROUTINE     A2711210
*    ( 'REPL , SUPPR , INSE , COUNT ' )  .                              A2711220
*                                                                       A2711230
* RETURN TO CALLER  BY  BCR 15,LINKA                                    A2711240
*                                                                       A2711250
* PROCESSING-                                                           A2711260
*                                                                       A2711270
*    1 RISNB -READ AN OLD RECORD ,                                      A2711280
*       - DURING A REPLACING,A DELETING OR A NUMBERING OPERATION        A2711290
*       - UP TO THE NEXT MODULE (MODULE PROCESSING),                    A2711300
*         OR  UP  TO  THE  OLD  RECORD  FOLLOWING  THE  LAST            A2711310
*         OLD RECORD TO BE CORRECTED (RECORDS PROCESSING).              A2711320
*    2 RISNC -READ AN 'UPDTCORR' CARD                                   A2711330
*       - DURING AN INSERTING OR A REPLACING OPERATION                  A2711340
*       - UP TO THE NEXT / UPDATE OR RIS CARD .                         A2711350
*    3 RISNF -WRITE 1 NEW RECORD ONTO  'UPDTNEW' ,                      A2711360
*       - DURING A NUMBERING,AN INSERTING OR A REPLACING OPERATION      A2711370
*         (INSERTION PHASE OF A REPLACEMENT) .                          A2711380
*       - FROM AN OLD RECORD(NUMBERING),OR FROM A MODIFICATION CARD     A2711390
*         (INSERTION OR REPLACEMENT ) .                                 A2711400
*    4 RISNF3- WRITE A DUPLICATE OF THE WRITTEN NEW RECORD , IF SO      A2711410
*        REQUESTED .                                                    A2711420
*    5 RISNG - PRINT ON THE SAME LINE                                   A2711430
*       - THE NEW RECORD IMAGE (REPLACEMENT,INSERTION OR NUMBERING ) .  A2711440
*       - THE IDENTIFICATION OF THE CORRESPONDING CORRECTION CARD       A2711450
*         (REPLACEMENT,OR INSERTION ) .                                 A2711460
*       - THE IDENTIFICATION OF THE CORRESPONDING OLD RECORD            A2711470
*         (REPLACEMENT,DELETING OR NUMBERING) .                         A2711480
*    6 RISNV -BRANCH TO RISNB IF THE FUNCTION HAS BEEN PERFORMED ,      A2711490
*        OTHERWISE BRANCH TO 'RISNP' .                                  A2711500
*    7 RISNP - END OF ROUTINE                                           A2711510
*       - READ THE NEXT CARD (AFTER DELETING OR NUMBERING FUNCTION)     A2711520
*       (THE NEXT CARD HAS ALREADY BEEN READ AFTER REPLACEMENT OR       A2711530
*        INSERTION) .                                                   A2711540
*       - TEST THE READ CARD.  THERE IS AN ERROR IF                     A2711550
*             -IT IS A MODIFICATION CARD (NEITHER / UPDATE NOR RIS)     A2711560
*             -IT IS A RIS CARD REQUESTING A REPLACEMENT,A DELETING     A2711570
```

```
*           OR A NUMBERING OF THE OLD MODULE WHICH HAS BEEN TREATED. * A2711580
*       - IF AN ERROR IS DETECTED,A MESSAGE IS PRINTED AND THE RIS    * A2711590
*         AND MODIFICATION CARDS RELATED TO THE SAME MODULE ARE       * A2711600
*         PRINTED WITHOUT INVOLVING CORRECTIONS .                     * A2711610
*    8 RISNL - ROUTINE INTERRUPTED BEFORE NORMAL END                  * A2711620
*       - DURING AN 'UPDTCORR' CARD READING , AN RIS CARD IS NOT      * A2711630
*         FOLLOWED BY A MODIFICATION CARD , OR THE IDENTIFICATION OF  * A2711640
*         A MODIFICATION CARD IS NOT CORRECT .                        * A2711650
*       - A MESSAGE IS PRINTED .                                      * A2711660
*       - THE REMAINDER OF THE CURRENT OLD MODULE IS COPIED ON UPDTNEW* A2711670
*       - THE RIS AND MODIFICATION CARDS RELATED TO THE SAME MODULE   * A2711680
*         ARE PRINTED WITHOUT INVOLVING CORRECTIONS .                 * A2711690
*                                                                     * A2711700
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2711710
         EJECT                                                        A2711720
RISNB    TM    SWSEOL,X'10'          LAST OLD RECORD OF SET IS      A2711730
         BC    1,RISNC               *TREATED(TO INSERT ACTION)     A2711740
         CLC   OLBIN(4),RPBIN2       LAST OLD RECORD FOUND          A2711750
         BC    4,RISNK               NO                             A2711760
         CLI   RPMOD,X'C3'           MODE IN PRECEDENT RIS IS       A2711770
         BC    8,RISNK               *PER CS -YES                   A2711780
         OI    SWSEOL,X'10'          LAST OLD RECORD TREATED        A2711790
RISNK    BAL   LINKC,RDOLD           READ 1 OLD RECORD              A2711800
         BC    15,RISNA              (OLD DATA END (7F2))           A2711810
         BC    15,RISNA              OLD FILE END (7F1)             A2711820
         BC    15,RISNA              OLD CS   END                   A2711830
         CLC   OLBIN(4),RPBIN2       LAST OLD RECORD FOUND          A2711840
         BC    12,RISNC              NO                             A2711850
         CLI   RPMOD,X'C3'           MODE IN PRECEDENT RIS IS       A2711860
         BC    8,RISNC               *PER CS -YES                   A2711870
RISNA    OI    SWSEOL,X'10'          LAST OLD RECORD TREATED        A2711880
RISNC    TM    SWSECL,X'20'          LAST MODIF TREATED             A2711890
         BC    1,RISNF               *(TO SUPPR.OR COUNT ACTION)    A2711900
         BAL   LINKC,RDCRD           READ A CARD                    A2711910
         BC    15,RISNS              CARD  DATA END                 A2711920
         BC    15,RISNS              CARD  FILE END                 A2711930
         BC    15,RISNS              CARD  CS   END (RIS)           A2711940
         BC    15,RISNS                   NOT CS END (RIS)         A2711950
         TM    SWSEC1,X'08'          1ST MODIF CARD OF SET          A2711960
         BC    8,RISND               YES-BRANCH                     A2711970
         CLC   MLCSD(CS),MPCSD       IS NOT SAME CS IN READ MOD.    A2711980
         BC    7,RISNL               *AS IN PRECEDING-ERROR         A2711990
         BC    15,RISNE              IS THE SAME CS  -BRANCH        A2712000
RISND    CLI   RPMOD,X'C3'           CORRECTION MODE PER CS         A2712010
         BC    8,RISNW               YES                            A2712020
         CLC   MLCSD(CS),RPCSD1      INCORRECT CS IDENT IN 1ST      A2712030
         BC    7,RISNL               *MODIF CARD AFTER RIS-YES      A2712040
         BC    15,RISNE              NO                             A2712050
RISNW    CLC   MLCSL(1),ZERO         THE LENGTH OF CS-ID PART IN    A2712060
         BC    8,RISNL               *READ MODIF IDENTIFICATION     A2712070
         CLC   MLCSL(1),KB8+1        *EQUAL 0 OR 8                  A2712080
         BC    8,RISNL               YES-ERROR                      A2712090
         BC    15,RISNE              NO                             A2712100
RISNS    TM    SWSEC1,X'08'          NOT ANY MODIF CARD HAS BEEN    A2712110
         BC    8,RISNL               *READ AFTER RIS-ERROR          A2712120
```

```
         OI    SWSECL,X'20'              LAST MODIF TREATED          A2712130
         TM    SWSEOL,X'10'              LAST OLD RECORD OF SET      A2712140
         BC    8,RISNG                   * HAS BEEN TREATED-NO       A2712150
         BC    15,RISNV                  * YES                       A2712160
RISNE    TM    MLVALD,X'01'              NOT INVALID NUMBER IN READ  A2712170
         BC    8,RISNF                   *MODIF CARD                 A2712180
         MVC   ERTBG+16(COD),MSGTBL      I TO ERROR CODES STORAGE    A2712190
         MVC   ERCOD+16(COD),MSGTBL      I TO PRINT BUFFER           A2712200
RISNF    TM    SWRISN,X'01'              IS COUNT REQUEST            A2712210
         BC    1,RISNF1                  YES-BRANCH                  A2712220
         TM    SWSECL,X'20'              LAST MODIF CARD OF SET IS   A2712230
         BC    1,RISNG                   *TREATED(SUPPRESS ACTION)   A2712240
         BAL   LINKC,ENRMA               STORE MODIF CARD TO NEW     A2712250
         BC    15,RISNF3                 *RECORD BUFFER              A2712260
RISNF1   TM    SWSEOL,X'10'              LAST OLD RECORD HAS BEEN    A2712270
         BC    1,RISNG                   *TREATED-YES                A2712280
         TM    SWSE01,X'04'              IS FIRST OLD RECORD OF SET  A2712290
         BC    8,RISNF2                  YES                         A2712300
         BAL   LINKC,ENRON               OLD RECORD TO NEW RECORD    A2712310
         BC    15,RISNF3                 *(NOT THE FIRST IN SET)     A2712320
RISNF2   BAL   LINKC,ENR01               1ST OLD TO NEW RECORD       A2712330
RISNF3   BAL   LINKF,OWRITE              NEW RECORD WRITING OPERAT.  A2712340
         CLI   RPDUPL,X'C4'              DUPLICATION REQUESTED       A2712350
         BC    7,RISNF4                  NO -BRANCH                  A2712360
         BAL   LINKF,ODUPL               NEW RECORD DUPLICATION      A2712370
         OI    DPLSW,X'03'               DUPLICATION PERFORMED       A2712380
RISNF4   BAL   LINKE,PREDA               NEW RECORD TO PRINT BUFFER  A2712390
         BAL   LINKE,PREDE               MODIF CARD TO PRINT BUFFER  A2712400
RISNG    TM    SWSEOL,X'10'              LAST OLD RECORD TREATED     A2712410
         BC    1,RISNX                   *(INSERT ACTION)-YES        A2712420
         BAL   LINKE,PREDD               OLD RECORD TO PRINT BUFFER  A2712430
         BC    15,RISNH                                              A2712440
RISNX    TM    SWSECL,X'20'              LAST MODIF CARD TREATED     A2712450
         BC    1,RISNV                   YES                         A2712460
RISNH    BAL   LINKE,PRLINA              PRINT ONE LIST LINE         A2712470
RISNV    OI    SWSET,X'0C'               1ST OLD REC.AND MOD.TREATED A2712480
RISNJ    TM    SWSET,X'30'               BOTH INPUT SETS END         A2712490
         BC    12,RISNB                  NO                          A2712500
*   END OF FUNCTION AT END  OF BOTH SETS (CORRECT)                    A2712510
RISNP    TM    SWRISN,X'03'              WAS SUPPR OR NUMBER.ROUTINE A2712520
         BC    4,RISNT                   YES                         A2712530
         TM    SWCSCE,X'02'              IS CARD CS END              A2712540
         BCR   1,LINKA                   YES-RETURN TO CALLING       A2712550
         CLC   RLCOD(4),RISCOD           AN RIS CARD HAS BEEN READ   A2712560
         BC    7,RISNQ                   NO-IS MODIF CARD-ERROR      A2712570
         BC    15,RISNR                                              A2712580
RISNT    BAL   LINKC,RDCRD               READ A CARD                 A2712590
         BC    15,0(LINKA)               CARD DATA END               A2712600
         BC    15,0(LINKA)               CARD FILE END               A2712610
         BC    15,0(LINKA)               CARD CS   END(RIS)          A2712620
         BC    15,RISNR                  NOT  CS   END(RIS)          A2712630
RISNQ    LA    WORK,ERME14               ERROR                       A2712640
         BAL   LINKD,PRME1               *PRINT ERROR MESSAGE        A2712650
         BAL   LINKB,SKCRDA              SKIP MODIF FILE UP TO 1ST   A2712660
         BCR   15,LINKA                  *RIS (NEXT CS)              A2712670
```

```
RISNR    CLI   RPMOD,X'C3'                THE PRECEDING RIS MODE IS   A2712680
         BCR   7,LINKA                    *PER CS- NO                 A2712690
         CLI   RLCNT,X'C9'                THE READ RIS CARD REQUESTS  A2712700
         BCR   8,LINKA                    * AN INSERTION - YES        A2712710
         BC    15,RISNQ                   NO-ERROR                    A2712720
*  EXCEPTIONAL END OF ROUTINE                                         A2712730
RISNL    OI    SWSECE,X'02'               IS CARD SET END             A2712740
         LA    WORK,ERME12                PRINT ERROR MESSAGE         A2712750
         BAL   LINKD,PRME1                *                           A2712760
RISNM    TM    SWCSOE,X'01'               IS OLD CS END               A2712770
         BC    1,RISNN                    *                           A2712780
         BAL   LINKB,SKLDC                COPY THE REMAINDER OF CS    A2712790
RISNN    TM    SWCSCE,X'02'               IS CARD CS END              A2712800
         BCR   1,LINKA                    YES-RETURN TO CALLING       A2712810
         BAL   LINKB,SKCRDA               SKIP CARD SETS UP TO        A2712820
         BCR   15,LINKA                   *NEXT CS                L.  A2712830
         EJECT                                                        A2712840
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2712850
*                                                                     * A2712860
* SKIP 'UPDTCORR' CARDS   - CALLED  BY   BAL LINKB,SKCRDX             * A2712870
*                                                                     * A2712880
* ENTRY AT SKCRDA - SKIP RIS AND MODIFICATION CARDS RELATED  TO THE   * A2712890
*                   SAME MODULE .                                     * A2712900
* ENTRY AT SKCRDM - SKIP RIS AND MODIFICATION CARDS RELATED  TO THE   * A2712910
*                   SAME FILE .                                       * A2712920
*                                                                     * A2712930
* THE CARD IMAGES OF / UPDATE AND RIS CARDS ARE PRINTED .             * A2712940
* THE IDENTIFICATIONS OF MODIFICATION CARDS ARE PRINTED .             * A2712950
*                                                                     * A2712960
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2712970
SKCRDA   NI    SWSK1,X'00'                SKIP UP TO NEXT CS          A2712980
         BC    15,SKCRDB                                              A2712990
SKCRDM   OI    SWSK1,X'01'                SKIP UP TO NEXT FILE        A2713000
SKCRDB   CLC   RLCOD(4),RISCOD            RIS CARD IN READING BUFFER  A2713010
         BC    8,SKCRDC                   YES                         A2713020
         CLC   GLCOD(9),SLASH             / UPDATE IN READING BUFFER  A2713030
         BC    8,SKCRDD                   YES                         A2713040
         BAL   LINKE,PREDE                MODIF CARD TO PRINT BUFFER  A2713050
         BAL   LINKE,PRLINA               PRINT 1 LIST LINE           A2713060
SKCRDF   BAL   LINKC,RDCRD                CARD READING OPERATION      A2713070
         BC    15,0(LINKB)                DATA END                    A2713080
         BC    15,0(LINKB)                FILE END                    A2713090
         BC    15,SKCRDG                  CS END (RIS)                A2713100
         BC    15,SKCRDB                  NOT CS END(RIS)-            A2713110
         BC    15,SKCRDB                  MODIF CARD -TO NEXT CARD    A2713120
SKCRDG   TM    SWSK1,X'01'                UP TO / UPDATE              A2713130
         BC    1,SKCRDB                   YES-TO NEXT CARD READING    A2713140
         BCR   15,LINKB                   NO- RETURN TO CALLING       A2713150
SKCRDC   BAL   LINKE,PRLINA               SKIP 1 LINE                 A2713160
         MVC   PRZONE+9(79),RLZONE+1      READ RIS CARD TO PRINT BUFF A2713170
         BC    15,SKCRDE                  TO PRINTING   OPERATION     A2713180
SKCRDD   BAL   LINKE,PRLINA               SKIP 1 LINE                 A2713190
         MVC   PRZONE+8(80),GLZONE        / UPDATE CARD TO PRINT BUFF A2713200
SKCRDE   BAL   LINKE,PRLINA               PRINT RIS OR / UPDATE CARD  A2713210
         BAL   LINKE,PRLINA               SKIP 1 LINE                 A2713220
```

```
         BC    15,SKCRDF                                                  A2713230
         EJECT                                                            A2713240
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2713250
*                                                                       * A2713260
* COPY ALL OR PART OF AN OLD MODULE - CALLED  BY BAL LINKB,SKLDX        * A2713270
*                                                                       * A2713280
* ENTRY AT SKLDA - COPY A WHOLE MODULE FOR WHICH NO  CORRECTION HAS     * A2713290
*                  BEEN REQUESTED .                                     * A2713300
*                                                                       * A2713310
* ENTRY AT SKLDB - COPY A RECORD(WITHOUT CORRECTION)FROM A  MODULE FOR* A2713320
*                  WHICH SOME PARTIAL CORRECTIONS HAVE BEEN REQUESTED.* A2713330
*                                                                       * A2713340
* ENTRY AT SKLDC - COPY THE REMAINDER OF A MODULE FOR WHICH SOME        * A2713350
*                  CORRECTIONS HAVE BEEN REQUESTED .                    * A2713360
*                                                                       * A2713370
* PROCESSING     - EACH OLD RECORD IS WRITTEN ON'UPDTNEW' .             * A2713380
*                - (SKLDB AND C ONLY) EACH OLD RECORD IS WRITTEN ON     * A2713390
*                  'DUPLFILE',IF REQUESTED BY THE CORRESPONDING RIS CARD A2713400
*                - THE NEW RECORDS AND THEIR OLD IDENTIFICATIONS ARE    * A2713410
*                  PRINTED ,EXCEPT   (SKLDA)WHEN THE CORRESPONDING      * A2713420
*                  / UPDATE CARD REQUESTS THE OPTION 'PRINT CORRECTED * A2713430
*                  MODULES ONLY ' .                                     * A2713440
*                                                                       * A2713450
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2713460
SKLDA    MVI   SWSK,X'00'              CS COPY                          A2713470
         MVC   MESS03+17(8),OLIDT      PRINT  'THE CS XXX WILL BE       A2713480
         LA    WORK,MESS03             * COPIED '                       A2713490
         BAL   LINKD,PRME3                                              A2713500
         BC    15,SKLDD                                                 A2713510
SKLDB    MVI   SWSK,X'03'              RECORD COPY                      A2713520
         BC    15,SKLDD                                                 A2713530
SKLDC    MVI   SWSK,X'01'              REMAINDER OF CS COPY             A2713540
SKLDD    BAL   LINKC,ENROA             OLD RECORD TO NEW RECORD         A2713550
         BAL   LINKF,OWRITE            NEW RECORD WRITING OPERAT.       A2713560
         TM    SWSKAH,X'01'            ALL CS TREATED                   A2713570
         BC    8,SKLDJ                 YES-BRANCH                       A2713580
         TM    SWSKRD,X'02'            IS TREATMENT FOR 1 RECORD        A2713590
         BC    8,SKLDH                 NO -BRANCH                       A2713600
         CLI   RLDUPL,X'C4'            DUPL.REQUEST BY READ RIS         A2713610
         BC    7,SKLDJ                 NO -BRANCH                       A2713620
         BC    15,SKLDI                YES                              A2713630
SKLDH    CLI   RPDUPL,X'C4'            DUPL.REQUEST BY PRECDT.RIS       A2713640
         BC    7,SKLDJ                 NO -BRANCH                       A2713650
SKLDI    BAL   LINKF,ODUPL             DUPLICATING OPERATION            A2713660
         OI    DPLSW,X'03'             DUPLICATION PERFORMED            A2713670
SKLDJ    TM    SWSK,X'01'              TOTAL CS TREADED                 A2713680
         BC    8,SKLDK                 YES                              A2713690
         BC    15,SKLDE                NO- PRINT                        A2713700
SKLDK    CLC   ERCOD(10),BLANK         ANY ERROR CODE FOR THIS          A2713710
         BC    7,SKLDE                 *RECORD IN ERROR CODE AREA       A2713720
         CLC   ERCOD+10(8),BLANK       *OF PRINT BUFFER                 A2713730
         BC    7,SKLDE                 YES-ERROR-PRINT THIS RECORD      A2713740
         TM    OLVALD,X'01'            THIS RECORD IS INVALID           A2713750
         BC    1,SKLDE                 YES                              A2713760
         TM    PRINSW,X'01'            IS'PRINT ALL CS'OPTION           A2713770
```

```
         BC    8,SKLDG                    NO-BRANCH-NOT PRINT          A2713780
SKLDE    BAL   LINKE,PREDD                OLD RECORD TO PRINT BUFFER   A2713790
SKLDF    BAL   LINKE,PREDA                NEW RECORD TO PRINT BUFFER   A2713800
         BAL   LINKE,PRLINA               PRINT 1 LISTE  LINE          A2713810
SKLDG    BAL   LINKC,RDOLD                READ 1 OLD RECORD            A2713820
         BC    15,0(LINKB)                (OLD DATA END (7F)2)         A2713830
         BC    15,0(LINKB)                OLD FILE END (7F)1           A2713840
         BC    15,0(LINKB)                OLD CS END(OLD RECORD)       A2713850
         TM    SWSKRD,X'02'               COPY AN OLD RECORD           A2713860
         BCR   1,LINKB                    YES-RETURN TO CALLER         A2713870
         BC    15,SKLDD                   NO-READ THE NEXT OLD RECORD  A2713880
         EJECT                                                         A2713890
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2713900
*                                                                       * A2713910
* COPY OR DELETE ALL THE MODULES OF AN OLD FILE                         * A2713920
*                      - CALLED  BY      BC 15,SKLDX                    * A2713930
*                                                                       * A2713940
* ENTRY AT SKLDN - COPY AN OLD FILE                                     * A2713950
*                - ALL THE RECORDS ARE WRITTEN ON UPDTNEW ,UNTIL        * A2713960
*                  A TAPE MARK IS FOUND .                               * A2713970
*                - WHEN A TAPE MARK IS READ,BRANCH TO CSTGC5            * A2713980
*                  (END OF FILE TREATMENT) .                            * A2713990
*                                                                       * A2714000
* ENTRY AT SKLDM - DELETE AN OLD FILE                                   * A2714010
*                - ALL THE OLD RECORDS ARE READ UP TO A TAPE MARK .     * A2714020
*                - BRANCH TO CSTGC6 (END OF FILE TREATMENT) .           * A2714030
*                                                                       * A2714040
*       NO OLD RECORD IS PRINTED DURING THIS ROUTINE .                  * A2714050
*                                                                       * A2714060
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2714070
SKLDM    NI    SWSKF,X'00'                DELETING REQUESTED           A2714080
         BC    15,SKLDP                                                A2714090
SKLDN    OI    SWSKF,X'04'                COPY REQUESTED               A2714100
SKLDO    MVC   NWZONE(80),OLZONE          WRITE THE READ OLD RECORD    A2714110
         BAL   LINKF,OWRITE               *ON UPDTNEW                  A2714120
SKLDP    BAL   LINKC,RDOLD                READ AN OLD RECORD           A2714130
         BC    15,SKLDR                   (OLD DATA END (7F)2)         A2714140
         BC    15,SKLDR                   OLD FILE END (7F)1           A2714150
         BC    15,SKLDQ                   OLD CS END (OLD RECORD)      A2714160
SKLDQ    TM    SWSKF,X'04'                DELETING REQUESTED           A2714170
         BC    8,SKLDP                    YES-DELETE NEXT RECORD       A2714180
         BC    15,SKLDO                   NO-COPY NEXT RECORD          A2714190
SKLDR    TM    SWSKF,X'04'                DELETING REQUESTED           A2714200
         BC    8,CSTGC6                   YES-TO END OF FILE TREATMNT  A2714210
         BC    15,CSTGC5                  NO- TO END OF FILE TREATMNT  A2714220
         EJECT                                                         A2714230
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2714240
*                                                                       * A2714250
* BREACK UP OF AN  IDENTIFICATION FIELD  - CALLED  BY                   * A2714260
*  LA  WORK,IDT    IDT,8 BYTES ,  THE IDENTIFICATION FIELD    ( INPUT)* A2714270
*  LA WORKA,CS     CS ,8 BYTES ,  MODULE IDENTIFICATION PART  (OUTPUT)* A2714280
*  LA WORKB,NUM    NUM,8 BYTES ,  NUMERICAL PART OF IDT       (OUTPUT)* A2714290
*  LA WORKC,CSL    CSL,1 BYTE  ,  LENGTH OF MODULE IDENT.PART (OUTPUT)* A2714300
* BAL LINKD,RSCAN                                                       * A2714310
*                                                                       * A2714320
```

```
*      FOR EXAMPLE *IDT=ABC12345* GIVES* CS=ABC     * (5 BLANKS)      * A2714330
*                                       *NUM=00012345*                 * A2714340
*                                       *CSL=3*                        * A2714350
*                                                                      * A2714360
*                  *IDT=LMNB1234* GIVES* CS=LMNB     * (4 BLANKS)      * A2714370
*                                       *NUM=00001234*                 * A2714380
*                                       *CSL=4*                        * A2714390
*                                                                      * A2714400
*     -THE LENGTH OF THE MODULE IDENTIFICATION                         * A2714410
*        IS 3 IF THE 4TH LEFT BYTE OF IDT IS NUMERICAL OR BLANK ,      * A2714420
*        OTHERWISE IT IS 4 .                                           * A2714430
*                                                                      * A2714440
* - THE LEFT PART OF 'NUM'  IS COMPLETED BY      ZEROS IF THE LAST     * A2714450
*        RIGHT BYTE OF IDT IS NUMERICAL .                              * A2714460
*      FOR EXAMPLE *IDT=LMNB   1* GIVES* CS=LMNB     * (4 BLANKS)      * A2714470
*                                       *NUM=00000001*                 * A2714480
*                                       *CSL=4*                        * A2714490
*                                                                      * A2714500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2714510
RSCAN    MVC   0(8,WORKA),BLANK          BLANKS TO 'CS'                 A2714520
         MVC   0(8,WORKB),BLANK          BLANKS TO 'NUM'                A2714530
         TM    3(WORK),X'F0'             BRANCH                         A2714540
         BC    14,RSCANP                 * TO RSCANU                    A2714550
         CLI   3(WORK),X'F9'             * IF BYTE IDT+3 IS NUMERIC.    A2714560
         BC    12,RSCANU                 *     OR BLANK                 A2714570
         BC    15,RSCANA                 *                              A2714580
RSCANP   CLI   3(WORK),X'40'             * TO RSCANA                    A2714590
         BC    8,RSCANU                  * IN OTHER CASE                A2714600
RSCANA   MVI   0(WORKC),X'04'            CS LENGTH = 4                  A2714610
         MVC   0(4,WORKA),0(WORK)        STORE CS   (4 LEFT BYTES)      A2714620
         MVC   4(4,WORKB),4(WORK)        STORE NUM  (4 RIGHT BYTES)     A2714630
         BC    15,RSCANK                                                A2714640
RSCANU   MVI   0(WORKC),X'03'            CS LENGTH = 3                  A2714650
         MVC   0(3,WORKA),0(WORK)        STORE CS   (3 LEFT  BYTES)     A2714660
         MVC   3(5,WORKB),3(WORK)        STORE NUM  (5 RIGHT BYTES)     A2714670
*     STORE ZEROS IN LEFT PART OF NUM IF LAST RIGHT BYTE IS NUMERICAL   A2714680
RSCANK   TM    7(WORKB),X'F0'            LAST BYTE IN NUM IS NUMER.     A2714690
         BCR   14,LINKD                  *                              A2714700
         CLI   7(WORKB),X'F9'            *                              A2714710
         BCR   2,LINKD                   NO- RETURN TO CALLER           A2714720
RSCANN   CLI   0(WORKB),X'40'            BYTE NUM+N(N=0 TO 6)= BLANK    A2714730
         BCR   7,LINKD                   NO-RETURN TO CALLING           A2714740
RSCANM   MVC   0(1,WORKB),ZEROF           ZERO TO BYTE NUM+N             A2714750
         LA    WORKB,1(WORKB)            NEXT ADDR.OF NUM+N BYTE        A2714760
         BC    15,RSCANN                                                A2714770
         EJECT                                                          A2714780
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2714790
*                                                                      * A2714800
*'UPDTCORR-CARD'READING ROUTINE - CALLED  BY BAL LINKC,RDCRD            * A2714810
*                                                                      * A2714820
*     RETURN TO                                                        * A2714830
*  0(LINKC)-  DATA END-  ALL THE CARDS HAVE BEEN READ                  * A2714840
*  4(LINKC)-  FILE END- IS A / UPDATE CARD                             * A2714850
*  8(LINKC)-  CS   END- IS AN RIS CARD (CS IDENTIFICATION OTHER THAN   * A2714860
*                       THE PRECEDING ONE)                             * A2714870
```

```
* 12(LINKC)-NOT CS END- IS AN RIS CARD (SAME CS IDENTIFICATION AS IN  * A2714880
*                       PRECEDING RIS CARD)                           * A2714890
* 16(LINKC)-MODIF.CARD- IS A MODIFICATION CARD                        * A2714900
*                                                                     * A2714910
* PROCESSING-                                                        * A2714920
*     -THE READ CARD IS SAVED IN 'PRECEDING CARD AREA'(ONE AREA FOR   * A2714930
*      EACH / UPDATE,RIS OR MODIFICATION CARD TYPE) .                 * A2714940
*                                                                     * A2714950
*     -THE NEXT CARD IS READ .                                        * A2714960
*                                                                     * A2714970
*     -1ST CASE (DATA END).                                           * A2714980
*        -SET THE CORRESPONDING SWITCHES ONE .                        * A2714990
*                                                                     * A2715000
*     -2ND CASE (FILE END).                                           * A2715010
*        -SET THE CORRESPONDING SWITCHES ONE .                        * A2715020
*        -EXECUTE THE 'RSCAN' ROUTINE TO BREACK UP THE IDENTIFICATION* A2715030
*         PUNCHED IN THE READ / UPDATE CARD .                         * A2715040
*                                                                     * A2715050
*     -3RD CASE (    CS END) (RIS CARD)                               * A2715060
*     -4TH CASE (NOT CS END) (RIS CARD)                               * A2715070
*        -EXECUTE THE 'RSCAN' ROUTINE TO BREACK UP THE 2 IDENTIFICA- * A2715080
*       TIONS,THE INITIAL NUMBER FIELD AND THE NUMBERING STEP .       * A2715090
*        -CONVERT TO BINARY THE NUMERICAL PARTS OF THESE FIELDS (OR   * A2715100
*             REPLACE EACH BLANK FIELD WITH ZEROS) .                  * A2715110
*        -TEST THE PARAMETERS OF   THE READ RIS CARD AND SET THE      * A2715120
*             INVALIDITY SWITCH ON IF ANY ERROR IS DETECTED .         * A2715130
*        -ASSIGN   10 (MODULE PROCESSING MODE),OR 1(RECORD PROCESSING* A2715140
*             MODE) TO   THE BLANK FIELDS 'INITIAL NUMBER'AND 'STEP'.* A2715150
*        -STORE THE 'DUPLICATION'CODE                                 * A2715160
*             OF THE PRECEDING RIS CARD IN THE READ                   * A2715170
*             RIS CARD IF THIS CARD REQUESTS   'RECORD PROCESSING'    * A2715180
*             IN THE SAME MODULE AS THAT CONCERNED BY THE PRECEDING   * A2715190
*             RIS CARD .                                              * A2715200
*                                                                     * A2715210
*     -5TH CASE (MODIFICATION CARD)                                   * A2715220
*        -EXECUTE THE 'RSCAN' ROUTINE TO BREACK UP THE IDENTIFICATION* A2715230
*        -CONVERT TO BINARY THE NUMERICAL PART OF THIS FIELD (OR      * A2715240
*             REPLACE IT BY ZERO IF IT IS BLANK) .                    * A2715250
*        -SET THE INVALIDITY SWITCH IF ANY ERROR IS DETECTED IN THIS * A2715260
*             FIELD .                                                 * A2715270
*                                                                     * A2715280
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2715290
         EJECT                                                        A2715300
RDCRD    CLC   RLCOD(4),RISCOD          RIS CARD IN BUFFER           A2715310
         BC    8,RDCRDA                 YES                          A2715320
         CLC   GLCOD(9),SLASH           / UPDATE CARD IN BUFFER      A2715330
         BC    8,RDCRDE                 YES                          A2715340
         BAL   LINKE,PRCMOD             READ MODIF TO PRECEDING      A2715350
         BC    15,RDCRD1                                             A2715360
RDCRDA   BAL   LINKE,PRCRIS             READ RIS TO PRECEDING        A2715370
         BC    15,RDCRD1                                             A2715380
RDCRDE   BAL   LINKE,PRCPAR             READ / UPDATE TO PRECEDING   A2715390
RDCRD1   BAL   LINKF,ICARD              CARD READING OPERATION       A2715400
         OI    SWCSC1,X'08'             IS NOT 1ST RIS OF CS         A2715410
         CLC   RISBUF(1),TPMARK         IS CARD DATA END             A2715420
```

```
         BC    8,RDCRDF                  YES                          A2715430
         CLC   GLCOD(9),SLASH            IS CARD FILE END             A2715440
         BC    8,RDCRDB                  YES                          A2715450
         CLC   RLCOD(4),RISCOD           IS RIS CARD                  A2715460
         BC    8,RDCRDG                  YES                          A2715470
*      MODIF CARD IS READ                                             A2715480
         NI    MLVALD,X'FE'              IS VALID MODIF CARD          A2715490
         LA    WORK,MLIDT                FROM MLIDT,STORE             A2715500
         LA    WORKA,MLCSD               *CS  PART IN MLCSD           A2715510
         LA    WORKB,MLNUM               *NUM PART IN MLNUM           A2715520
         LA    WORKC,MLCSL               *CS LENGTH IN MLCSL          A2715530
         BAL   LINKD,RSCAN               *                            A2715540
         LA    POINTR,MLNUM              CONVERT TO BINARY THE NUMB.  A2715550
         LA    CONTR,NUM(0,0)            * ER OF  MODIF CARD- ZEROS   A2715560
         BAL   LINKE,HEXBA               *IF NUMBER IS BLANK          A2715570
         BC    15,RDCRDC                 IS NOT NUMERICAL             A2715580
         ST    CONTR,MLBIN               STORE BINARY NUMBER          A2715590
         CLC   MLBIN(4),MPBIN            IS INCREASING NUMBER         A2715600
         BC    2,RDCRDD                  YES-BRANCH                   A2715610
         CLC   MLNBR(4),ZERO             IS 1ST MODIF AFTER RIS CARD  A2715620
         BC    8,RDCRDD                  YES                          A2715630
RDCRDC   OI    MLVALD,X'01'              IS INVALID MODIF NUMBER      A2715640
RDCRDD   L     WORKA,MLNBR               UP TO DATE THE MODIFS CARDS  A2715650
         LA    WORKA,1(WORKA)            *COUNT                       A2715660
         ST    WORKA,MLNBR               *                            A2715670
         BC    15,16(LINKC)              MODIF CARD -CALLING+16       A2715680
* / UPDATE CARD  IS READ (CARD FILE END)                               A2715690
RDCRDB   OI    SWFCE,X'02'               CARD FILE END                A2715700
         OI    SWCSCE,X'02'              CARD CS END                  A2715710
         MVI   GLFILE,X'E2'            * ALWAYS SYMBOLIC FILE OPTION  A2715720
         LA    WORK,GLIDT                STORE CS PART OF GLIDT       A2715730
         LA    WORKA,GLCSD               * IN GLCSD                   A2715740
         LA    WORKB,WKAREA              * AND  HIS LENGTH IN GLCSL   A2715750
         LA    WORKC,GLCSL               *                            A2715760
         BAL   LINKD,RSCAN               *                            A2715770
         BC    15,4(LINKC)               CARD FILE END-CALLING+4      A2715780
*   TAPE MARK (7F)IS READ (DATA FILE END)                              A2715790
RDCRDF   OI    SWDCE,X'02'               CARD DATA END                A2715800
         OI    SWFCE,X'02'               CARD FILE END                A2715810
         OI    SWCSCE,X'02'              CARD CS   END                A2715820
         BCR   15,LINKC                  CARD DATA END-CALLING+0      A2715830
*   RIS CARD IS READ     ( CS END OR NOT)                              A2715840
RDCRDG   NI    RLVALD,X'FE'              IS VALID RIS                 A2715850
         LA    WORK,RLIDT1               FROM RLIDT1,STORE            A2715860
         LA    WORKA,RLCSD1              *CS  PART IN RLCSD1          A2715870
         LA    WORKB,RLNUM1              *NUM PART IN RLNUM1          A2715880
         LA    WORKC,RLCS1L              *CS LENGTH IN RLCS1L         A2715890
         BAL   LINKD,RSCAN               *                            A2715900
         LA    WORK,RLIDT2               FROM RLIDT2,STORE            A2715910
         LA    WORKA,RLCSD2              *CS PART IN RLCSD2           A2715920
         LA    WORKB,RLNUM2              *NUM PART IN RLNUM2          A2715930
         LA    WORKC,RLCS2L              *CS LENGTH IN RLCS1L         A2715940
         BAL   LINKD,RSCAN               *                            A2715950
         LA    WORK,RLNUM                FROM RLNUM(INITIAL NUMB)ST.  A2715960
         LA    WORKA,RLCSDI              *CS PART IN RLCSDI           A2715970
```

```
         LA    WORKB,RLNUMI          *NUM PART IN RLNUMI          A2715980
         LA    WORKC,RLCSNL          *          CS LENGTH         A2715990
         BAL   LINKD,RSCAN           *            IN RLCSNL       A2716000
         LA    WORK,RLSTPZ           FROM RLSTPZ(STEP),STORE     A2716010
         LA    WORKA,WKAREA          *                           A2716020
         LA    WORKB,RLSTP           *NUM PART IN RLSTP          A2716030
         LA    WORKC,RLCSSL          *          CS LENGTH        A2716040
         BAL   LINKD,RSCAN           *            IN RLCSSL       A2716050
         LA    POINTR,RLNUM1         CONVERT TO BINARY NUM1 OF   A2716060
         LA    CONTR,NUM(0,0)        *RIS CARD-(ZERO IF NUMBER   A2716070
         BAL   LINKE,HEXBA           *IS BLANK)                  A2716080
         BC    15,RDCRDH             IS NOT NUMERICAL            A2716090
         BC    15,RDCRDI             IS NUMERICAL                A2716100
RDCRDH   OI    RLVALD,X'01'          INVALID RIS                 A2716110
RDCRDI   ST    CONTR,RLBIN1          STORE BINARY NUMBER1        A2716120
         LA    POINTR,RLNUM2         CONVERT TO BINARY NUM2 OF   A2716130
         LA    CONTR,NUM(0,0)        *RIS CARD-(ZERO IF NUMBER   A2716140
         BAL   LINKE,HEXBA           *IS BLANK)                  A2716150
         BC    15,RDCRDJ             IS NOT NUMERICAL            A2716160
         BC    15,RDCRDK             IS NUMERICAL                A2716170
RDCRDJ   OI    RLVALD,X'01'          INVALID RIS                 A2716180
RDCRDK   ST    CONTR,RLBIN2          STORE BINARY NUMBER2        A2716190
         CLI   RLMOD,X'C3'           CORRECTION MODE IS PER CS   A2716200
         BC    8,RDCRDN              YES-BRANCH                  A2716210
         TM    FILESW,X'02'          IS SYMBOLIC FILE OPTION     A2716220
         BC    1,RDCRDN              YES-BRANCH                  A2716230
         OI    RLVALD,X'01'          INVALID RIS CARD            A2716240
RDCRDN   LA    POINTR,RLSTP          CONVERT TO BINARY'STEP' OF  A2716250
         LA    CONTR,NUM(0,0)        *RIS CARD(ZERO IF STEP IS   A2716260
         BAL   LINKE,HEXBA           *BLANK)                     A2716270
         BC    15,RDCRDP             IS NOT NUMERICAL            A2716280
         BC    15,RDCRDQ             IS NUMERICAL(OR BLANK)      A2716290
RDCRDP   OI    RLVALD,X'01'          INVALID RIS                 A2716300
RDCRDQ   ST    CONTR,RLSTEP          STORE BINARY STEP           A2716310
         LA    POINTR,RLNUMI         CONVERT TO BINARY THE       A2716320
         LA    CONTR,NUM(0,0)        *INITIAL NUMBER (ZERO IF IT A2716330
         BAL   LINKE,HEXBA           *IS BLANK)                  A2716340
         BC    15,RDCRDR             IS NOT NUMERICAL            A2716350
         BC    15,RDCRDS             IS NUMERICAL (OR BLANK)     A2716360
RDCRDR   OI    RLVALD,X'01'          INVALID RIS                 A2716370
RDCRDS   ST    CONTR,RLBINI          STORE BINARY INITIAL NUMBER A2716380
         SR    WORK,WORK             CS1 LENGTH TO WORK          A2716390
         IC    WORK,RLCS1L           *                           A2716400
         SR    WORKA,WORKA           CS2 LENGTH TO WORKA         A2716410
         IC    WORKA,RLCS2L          *                           A2716420
         CLI   RLCNT,X'C9'           INSERT REQUEST              A2716430
         BC    7,RDCRDX              NO                          A2716440
         CH    WORK,KB8              CS1 LENGTH=8                A2716450
         BC    8,RDCRDV              YES-ERROR                   A2716460
         BC    15,RDCRDY             YES-CORRECT                 A2716470
RDCRDX   CLI   RLMOD,X'C3'           CORRECTION MODE PER CS      A2716480
         BC    8,RDCRDT              YES                         A2716490
         CLC   RLCSD1(CS),RLCSD2     CS1 EQUAL CS2               A2716500
         BC    7,RDCRDV              NO -ERROR                   A2716510
         CLC   RLBIN1(4),RLBIN2      BIN1     GREATER  THAN BIN2 A2716520
```

```
         BC    2,RDCRDV                  YES-ERROR                      A2716530
         CH    WORKA,KB8                 CS2 LENGTH = 8                 A2716540
         BC    8,RDCRDV                  YES-ERROR                      A2716550
         LTR   WORKA,WORKA               CS2 LENGTH = 0                 A2716560
         BC    8,RDCRDV                  YES-ERROR                      A2716570
RDCRDT   CH    WORK,KB8                  CS1 LENGTH = 8                 A2716580
         BC    8,RDCRDV                  YES-ERROR                      A2716590
         LTR   WORK,WORK                 CS1 LENGTH = 0                 A2716600
         BC    8,RDCRDV                  YES-ERROR                      A2716610
         CLI   RLCNT,X'E2'               DELETING REQUESTED             A2716620
         BC    8,RDCRDW                  YES                            A2716630
RDCRDY   CLC   RLSTEP(4),ZERO            THE BINARY STEP IS NULL        A2716640
         BC    7,RDCRDU                  NO-                            A2716650
         XC    RLSTEP(4),RLSTEP          REPLACE ZERO,IN BINARY STEP    A2716660
         MVI   RLSTEP+3,X'0A'            * BY 10                        A2716670
         CLI   RLMOD,X'C3'               * (MODE PER CS)                A2716680
         BC    8,RDCRDU                  * BY 1                         A2716690
         MVI   RLSTEP+3,X'01'            *(MODE PER RECORD)             A2716700
RDCRDU   CLC   RLCSNL(1),KB8+1           CS PART LENGTH OF NEW CS =8    A2716710
         BC    8,RDCRDV                  YES -ERROR                     A2716720
         CLC   RLNUMI(NUM),BLANK         IF INITIAL NUMBER IS BLANK     A2716730
         BC    7,RDCRDZ                  *REPLACE 0 IN THIS BIN.VAL     A2716740
         XC    RLBINI(4),RLBINI          * BY 10                        A2716750
         MVI   RLBINI+3,X'0A'            *(MODE PER CS)                 A2716760
         CLI   RLMOD,X'C3'               * BY 1                         A2716770
         BC    8,RDCRDZ                  *(MODE PER RECORD)             A2716780
         MVI   RLBINI+3,X'01'            *                              A2716790
RDCRDZ   CLI   RLCNT,X'C9'               INSERT REQUEST                 A2716800
         BC    8,RDCRDW                  YES                            A2716810
         CLI   RLCNT,X'D5'               NUMBERING REQUEST              A2716820
         BC    8,RDCRDW                  YES                            A2716830
         CLI   RLCNT,X'D9'               SUPPRESS REQUEST               A2716840
         BC    8,RDCRDW                  YES                            A2716850
RDCRDV   OI    RLVALD,X'01'              IS INVALID RIS CARD            A2716860
RDCRDW   XC    MLNBR(4),MLNBR            ANNUL THE MOD. COUNT BY SET    A2716870
         XC    MPBIN(4),MPBIN            ANNUL THE PRECDT.MODIF NUMB    A2716880
         CLC   RLCSD1(CS),RPCSD1         READ CS # PRECEDING CS         A2716890
         BC    7,RDCRDO                  NO-BRANCH                      A2716900
         CLI   RLMOD,X'C3'               CORRECTION MODE PER CS         A2716910
         BC    8,12(LINKC)               YES-NOT CS END-CALLER+12       A2716920
         MVC   RLDUPL(1),RPDUPL          SAVE DUPLICATING CODE          A2716930
         BC    15,12(LINKC)              NOT CS END(RIS)-CALLER+12      A2716940
RDCRDO   OI    SWCSCE,X'02'              CARD CS END                    A2716950
         BC    15,8(LINKC)               CARD CS END(RIS).CALLER+8      A2716960
         EJECT                                                          A2716970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2716980
*                                                                       * A2716990
* OLD RECORD READING ROUTINE - CALLED  BY  BAL LINKC,RDOLD              * A2717000
*                                                                       * A2717010
*     RETURN TO                                                         * A2717020
*                                                                       * A2717030
*  0(LINKC)-  DATA END-IS THE LAST TAPE MARK  OF  UPDTOLD             * A2717040
*  4(LINKC)-  FILE END-IS A TAPE MARK AT THE END OF A FILE            * A2717050
*  8(LINKC)-    CS END-IS A RECORD(CS IDENT.OTHER THAN THE PRECEDING) * A2717060
* 12(LINKC)-NOT CS END-IS A RECORD(SAME CS IDENTIFIC.AS IN PRECEDING) * A2717070
```

```
*                                                                     * A2717080
* PROCESSING                                                          * A2717090
*     - SAVE THE READ OLD RECORD IN 'PRECEDING RECORD'AREA .          * A2717100
*                                                                     * A2717110
*     - READ THE NEXT RECORD .                                        * A2717120
*                                                                     * A2717130
*     - 1ST CASE (DATA END)                                           * A2717140
*     - 2ND CASE (FILE END)                                           * A2717150
*        - SET APPROPRIATE SWITCHES ON .                              * A2717160
*                                                                     * A2717170
*     - 3RD CASE (CS END)                                             * A2717180
*     - 4TH CASE (NOT CS END)                                         * A2717190
*        -EXECUTE THE 'RSCAN' ROUTINE TO BREACK UP THE IDENTIFICATION* A2717200
*        - CONVERT TO BINARY ITS NUMERICAL PART .                     * A2717210
*        - SET THE INVALIDITY SWITCH IF ANY ERROR IS DETECTED .       * A2717220
*                                                                     * A2717230
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2717240
RDOLD    TM    SWOLD,X'01'              WAS DATA OR FILE END          A2717250
         BC    1,RDOLDB                 YES                           A2717260
         BAL   LINKE,PRCOLD             READ OLD RECORD TO PRECED.    A2717270
RDOLDB   BAL   LINKF,IRECD              OLD RECORD READING OPERAT.    A2717280
         OI    SWCSO1,X'04'             1ST OLD REC.TREATED IN CS     A2717290
         OI    SWFO1,X'04'              1ST OLD REC.TREATED IN FILE   A2717300
         NI    OLVALD,X'00'             VALID NUMBER IN READ RECORD   A2717310
         TM    SWOLD,X'03'              NEITHER DATA NOR FILE END     A2717320
         BC    8,RDOLDD                 * BRANCH                      A2717330
         OI    SWFOE,X'01'              OLD FILE END                  A2717340
         OI    SWCSOE,X'01'             OLD CS   END                  A2717350
         XC    OLIDT(8),OLIDT           0 TO ID ZONE OF READ RECORD   A2717360
         MVC   OLCSD(CS),SLASH                                        A2717370
         TM    SWOLD,X'02'              IS OLD DATA END               A2717380
         BC    1,RDOLDA                 YES                           A2717390
         BC    15,4(LINKC)              OLD FILE END(7F)1-CALLER+4    A2717400
RDOLDA   OI    SWDOE,X'01'              OLD DATA END (7F)2            A2717410
         BCR   15,LINKC                 TO CALLER                     A2717420
RDOLDD   LA    WORK,OLIDT               FROM OLIDT,STORE              A2717430
         LA    WORKA,OLCSD              *CS PART IN OLCSD             A2717440
         LA    WORKB,OLNUM              *NUM PART IN OLNUM            A2717450
         LA    WORKC,OLCSL              *CS LENGTH IN OLCSL           A2717460
         BAL   LINKD,RSCAN              *                             A2717470
         SR    CONTR,CONTR                                            A2717480
         CLI   OLCSL,X'00'              CS LENGTH IS NULL             A2717490
         BC    8,RDOLDE                 YES-ERROR                     A2717500
         CLC   OLCSL(1),KB8+1           CS LENGTH = 8                 A2717510
         BC    8,RDOLDE                 YES-ERROR                     A2717520
         LA    POINTR,OLNUM             CONVERT TO BINARY-NUMBER OF   A2717530
         LA    CONTR,NUM(0,0)           *OLD RECORD                   A2717540
         BAL   LINKE,HEXBB              *                             A2717550
         BC    15,RDOLDE                IS NOT NUMERICAL              A2717560
         ST    CONTR,OLBIN              STORE BINARY NUMBER           A2717570
         CLC   OLCSD(CS),OPCSD          READ CS # PRECEDING CS        A2717580
         BC    7,RDOLDH                 NO-BRANCH                     A2717590
         CLC   OLBIN(4),OPBIN           INCREASING OLD NUMBER IN CS   A2717600
         BC    2,12(LINKC)              YES-NOT CS END   CALLER+12    A2717610
RDOLDF   OI    OLVALD,X'01'             IS INVALID NUMBER             A2717620
```

```
         CLC   OLCSD(CS),OPCSD              SAME CS IN READ RECORD AS   A2717630
         BC    8,12(LINKC)                  *IN PRECEDING               A2717640
RDOLDH   OI    SWCSOE,X'01'                 OLD CS END                  A2717650
         BC    15,8(LINKC)                  TO CALLER+8                 A2717660
RDOLDE   ST    CONTR,OLBIN                  STORE ZERO TO BINARY NUMBER A2717670
         MVC   OLNUM(NUM),ZEROF                                         A2717680
         BC    15,RDOLDF                    *ERROR                      A2717690
         EJECT                                                          A2717700
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2717710
*                                                                       A2717720
* NEW RECORD EDITING ROUTINE  - CALLED  BY BAL LINKC,ENRXX              A2717730
*                                                                       A2717740
*  ENTRY AT ENRMA - NEW RECORD CREATED BY  A MODIFICATION CARD TO BE    A2717750
*                   INSERTED (REPLACEMENT OR INSERTION) .               A2717760
*  ENTRY AT ENROA - NEW RECORD CREATED BY  AN OLD RECORD TO BE COPIED   A2717770
*  ENTRY AT ENRO1 - NEW RECORD CREATED BY  THE 1ST OLD RECORD OF A SET  A2717780
*                   TO BE NUMBERED .                                    A2717790
*  ENTRY AT ENRON - NEW RECORD CREATED BY  ANY OLD RECORD OF A SET      A2717800
*                   TO BE NUMBERED .                                    A2717810
*                                                                       A2717820
*  RETURN TO CALLER  BY BCR 15,LINKC .                                  A2717830
*                                                                       A2717840
*  PROCESSING                                                           A2717850
* 1-   DETERMINING      THE BINARY NEW NUMBER TO BE WRITTEN             A2717860
* ************************************************************** *     A2717870
* *     *IS THE *IS THE * THE NEW NUMBER*THIS NUMBER* THE NEW NUMBER* *  A2717880
* *     *1ST NEW*1ST NEW*     SHOULD     * IS GREATER*    WILL  BE    * *  A2717890
* *     *RECORD *RECORD *       BE       *  THAN THE *                * *  A2717900
* *     *OF THE *ISSUED *                *NUMBER OF  *                * *  A2717910
* *     *MODULE * BY  A *                *    THE    *                * *  A2717920
* *     *       *CORREC-*                * PRECEDING *                * *  A2717930
* *     *       *TION   *                *NEW RECORD *                * *  A2717940
* ************************************************************** *     A2717950
* *     *       *       *                *           *                * *  A2717960
* *ENRMA* YES   * (YES) *                *           *INITIAL NUMBER * *  A2717970
* *     ********************************************    FROM        * *  A2717980
* *     * NO    *  YES  * INITIAL NUMBER*   YES     *  RIS  CARD    * *  A2717990
* *     *       *       *                *           *               * *  A2718000
* *     *       *       * FROM RIS CARD **************************** * *  A2718010
* *     *       *       *                *   NO      * PRECEDING     * *  A2718020
* *     *       *       *                *           *NEW NUMBER + 1 * *  A2718030
* *     *       ********************************************************  A2718040
* *     *       *  NO   *                *           *1ST MULTIPLE OF* *  A2718050
* ************************************************STEP GREATER   * *  A2718060
* *ENRON* (NO)  * (NO)  *                *           *THAN THE PRECE-* *  A2718070
* *     *       *       *                *           *DING NEW NUMBER* *  A2718080
* ************************************************************** *     A2718090
* *ENRO1* YES   * (YES) *                *           *INITIAL NUMBER * *  A2718100
* *     ********************************************    FROM        * *  A2718110
* *     * NO    * (YES) * INITIAL NUMBER*   YES     *  RIS  CARD    * *  A2718120
* *     *       *       * FROM RIS CARD **************************** * *  A2718130
* *     *       *       *                *   NO      * PRECEDING     * *  A2718140
* *     *       *       *                *           *NEW NUMBER + 1 * *  A2718150
* ************************************************************** *     A2718160
* *ENROA* YES   *       *                *           *    NUMBER     * *  A2718170
```

```
* *     ******************************************     FROM      * * A2718180
* *     * NO    *       *NUMBER FROM    *   YES     *  OLD RECORD   * * A2718190
* *     *       *       *  OLD RECORD   ***************************** * A2718200
* *     *       *       *               *   NO      * PRECEDING     * * A2718210
* *     *       *       *               *           *NEW NUMBER + 1 * * A2718220
* ************************************************************************ * A2718230
*                                                                          * A2718240
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2718250
        EJECT                                                               A2718260
* * * * * * * * * * * *  CONTINUED  * * * * * * * * * * * * * * * * * * * * A2718270
*                                                                          * A2718280
*-2-   ENRMOD- COMPUTE THE MAXIMUM VALUE OF A NEW NUMBER .                * A2718290
*                                                                          * A2718300
*-3-   ENRMOJ- CHECK   POSSIBLE OVERFLOW ON THE PROPOSED NEW NUMBER       * A2718310
* ************************************************************************ * A2718320
* * IF THE PROPOSED NEW * IF THE NUMBER OF    * THEN THE FINAL      * * A2718330
* *     NUMBER IS       * THE PRECEDING NEW   * VALUE OF THE NEW    * * A2718340
* *                     *    RECORD +1  IS    *   NUMBER WILL BE    * * A2718350
* ************************************************************************ * A2718360
* * LESS THAN THE       *                      * THAT OF THE PROPOSED* * A2718370
* *  MAXIMUM VALUE      *                      *     NEW NUMBER      * * A2718380
* ************************************************************************ * A2718390
* * EQUAL TO  THE       *                      * THAT OF THE         * * A2718400
* *  MAXIMUM VALUE      *                      * PRECEDING NEW       * * A2718410
* ****************************************** RECORD   + 1          * * A2718420
* * GREATER THAN THE    * NOT GREATER THAN    *                     * * A2718430
* *  MAXIMUM VALUE      * THE MAXIMUM VALUE   *                     * * A2718440
* *                     ***************************************** * A2718450
* *                     * GREATER THAN        * SET TO  ONE         * * A2718460
* *                     * THE MAXIMUM VALUE   * (OVERFLOW)          * * A2718470
* ************************************************************************ * A2718480
*                                                                          * A2718490
*-4-   ENRMOM - MOVE  TO THE NEW RECORD BUFFER #                          * A2718500
*       -THE TEXT (COL 1 THROUGH 72) FROM THE MODIFICATION CARD OR        * A2718510
*           FROM THE OLD RECORD,                                          * A2718520
*       -THE MODULE IDENTIFICATION FROM THE MODIFICATION CARD OR FROM     * A2718530
*            THE OLD RECORD OR FROM THE RIS CARD (IF THIS RIS CARD        * A2718540
*            REQUESTS   NUMBERING WITH CHANGE IN AN IDENTIFICATION) .     * A2718550
*                                                                          * A2718560
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2718570
ENRMA    OI    SWENR,X'01'                FROM MODIF CARD                 A2718580
         MVC   NWBIN+4(4),NWBIN           SAVE PRECD.BINARY NEW NUMB.     A2718590
         CLC   MLCSD(CS),NWCSD            1ST NEW RECORD                  A2718600
         BC    7,ENRME                    YES                             A2718610
         CLC   MLNBR(4),K1                1ST MODIF CARD AFTER RIS        A2718620
         BC    8,ENRMB                    YES-BRANCH                      A2718630
         L     WORKC,RPSTEP               LOAD STEP FROM PRECEDT.RIS      A2718640
         BC    15,ENRMOB                  TO NUMBERING (STEP USED)        A2718650
ENRMB    CLC   RPBINI(4),NWBIN            INITIAL NUMBER IS CORRECT       A2718660
         BC    2,ENRMD                    YES-BRANCH                      A2718670
ENRMB1   MVC   ERTBG+12(COD),MSGTBN       NO-SET ERROR CODE TO ERTBG      A2718680
         MVC   ERCOD+12(COD),MSGTBN       *AND TO PRINT BUFFER            A2718690
         L     WORKC,NWBIN                NEW NUMBER # PRECEDING+1        A2718700
         LA    WORKC,1(WORKC)             *                               A2718710
         ST    WORKC,NWBIN                *                               A2718720
```

```
         BC    15,ENRMOD                                                A2718730
ENRME    MVC   NWBIN+4(4),ZERO     0 TO PRECEDING BIN.NEW NUMB          A2718740
ENRMD    MVC   NWBIN(4),RPBINI     INIT.NUMB.FROM PRCDING RIS           A2718750
         BC    15,ENRMOD           *BECOMES NEW NUMBER-BRANCH           A2718760
ENRON    NI    SWENR,X'FE'         FROM OLD RECORD                      A2718770
         MVC   NWBIN+4(4),NWBIN    SAVE PRECED.BINARY NEW NUMB          A2718780
         L     WORKC,RLSTEP        LOAD STEP FROM READ RIS              A2718790
         BC    15,ENRMOB           TO NUMBERING (STEP USED)             A2718800
ENRO1    NI    SWENR,X'FE'         FROM OLD RECORD                      A2718810
         MVC   NWBIN+4(4),NWBIN    SAVE PRECD.BINARY NEW NUMB.          A2718820
         CLC   OLCSD(CS),NWCSD     1ST OLD RECORD IN NEW CS             A2718830
         BC    7,ENRO3             YES                                  A2718840
         CLC   RLBINI(4),NWBIN     INITIAL NUMBER IS CORRECT            A2718850
         BC    2,ENRO2             YES-BRANCH                           A2718860
         BC    15,ENRMB1           NO-ERROR                             A2718870
ENRO3    MVC   NWBIN+4(4),ZERO     0 TO PRECD.BIN.NEW NUMBER            A2718880
ENRO2    MVC   NWBIN(4),RLBINI     INIT.NUMB.FROM READ RIS              A2718890
         BC    15,ENRMOD                                                A2718900
ENROA    NI    SWENR,X'FE'         FROM OLD RECORD                      A2718910
         MVC   NWBIN+4(4),NWBIN    SAVE PRECD.BIN.NEW NUMBER            A2718920
         NI    SWIDT,X'FE'         OLD CS ID NON CHANGED                A2718930
         CLC   OLCSD(CS),NWCSD     1ST OLD RECORD IN NEW CS             A2718940
         BC    7,ENROD             YES                                  A2718950
         CLC   OLBIN(4),NWBIN      CORRECT OLD NUMBER TO BE             A2718960
         BC    12,ENRMB1           *COPIED-NO                           A2718970
         BC    15,ENROB                                                 A2718980
ENROD    MVC   NWBIN+4(4),ZERO     0 TO PRECD.BIN.NEW NUMBER            A2718990
ENROB    MVC   NWBIN(4),OLBIN      OLD NUMBER TO NEW NUMBER             A2719000
         BC    15,ENRMOD                                                A2719010
ENRMOB   L     WORKB,NWBIN         STORE THE PRECEDING BINARY           A2719020
         L     WORK,NWBIN          *NEW NUMBER+1                        A2719030
         LA    WORKB,1(WORKB)      *TO THE BINARY NEW NUMBER            A2719040
         ST    WORKB,NWBIN         *TO BE WRITTEN                       A2719050
         SR    WORKA,WORKA         NEW BIN DIVIDED BY STEP              A2719060
         DR    WORKA,WORKC         *                                    A2719070
         L     WORKB,NWBIN         LOAD NEW BIN                         A2719080
         LTR   WORKA,WORKA         THE NEW NUMBER IS A                  A2719090
         BC    8,ENRMOF            *MULTIPLE OF THE STEP                A2719100
         AR    WORKB,WORKC         NEW NUMBER + STEP                    A2719110
         SR    WORKB,WORKA         * - REMAINDER OF DIVISION            A2719120
ENRMOF   ST    WORKB,NWBIN         STORE THIS VALUE                     A2719130
ENRMOD   SR    WORK,WORK                                                A2719140
         MVC   MXNUM1(NUM),K999    99999999 TO HIGH VALUE               A2719150
         IC    WORK,OLCSL          CSID.LENGTH OF OLD RECORD            A2719160
         TM    SWIDT,X'01'         CHANGE CS ID REQUESTED               A2719170
         BC    8,ENRMOE            NO                                   A2719180
         IC    WORK,RPCSNL         CSID.LENGTH FROM PRECED.RIS          A2719190
ENRMOE   TM    SWENR,X'01'         IS TREATMENT BY OLD RECORD           A2719200
         BC    8,ENRMOH            YES                                  A2719210
         IC    WORK,MLCSL          CSID.LENGTH OF OLD RECORD            A2719220
ENRMOH   STC   WORK,NWCSL          CSID.LENGTH OF NEW RECORD            A2719230
         BCTR  WORK,0              CSID LENGTH  -1 TO MVC               A2719240
         STC   WORK,ENRMOP+1       *                                    A2719250
         STC   WORK,ENRMOI+1       *                                    A2719260
ENRMOI   MVC   MXNUM1(1),ZEROF     ZEROS TO LEFT PART OF H.VAL          A2719270
```

```
         LA    POINTR,MXNUM1             CONVERT TO BINARY THE HIGH  A2719280
         LA    CONTR,NUM(0,0)            * VALUE OF NEW NUMBER       A2719290
         BAL   LINKE,HEXBA               *                           A2719300
         BC    15,ENRMOJ                 *                           A2719310
ENRMOJ   ST    CONTR,MXBIN1              * AND STORE THE BINARY VAL. A2719320
         L     WORK,NWBIN+4              INCREMENT  THE PRECEDING    A2719330
         A     WORK,K1                   * BINARY NEW NUMBER BY 1    A2719340
         ST    WORK,NWBIN+4              *                           A2719350
         CLC   NWBIN(4),MXBIN1           THE BINARY NEW NUMBER IS    A2719360
         BC    4,ENRMOH                  *LESS THAN BINARY HIGH VAL. A2719370
         BC    8,ENRMOL                  * EQUAL                     A2719380
         CLC   NWBIN+4(4),MXBIN1         THE PRECED. BIN.NEW NUMB.IS A2719390
         BC    2,ENRMOK                  *GREATER  THAN THE HIGH.VAL A2719400
ENRMOL   MVC   NWBIN(4),NWBIN+4          PRECEDING NEW NUMBER +1 TO  A2719410
         BC    15,ENRMOM                 * BINARY NEW NUMBER         A2719420
ENRMOK   MVC   ERTBG+14(COD),MSGTBM      ERROR CODE(FOR OVERFLOW)TO  A2719430
         MVC   ERCOD+14(COD),MSGTBM      *ERTBG AND TO PRINT BUFFER  A2719440
         MVC   NWBIN(4),K1               1 TO BINARY NEW NUMBER      A2719450
ENRMOM   LA    POINTR,NWNUM              CONVERT THE BINARY NEW      A2719460
         LA    CONTR,NUM(0,0)            *NUMBER TO 'ZONED'AND STORE A2719470
         L     WORK,NWBIN                *IN 0(POINTR)               A2719480
         BAL   LINKE,BINHXA              *                           A2719490
         TM    SWENR,X'01'               OLD RECORD TREATMENT        A2719500
         BC    8,ENRMON                  YES-BRANCH                  A2719510
         MVC   NWCOD(1),MLCOD            MOVE ,FROM READ MODIF CARD  A2719520
         MVC   NWTXT(71),MLTXT           *TO NEW RECORD-             A2719530
         MVC   NWIDT(NUM),NWNUM          * CODE AND TEXT             A2719540
         MVC   NWCSD(CS),MLCSD           * NUMBER AND CS             A2719550
ENRMOP   MVC   NWIDT(1),NWCSD            *(LENGTH OF CS = CSL)       A2719560
         BCR   15,LINKC                  RETURN TO CALLER            A2719570
ENRMON   MVC   NWCOD(1),OLCOD            FROM READ OLD RECORD,TO NEW A2719580
         MVC   NWTXT(71),OLTXT           *RECORD-CODE AND TEXT       A2719590
         MVC   NWIDT(NUM),NWNUM          *AND NEW NUMBER             A2719600
         TM    SWIDT,X'01'               CHANGE CSID REQUESTED       A2719610
         BC    1,ENRMOQ                  YES                         A2719620
         MVC   NWCSD(CS),OLCSD           NEW CSID FROM OLD RECORD    A2719630
         BC    15,ENRMOP                 *                           A2719640
ENRMOQ   MVC   NWCSD(CS),RPCSDI          NEW CSID FROM PRECEDING RIS A2719650
         BC    15,ENRMOP                 *                           A2719660
         EJECT                                                       A2719670
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2719680
*                                                                     A2719690
* CONVERT A ZONED FIELD TO BINARY - CALLED  BY                        A2719700
*  LA POINTR,ZONED       ZONED FIELD ADDRESS                          A2719710
*  LA CONTR,LENGTH(0,0)  ZONED BYTES NUMBER  (WILL CONTAIN THE RESULT)A2719720
* BAL LINKE,HEXBX                                                    A2719730
*                                                                     A2719740
* ENTRY AT HEXBA - SET THE RESULT TO ZERO IF THE FIELD IS ALL BLANK,  A2719750
*                - OTHERWISE CONVERT THIS FIELD TO BINARY.            A2719760
* ENTRY AT HEXBB - CONVERT THE FIELD TO BINARY .                      A2719770
*                                                                     A2719780
* CONVERSION- ALL THE BYTES OF THE FIELD MUST BE NUMERICAL .OTHERWISE A2719790
*             THE FUNCTION IS INTERRUPTED AND THE RESULT IS SET TO    A2719800
*             ZERO.                                                   A2719810
*           - THE BINARY RESULT IS LOADED IN CONTR .                  A2719820
```

```
*                                                                     * A2719830
* EXCEPTIONAL RETURN TO  0(LINKE)                                     * A2719840
* NORMAL RETURN TO 4(LINKE)                                           * A2719850
*                                                                     * A2719860
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2719870
HEXBA    LR    WORKA,CONTR              IF ZONED FIELD IS BLANK      A2719880
         BCTR  WORKA,0                  * ANNUL THE BINARY RESULT    A2719890
         STC   WORKA,HEXBA1+1           * AND BRANCH TO NORMAL RET.  A2719900
HEXBA1   CLC   BLANK(4),0(POINTR)        *   OTHERWISE                A2719910
         BC    7,HEXBB                  * BRANCH TO CONVERT.         A2719920
         SR    CONTR,CONTR              *                            A2719930
         BC    15,4(LINKE)              *                            A2719940
HEXBB    LR    WORK,POINTR              SAVE  ZONED FIELD ADDRESS    A2719950
         LR    WORKA,CONTR              SAVE ZONED FIELD LENGTH      A2719960
         BC    15,HEXBD                                              A2719970
HEXBC    LA    WORK,1(WORK)             NEXT ZONED BYTE ADDRESS      A2719980
HEXBD    CLI   0(WORK),X'F0'            IS  NUMERICAL BYTE           A2719990
         BC    4,HEXBE                  NO- ERROR                    A2720000
         CLI   0(WORK),X'F9'            IS NUMERICAL BYTE            A2720010
         BC    2,HEXBE                  NO-ERROR                     A2720020
         BCT   WORKA,HEXBC              TO NEXT CHARACTER            A2720030
         NI    0(WORK),X'CF'            '+' TO LAST ZONED BYTE       A2720040
         BCTR  CONTR,0                  LENGTH-1                     A2720050
         STC   CONTR,HEXBD1+1           STORE LENGTHS OF OPERANDS    A2720060
         OI    HEXBD1+1,X'70'           *OF NEXT PACK                A2720070
HEXBD1   PACK  WKAREA(8),0(4,POINTR)     CONVERT                      A2720080
         CVB   CONTR,WKAREA             *TO BINARY                   A2720090
         OI    0(WORK),X'30'                                         A2720100
         BC    15,4(LINKE)              NORMAL RETURN TO CALLER      A2720110
HEXBE    SR    CONTR,CONTR              ZERO TO INVALID RESULT       A2720120
         BCR   15,LINKE                 EXCEPT.RETURN TO CALLER      A2720130
         EJECT                                                       A2720140
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2720150
*                                                                     * A2720160
* CONVERT A BINARY FIELD TO ZONED -   CALLED  BY                     * A2720170
*  LA POINTR,ZONED         RESULT FIELD ADDRESS                       * A2720180
*  LA CONTR,LENGTH(0,0)    RESULT BYTES NUMBER                        * A2720190
*  L  WORK,BINARY          BINARY WORD TO BE CONVERTED                * A2720200
* BAL LINKE,BINHXA                                                    * A2720210
*                                                                     * A2720220
* RETURN TO 0(LINKE)                                                  * A2720230
*                                                                     * A2720240
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2720250
BINHXA   CVD   WORK,WKAREA              CONVERT TO DECIMAL           A2720260
         BCTR  CONTR,0                  ZONED FIELD LENGTH - 1       A2720270
         LR    WORKB,CONTR                                           A2720280
         SLA   CONTR,4                  THE LENGTH IN UNPK           A2720290
         LA    CONTR,7(CONTR)           * IS  X'X8'                  A2720300
         STC   CONTR,BINHXB+1           *                            A2720310
BINHXB   UNPK  0(4,POINTR),WKAREA       CONVERT DECIMAL TO ZONED     A2720320
         AR    POINTR,WORKB             LAST BYTE ADDRESS            A2720330
         OI    0(POINTR),X'F0'          SUPPRESS SIGN IN LAST BYTE   A2720340
         BCR   15,LINKE                                              A2720350
         EJECT                                                       A2720360
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2720370
```

```
*                                                                     * A2720380
* SAVE  A  READ DATA TO APPROPRIATE AREA - CALLED  BY BAL LINKE,PRCXXX* A2720390
*                                                                     * A2720400
* PRCRIS - FOR RIS CARD                                               * A2720410
* PRCMOD - FOR MODIFICATION CARD                                      * A2720420
* PRCOLD - FOR OLD RECORD                                             * A2720430
* PRCPAR - FOR / UPDATE CARD                                          * A2720440
*                                                                     * A2720450
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2720460
PRCRIS   MVC   RPIDT1(8),RLIDT1          RIS CARD                     A2720470
         MVC   RPIDT2(8),RLIDT2                                       A2720480
         MVC   RPCNT(1),RLCNT                                         A2720490
         MVC   RPMOD(1),RLMOD                                         A2720500
         MVC   RPNUM(8),RLNUM                                         A2720510
         MVC   RPSTPZ(8),RLSTPZ                                       A2720520
         MVC   RPDUPL(1),RLDUPL                                       A2720530
         MVC   RPCSD1(8),RLCSD1                                       A2720540
         MVC   RPNUM1(8),RLNUM1                                       A2720550
         MVC   RPBIN1(4),RLBIN1                                       A2720560
         MVC   RPCSD2(8),RLCSD2                                       A2720570
         MVC   RPNUM2(8),RLNUM2                                       A2720580
         MVC   RPBIN2(4),RLBIN2                                       A2720590
         MVC   RPVALD(1),RLVALD                                       A2720600
         MVC   RPCSDI(8),RLCSDI                                       A2720610
         MVC   RPNUMI(8),RLNUMI                                       A2720620
         MVC   RPBINI(4),RLBINI                                       A2720630
         MVC   RPSTP(8),RLSTP                                         A2720640
         MVC   RPSTEP(4),RLSTEP                                       A2720650
         MVC   RPCS1L(1),RLCS1L                                       A2720660
         MVC   RPCS2L(1),RLCS2L                                       A2720670
         MVC   RPCSNL(1),RLCSNL                                       A2720680
         MVC   RPCSSL(1),RLCSSL                                       A2720690
         BCR   15,LINKE                 RETURN TO CALLER             A2720700
PRCMOD   MVC   MPIDT(8),MLIDT           MODIF CARD                   A2720710
         MVC   MPCSD(8),MLCSD                                         A2720720
         MVC   MPNUM(8),MLNUM                                         A2720730
         MVC   MPBIN(4),MLBIN                                         A2720740
         MVC   MPNBR(4),MLNBR                                         A2720750
         MVC   MPVALD(1),MLVALD                                       A2720760
         MVC   MPCSL(1),MLCSL                                         A2720770
         BCR   15,LINKE                 RETURN TO CALLER             A2720780
PRCOLD   MVC   OPIDT(8),OLIDT           OLD RECORD                   A2720790
         MVC   OPCSD(8),OLCSD                                         A2720800
         MVC   OPNUM(8),OLNUM                                         A2720810
         MVC   OPBIN(4),OLBIN                                         A2720820
         MVC   OPVALD(1),OLVALD                                       A2720830
         MVC   OPCSL(1),OLCSL                                         A2720840
         BCR   15,LINKE                 RETURN TO CALLER             A2720850
PRCPAR   NI    FILESW,X'00'             / UPDATE CARD                A2720860
         CLI   GLFILE,X'E2'             IS SYMBOLIC FILE OPTION      A2720870
         BC    7,PRCPA1                 NO -BRANCH                   A2720880
         OI    FILESW,X'02'             YES-SET BIT6 OF FILESW       A2720890
PRCPA1   CLI   GLPRIN,X'C1'             IS ALL CS LISTED OPTION      A2720900
         BC    7,PRCPA2                 NO -BRANCH                   A2720910
         OI    PRINSW,X'01'             YES-SET BIT7 OF PRINSW       A2720920
```

```
PRCPA2   MVC   GLFILE(1),BLANK                                          A2720930
         MVC   GPZONE(40),GLZONE                                       A2720940
         MVC   GPCSD(8),GLCSD                                          A2720950
         MVC   GPCSL(1),GLCSL                                          A2720960
         BCR   15,LINKE                    RETURN TO CALLER            A2720970
         EJECT                                                         A2720980
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2720990
*                                                                      A2721000
* PRINT A MESSAGE ON PRINTER  - CALLED  BY                             A2721010
*     LA WORK,MESSAGE                                                  A2721020
*    BAL LINKD,PRMEX                                                   A2721030
*                                                                      A2721040
* ENTRY AT PRME1 - PRINT AN ERROR MESSAGE (SET THE SWITCH 'ANY ERROR   A2721050
*                  MESSAGE HAS BEEN PRINTED ON0 ) .                    A2721060
* ENTRY AT PRME3 - PRINT AN INFORMATION MESSAGE                        A2721070
*                                                                      A2721080
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721090
PRME1    OI    SWERR,X'03'               ERROR DETECTED              A2721100
PRME3    IC    WORKA,0(WORK)             MESSAGE LENGTH              A2721110
         BCTR  WORKA,0                   * -1                        A2721120
         CLI   1(WORK),X'F9'             *                           A2721130
         BC    7,PRME4                   *                           A2721140
         BCTR  WORKA,0                   * -1(SUPPRESS X'15' )       A2721150
PRME4    STC   WORKA,PRME2+1             * TO MVC LENGTH             A2721160
PRME2    MVC   PRZONE+10(1),1(WORK)      ERROR MESSAGE ITSELF        A2721170
         MVC   PRZONE+8(2),UPDCOD                                    A2721180
         BAL   LINKE,PRLINA              PRINTING OPERATION          A2721190
         BCR   15,LINKD                                              A2721200
         EJECT                                                       A2721210
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721220
*                                                                      A2721230
* PRINT A MESSAGE ON PRINTER-KEYBOARD    -   CALLED  BY               A2721240
*     LA WORK,MESSAGE                                                  A2721250
*    BAL LINKE,EDCSLX                                                  A2721260
*                                                                      A2721270
* ENTRY AT EDCSL1 - PRINT AN ERROR MESSAGE (SET THE SWITCH'ANY ERROR   A2721280
*                   MESSAGE HAS BEEN PRINTED ON0 ) .                   A2721290
* ENTRY AT EDCSL2 - PRINT AN INFORMATION MESSAGE                       A2721300
*                                                                      A2721310
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721320
EDCSL1   OI    SWERR,X'03'               ERROR DETECTED              A2721330
EDCSL2   BCR   0,0                                                   A2721340
EDCSL3   SR    WORKA,WORKA                                           A2721350
         IC    WORKA,0(WORK)             MESSAGE LENGTH              A2721360
         BCTR  WORKA,0                   * -1                        A2721370
         STC   WORKA,EDCSL4+1            *TO MVC LENGTH              A2721380
         LA    WORKA,3(WORKA)            * +3(SIMULATOR CODE)        A2721390
         STC   WORKA,OWCSL+2             *TO BUFFER LENGTH FOR SVC 4 A2721400
EDCSL4   MVC   CLZONE+2(1),1(WORK)       MESSAGE TO BUFFER           A2721410
         MVC   CLZONE(2),SIMCOD                                      A2721420
         BAL   LINKF,OWCSL               WRITE THE MESSAGE           A2721430
         MVI   CLZONE,X'40'              CLEAR THE BUFFER            A2721440
         MVC   CLZONE+1(79),CLZONE                                   A2721450
         BCR   15,LINKE                  RETURN TO CALLER            A2721460
         EJECT                                                       A2721470
```

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721480
*                                                                             * A2721490
* EDIT A LINE OF PRINTER LISTING - CALLED  BY  BAL LINKE,PREDX                * A2721500
*                                                                             * A2721510
* PREDA -80 BYTES FROM A  NEW RECORD                                          * A2721520
* PREDB -80 BYTES FROM AN OLD RECORD                                          * A2721530
* PREDD -8  BYTES FROM THE IDENTIFICATION OF AN OLD RECORD                    * A2721540
* PREDE -8  BYTES FROM THE IDENTIFICATION OF A MODIFICATION CARD              * A2721550
* PREDF -8  BYTES FROM THE IDENTIFICATION OF A / UPDATE CARD                  * A2721560
*                                                                             * A2721570
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721580
PREDA    MVC   NBCOD(80),NWZONE          CODE, TEXT ,IDENT.                  A2721590
PREDC    CLC   NBCOD+68(3),IPLCOD        IS   IPL1                           A2721600
         BC    7,PREDC1                  *OR  IPL2                           A2721610
         CLC   NBCOD+71(1),BLANK         *OR  IPL3 COL 69 THROUGH 72         A2721620
         BC    8,PREDC1                  NO                                  A2721630
         MVI   NBCOD,X'40'               YES-BLANKS COL 1 THR. 68            A2721640
         MVC   NBCOD+1(67),NBCOD         *                                   A2721650
         BCR   15,LINKE                  RETURN TO CALLER                    A2721660
PREDC1   CLC   NBCOD(1),BINCOD           IS BINARY RECORD                    A2721670
         BCR   7,LINKE                   NO-RETURN TO CALLER                 A2721680
         MVI   NBTXT+3,X'40'             BLANKS COL 5 THR. 72                A2721690
         MVC   NBTXT+4(67),NBTXT+3       *                                   A2721700
         MVC   NBCOD(1),BLANK            *  AND COL 1                        A2721710
         BCR   15,LINKE                  RETURN TO CALLER                    A2721720
PREDB    MVC   NBCOD(80),OLZONE          CODE  TEXT  IDENT                   A2721730
         BC    15,PREDC                  RETURN TO CALLER                    A2721740
PREDD    MVC   OBIDT(8),OLIDT            IDENTIFICATION                      A2721750
         TM    OLVALD,X'01'              VALID NUMBER IN OLD RECORD.         A2721760
         BCR   8,LINKE                   YES-RETURN TO CALLER                A2721770
         MVC   ERTBG+10(COD),MSGTBP      NO-SET ERROR CODE TO                A2721780
         MVC   ERCOD+10(COD),MSGTBP      *ERTBG AND TO PRINT BUFFER          A2721790
         BCR   15,LINKE                  RETURN TO CALLER                    A2721800
PREDE    MVC   MBIDT(8),MLIDT            IDENTIFICATION                      A2721810
         BCR   15,LINKE                  RETURN TO CALLER                    A2721820
PREDF    MVC   MESS29+5(3),NOT                                               A2721830
         TM    PRINSW,X'01'              IS ALL CS PRINTED OPTION            A2721840
         BC    8,PREDF2                  NO -BRANCH                          A2721850
         MVC   MESS29+5(3),BLANK                                             A2721860
PREDF2   MVC   MESS29+34(CS),GPCSD        CS IDENTIFICATION                   A2721870
         BCR   15,LINKE                                                      A2721880
         EJECT                                                               A2721890
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721900
*                                                                             * A2721910
* PRINT A LINE ON PRINTER  -  CALLED  BY  BAL LINKE,PRLINA                    * A2721920
*                                                                             * A2721930
*   1- SKIP TO NEXT PAGE IF THE LINE TO BE PRINTED IS THE FIRST LINE          * A2721940
*      OF A FILE OR OF A MODULE                                               * A2721950
*   2- PRINT THE LINE                                                         * A2721960
*                                                                             * A2721970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2721980
PRLINA   TM    LIN1ST,X'01'              IS 1ST LINE OF CS                   A2721990
         BC    8,PRLINB                  YES- SKIP TO NEXT PAGE              A2722000
         CLC   NBRLIN(4),MAXLIN          IS FULL PAGE                        A2722010
         BC    12,PRLINC                 NO-BRANCH                           A2722020
```

```
         BC    15,PRLINC                * * SKIP BY COUNT SUSPENDED     A2722030
PRLINB   XC    NBRLIN(4),NBRLIN           ZERO TO LINES COUNTER         A2722040
         BAL   LINKF,OPRSKP               SKIP TO NEXT PAGE             A2722050
PRLINC   BAL   LINKF,OPRINT               LINE PRINTING OPERATION       A2722060
         OI    LIN1ST,X'01'               1ST LINE IS PRINTED           A2722070
         L     WORKC,NBRLIN               LINES COUNT                   A2722080
         LA    WORKC,1(WORKC)             *                             A2722090
         ST    WORKC,NBRLIN               *                             A2722100
         MVI   PRZONE,X'40'               CLEAR BUFFER                  A2722110
         MVC   PRZONE+1(119),PRZONE                                     A2722120
         BCR   15,LINKE                                                 A2722130
         EJECT                                                          A2722140
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2722150
*                                                                       * A2722160
* I/O OPERATIONS - READ CARD FROM UPDTCORR                              * A2722170
*   CALLED  BY  BAL LINKF,ICARD                                         * A2722180
*                                                                       * A2722190
*   FUNCTION EXECUTED BY A SVC 18 (IO PACKAGE)                          * A2722200
*                                                                       * A2722210
*   EXIT                                                                * A2722220
*     1-CONDITION '07'-THE FUNCTION HAS CORRECTLY BEEN PERFORMED        * A2722230
*                     -RETURN TO CALLER.                                * A2722240
*     2-CONDITION '03'-ALL THE CARDS HAVE BEEN READ                     * A2722250
*                     -STORE '7F'IN BUFFER                              * A2722260
*                     -RETURN TO CALLER.                                * A2722270
*     3-CONDITION '02'-DEVICE MALFUNCTION                               * A2722280
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB).       * A2722290
*     4-CONDITION '01'-THE 'UPDTCORR'SYMBOL IS UNKNOWN                  * A2722300
*                     -PRINT A MESSAGE ON THE PRINTER-KEYBOARD          * A2722310
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2722320
*                                                                       * A2722330
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2722340
         CNOP  0,4                                                      A2722350
ICARD    SVC   18                         READ CARD                     A2722360
         DC    C'UPDTCORR'                SYMBOLE                       A2722370
         DC    FL2'80'                    COUNT                         A2722380
         DC    A(RISBUF)                  BUFF.ADDRESS                  A2722390
         TM    RISBUF,X'07'               IS CORRECT                    A2722400
         BCR   1,LINKF                    YES-RETURN TO CALLER          A2722410
         TM    RISBUF,X'03'               UPDTCORR END                  A2722420
         BC    12,ICARDC                  NO-BRANCH                     A2722430
         MVC   RISBUF(1),TPMARK           YES                           A2722440
         BCR   15,LINKF                   TO CALLER                     A2722450
ICARDC   TM    RISBUF,X'01'               UPDTCORR UNKNOWN              A2722460
         BC    1,ICARDA                   YES-BRANCH                    A2722470
         TM    RISBUF,X'02'               DEVICE MALFUNCTION            A2722480
         BC    1,ERSTPC                   YES - STOP                    A2722490
         BC    15,ERSTPC                  STOP                          A2722500
ICARDA   LA    WORK,ERME18                WRITE ERROR MESSAGE           A2722510
         MVC   ERME18+24(8),ICARD+2       *                             A2722520
         BAL   LINKE,EDCSL1               * ON PRINTER-KEYBOARD         A2722530
         BC    15,ERSTPC                  STOP                          A2722540
         EJECT                                                          A2722550
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2722560
*                                                                       * A2722570
```

```
* I/O OPERATIONS - READ AN OLD RECORD FROM UPDTOLD                      * A2722580
*   CALLED  BY BAL LINKF,IRECD                                          * A2722590
*                                                                       * A2722600
*   FUNCTION EXECUTED BY A SVC 18 (IO PACKAGE)                          * A2722610
*                                                                       * A2722620
*   EXIT                                                                * A2722630
*     1-CONDITION '07'-THE FUNCTION HAS CORRECTLY BEEN PERFORMED        * A2722640
*                     -AN OLD RECORD HAS BEEN READ                      * A2722650
*                     -RESET THE SWITCH 'TAPE MARK READ'OFF             * A2722660
*                     -RETURN TO CALLER.                                * A2722670
*     2-CONDITION '03'-A TAPE MARK HAS BEEN READ                        * A2722680
*                     -SET THE SWITCH'OLD DATA END'ON,IF THE READ TAPE* A2722690
*                      MARK FOLLOWS AN OTHER TAPE MARK                  * A2722700
*                     -SET THE SWITCH'OLD FILE END'ON,IF THE READ TAPE* A2722710
*                      MARK FOLLOWS AN OLD RECORD                       * A2722720
*                     -RETURN TO CALLER.                                * A2722730
*     3-CONDITION '02'-THE 'UPDTOLD'SYMBOL IS UNKNOWN                   * A2722740
*                     -PRINT A MESSAGE ON THE PRINTER KEYBOARD          * A2722750
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2722760
*     4-CONDITION '01'-DEVICE MALFUNCTION                               * A2722770
*                     -BRANCH TO ERSTPC(EXCEPTIONAL END OF JOB)         * A2722780
*                                                                       * A2722790
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2722800
         CNOP  0,4                                                      A2722810
IRECD    SVC   18                        READ OLD RECORD                A2722820
         DC    C'UPDTOLD '               SYMBOLE                        A2722830
         DC    FL2'80'                   COUNT                          A2722840
         DC    A(OLDBUF)                 BUFF-ADDRESS                   A2722850
         TM    OLDBUF,X'07'              CORRECT                        A2722860
         BC    1,IRECDD                  YES-BRANCH                     A2722870
         TM    OLDBUF,X'03'              TAPE MARK IS READ              A2722880
         BC    12,IRECDC                 NO -BRANCH                     A2722890
         TM    SWOLD,X'01'               IS FIRST TAPE MARK             A2722900
         BC    8,IRECDB                  YES-BRANCH                     A2722910
         OI    SWOLD,X'02'               NO-IS OLD DATA END (7F)2       A2722920
         BCR   15,LINKF                  RETURN TO CALLER               A2722930
IRECDB   OI    SWOLD,X'01'               IS OLD FILE END (7F)1          A2722940
         BCR   15,LINKF                  RETURN TO CALLER               A2722950
IRECDD   NI    SWOLD,X'00'               NOT ANY END                    A2722960
         BCR   15,LINKF                  RETURN TO CALLER               A2722970
IRECDC   TM    OLDBUF,X'01'              UPDTOLD UNKNOWN                A2722980
         BC    1,IRECDA                  YES-BRANCH                     A2722990
         TM    OLDBUF,X'02'              DEVICE MALFUNCTION             A2723000
         BC    1,ERSTPC                  YES -STOP                      A2723010
         BC    15,ERSTPC                 STOP                           A2723020
IRECDA   LA    WORK,ERME18               WRITE ERROR MESSAGE            A2723030
         MVC   ERME18+24(8),IRECD+2      *                              A2723040
         BAL   LINKE,EDCSL1              * ON PRINTER KEYBOARD          A2723050
         BC    15,ERSTPC                 STOP                           A2723060
         EJECT                                                          A2723070
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2723080
*                                                                       * A2723090
* I/O OPERATIONS - WRITE A NEW RECORD ON UPDTNEW                        * A2723100
*   CALLED  BY  BAL LINKF,OWRITE      WRITE A NEW RECORD                * A2723110
*               BAL LINKF,OWTMRK      WRITE A TAPE MARK                 * A2723120
```

```
*                                                                       * A2723130
*   FUNCTION EXECUTED  BY A SVC 18 (IO PACKAGE)                         * A2723140
*                                                                       * A2723150
*   EXIT                                                                * A2723160
*     1-CONDITION '07'-THE FUNCTION HAS CORRECTLY BEEN PERFORMED        * A2723170
*                     -RETURN TO CALLER.                                * A2723180
*     2-CONDITION '03'-STICKER DETECTED                                 * A2723190
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2723200
*     3-CONDITION '02'-THE 'UPDTNEW' SYMBOL IS UNKNOWN                  * A2723210
*                     -PRINT A MESSAGE ON THE PRINTER-KEYBOARD          * A2723220
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2723230
*     4-CONDITION '01'-DEVICE MALFUNCTION                               * A2723240
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2723250
*                                                                       * A2723260
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2723270
OWRITE   MVI   OWRIT1+1,80                  80 TO BUFFER  LENGTH        A2723280
         BC    15,OWRIT2                                                A2723290
OWTMRK   MVI   OWRIT1+1,X'01'               01 TO BUFFER  LENGTH        A2723300
         MVC   NEWBUF+1(1),TPMARK           TAPE MARK TO FIRST BYTE     A2723310
         CNOP  0,4                                                      A2723320
OWRIT2   SVC   18                           WRITE ON UPDTNEW            A2723330
         DC    C'UPDTNEW '                  SYMBOL                      A2723340
OWRIT1   DC    FL2'80'                      COUNT                       A2723350
         DC    A(NEWBUF)                    BUFFER ADDRESS              A2723360
         TM    NEWBUF,X'07'                 CORRECT                     A2723370
         BCR   1,LINKF                      YES-RETURN TO CALLER        A2723380
         TM    NEWBUF,X'03'                 STICKER DETECTED            A2723390
         BC    1,ERSTPC                *    YES - STOP                  A2723400
         TM    NEWBUF,X'01'                 UPDTNEW UNKNOWN             A2723410
         BC    1,OWRITA                     YES-BRANCH                  A2723420
         TM    NEWBUF,X'02'                 DEVICE MALFUNCTION          A2723430
         BC    1,ERSTPC                     YES-STOP                    A2723440
         BC    15,ERSTPC                    STOP                        A2723450
OWRITA   LA    WORK,ERME18                  WRITE ERROR MESSAGE         A2723460
         MVC   ERME18+24(8),OWRIT2+2        *                           A2723470
         BAL   LINKE,EDCSL1                 *ON PRINTER KEYBOARD        A2723480
         BC    15,ERSTPC                    STOP                        A2723490
         EJECT                                                          A2723500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2723510
*                                                                       * A2723520
* I/O OPERATIONS - WRITE A DUPLICATE OF A NEW RECORD ON DUPLFILE        * A2723530
*   CALLED  BY  BAL LINKF,ODUPL   WRITE A DUPLICATE                     * A2723540
*               BAL LINKF,ODTMRK  WRITE A TAPE MARK                     * A2723550
*                                                                       * A2723560
*   FUNCTION EXECUTED BY A SVC 18  (IO PACKAGE)                         * A2723570
*                                                                       * A2723580
*   EXIT                                                                * A2723590
*     1-CONDITION '07'-THE FUNCTION HAS CORRECTLY BEEN PERFORMED        * A2723600
*                     -RETURN TO CALLER.                                * A2723610
*     2-CONDITION '03' -STICKER DETECTED                                * A2723620
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .      * A2723630
*     3-CONDITION '02' -THE 'DUPLFILE'SYMBOL IS UNKNOWN.THIS CONDITION* A2723640
*                       DOES NOT OCCUR,AS THE FUNCTION IS NOT STARTED * A2723650
*                       IF THE CORRESPONDING DEVSUP CARD IS ABSENT.   * A2723660
*     4-CONDITION '01' -DEVICE MALFUNCTION                              * A2723670
```

```
*                    -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB)     * A2723680
*                                                                   * A2723690
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2723700
ODUPL    MVI   ODUPL1+1,80                 80 TO BUFFER LENGTH         A2723710
         BC    15,ODUPL2                                             A2723720
ODTMRK   MVI   ODUPL1+1,X'01'              01 TO BUFFER LENGTH         A2723730
         MVC   DPLBUF+1(1),TPMARK                                    A2723740
         CNOP  0,4                                                   A2723750
ODUPL2   SVC   18                          WRITE ON DUPLFILE           A2723760
         DC    C'DUPLFILE'                 SYMBOL                      A2723770
ODUPL1   DC    FL2'80'                     COUNT                       A2723780
         DC    A(DPLBUF)                   BUFFER ADDRESS              A2723790
         TM    DPLBUF,X'07'                CORRECT                     A2723800
         BCR   1,LINKF                     YES-RETURN TO CALLER        A2723810
         TM    DPLBUF,X'03'                STICKER  DETECTED           A2723820
         BC    1,ERSTPC              *     *STOP                       A2723830
         TM    DPLBUF,X'01'                DUPLFILE UNKNOWN            A2723840
         BC    1,ODUPLA                    YES-BRANCH                  A2723850
         TM    DPLBUF,X'02'                DEVICE MALFUNCTION          A2723860
         BC    1,ERSTPC                    YES-STOP                    A2723870
         BC    15,ERSTPC                   STOP                        A2723880
ODUPLA   LA    WORK,ERME18                 WRITE ERROR MESSAGE         A2723890
         MVC   ERME18+24(8),ODUPL2+2       *                           A2723900
         BAL   LINKE,EDCSL1                *ON PRINTER-KEYBOARD        A2723910
         BC    15,ERSTPC                   STOP                        A2723920
         EJECT                                                         A2723930
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2723940
*                                                                     * A2723950
* I/O OPERATIONS - PRINT A LINE ON PRINTER                            * A2723960
*   CALLED  BY BAL LINKF,OPRINT   PRINT A LINE                        * A2723970
*              BAL LINKF,OPRSKP   SKIP TO NEXT PAGE                   * A2723980
*              BAL LINKF,OPTMRK   WRITE A TAPE MARK (PRINTER IS       * A2723990
*                                          ASSIGNED  TO TAPE UNIT)    * A2724000
*                                                                     * A2724010
*   FUNCTION EXECUTED BY A SVC 18 (IO PACKAGE)                        * A2724020
*   EXIT                                                              * A2724030
*                                                                     * A2724040
*     1-CONDITION '07'-THE FUNCTION HAS CORRECTLY BEEN PERFORMED      * A2724050
*                     -RETURN TO CALLER.                              * A2724060
*     2-CONDITION '03' -STICKER DETECTED                              * A2724070
*                     -BRANCH TO ERSTPC (EXCEPTIONAL END OF JOB) .    * A2724080
*     3-CONDITION '02' -THE'PRINTER' SYMBOL IS UNKNOWN-THIS CONDITION * A2724090
*                       DOES NOT OCCUR,AS THE FUNCTION IS NOT STARTED * A2724100
*                       IF THE CORRESPONDING DEVSUP CARD IS ABSENT .  * A2724110
*     4-CONDITION '01' -DEVICE MALFUNCTION                            * A2724120
*                     -BRANCH TO ERSTPH (EXCEPTIONAL END OF JOB)      * A2724130
*                                                                     * A2724140
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2724150
OPTMRK   MVC   PRTBUF+1(1),TPMARK          TAPE MARK TO PRINT BUFFER   A2724160
         MVI   OPRIN1+1,X'01'              01 TO BUFFER LENGTH         A2724170
         BC    15,OPRIN2                                             A2724180
OPRINT   MVI   PRTBUF+1,X'00'              SPACE 1 LINE                A2724190
         MVI   OPRIN1+1,120                * AFTER PRINTING            A2724200
         BC    15,OPRIN2                                             A2724210
OPRSKP   MVI   PRTBUF+1,X'F1'              SKIP AT NEXT PAGE           A2724220
```

```
         MVI   OPRIN1+1,X'03'            PRINT 3 BLANKS              A2724230
         CNOP  0,4                                                   A2724240
OPRIN2   SVC   18                                                    A2724250
         DC    C'PRINTER '               SYMBOL                      A2724260
OPRIN1   DC    FL2'120'                  COUNT                       A2724270
         DC    A(PRTBUF)                 BUFFER ADDRESS              A2724280
         TM    PRTBUF,X'07'              CORRECT                     A2724290
         BCR   1,LINKF                                               A2724300
         TM    PRTBUF,X'03'              STICKER DETECTED            A2724310
         BC    1,ERSTPC                  YES- STOP                   A2724320
         TM    PRTBUF,X'01'              PRINTER UNKNOWN             A2724330
         BC    1,OPRINA                  YES                         A2724340
         TM    PRTBUF,X'02'              DEVICE MALFUNCTION          A2724350
         BC    1,ERSTPH                  YES-STOP                    A2724360
         BC    15,ERSTPH                 STOP                        A2724370
OPRINA   LA    WORK,ERME18               WRITE ERROR MESSAGE         A2724380
         MVC   ERME18+24(8),OPRIN2+2     *                           A2724390
         BAL   LINKE,EDCSL1              *ON PRINTER-KEYBOARD        A2724400
         BC    15,ERSTPH                 STOP                        A2724410
         EJECT                                                       A2724420
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2724430
*                                                                    * A2724440
* I/O OPERATIONS - WRITE A MESSAGE ON THE PRINTER-KEYBOARD           * A2724450
*   CALLED  BY BAL LINKF,OWCSL                                       * A2724460
*                                                                    * A2724470
*   FUNCTION EXECUTED BY A SVC 4 (CONTROL PROGRAM)                   * A2724480
*                                                                    * A2724490
*   EXIT                                                             * A2724500
*     1-CONDITION '00' -FUNCTION ACCEPTED,NOT PERFORMED              * A2724510
*                      -WAIT OTHER CONDITION .                       * A2724520
*     2-CONDITION '07' -FUNCTION CORRECTLY PERFORMED                 * A2724530
*                     -RETURN TO CALLER.                             * A2724540
*     3-CONDITION '01' -FUNCTION PERFORMED                           * A2724550
*                      -PERSISTENT ERROR (BAD SPELLING)              * A2724560
*                     -RETURN TO CALLER.                             * A2724570
*     4-CONDITION '03' -PROGRAM ERROR,FUNCTION NOT STARTED           * A2724580
*                      -BRANCH TO ERSTPE (EXCEPTIONAL END OF JOB)    * A2724590
*                                                                    * A2724600
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2724610
         CNOP  2,4                                                   A2724620
OWCSL    SVC   4                                                     A2724630
         DC    X'64'                     BUFFER LENGTH   100 BYTES   A2724640
         DC    AL3(CSLBUF)               BUFFER ADDRESS              A2724650
OWCSL1   TM    CSLBUF,X'07'              WRITING OPERATION IS        A2724660
         BC    8,OWCSL1                  *NOT PERFORMED -WAIT        A2724670
         BCR   1,LINKF                   *GOOD PERFORMED-TO CALLER   A2724680
         TM    CSLBUF,X'03'                                          A2724690
         BC    1,ERSTPA                  *NOT STARTED - STOP         A2724700
         TM    CSLBUF,X'01'                                          A2724710
         BCR   1,LINKF                   *PERF.(BAD SPELLING)-CALL.  A2724720
         BC    15,ERSTPA                 STOP                        A2724730
         EJECT                                                       A2724740
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2724750
*                                                                    * A2724760
* I/O OPERATIONS                                                     * A2724770
```

```
* REQUEST FOR IO CHARACTERISTICS OF A DEVICE                          * A2724780
*   CALLED  BY BAL LINKF,DEVTST                                       * A2724790
*                                                                     * A2724800
*   FUNCTION EXECUTED BY A SVC 17 (IO PACKAGE)                        * A2724810
*        (DVNAME CONTAINS THE NAME OF THE DEVICE TO BE TESTED         * A2724820
*                              ( @UPDTOLD@ FOR EXAMPLE )              * A2724830
*        (DVADD CONTAINS 4 ZEROS )                                    * A2724840
*        (DVTYP CONTAINS 4 BLANKS)                                    * A2724850
*        (DVIO  CONTAINS 1 BLANK )                                    * A2724860
*                                                                     * A2724870
*   EXIT                                                              * A2724880
*     1-DVIO+4 -IO PACKAGE ANSWERS 'THE DEVICE IS ASSIGNED'           * A2724890
*              -DVADD CONTAINS THE DEVICE ADDRESS                     * A2724900
*              -DVTYP CONTAINS THE DEVICE TYPE                        * A2724910
*              -DVIO  CONTAINS EITHER'I'(INPUT) OR'O'(OUTPUT)         * A2724920
*              -BRANCH TO 4(LINKF)                                    * A2724930
*                                                                     * A2724940
*     2-DVERR  -IO PACKAGE ANSWERS 'THE DEVICE IS NOT ASSIGNED'       * A2724950
*              -DVADD,DVTYP,DVIO ARE NOT CHANGED                      * A2724960
*              -BRANCH TO 0(LINKF)                                    * A2724970
*                                                                     * A2724980
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2724990
DEVTST   MVC   DVADD(2),ZERO                                           A2725000
         MVC   DVTYP(4),BLANK                                         A2725010
         MVC   DVIO(1),BLANK                                          A2725020
         CNOP  0,4                                                    A2725030
         SVC   17                                                     A2725040
DVNAME   DS    8C                      NAME OF TESTED  DEVICE        A2725050
DVADD    DC    X'0000'                 DEVICE ADDR.(STORED BY SVC)  A2725060
DVTYP    DC    C'    '                 DEVICE TYPE (STORED BY SVC)  A2725070
DVIO     DC    C' '                    I OR O      (STORED BY SVC)  A2725080
         DC    AL3(DVERR)              EXCEPTIONAL RETURN           A2725090
         BC    15,4(LINKF)             DEVICE IS ASSIGNED           A2725100
DVERR    BCR   15,LINKF                DEVICE IS NOT ASSIGNED       A2725110
         EJECT                                                        A2725120
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725130
*                                                                     * A2725140
* I/O OPERATIONS  -  REWIND A TAPE                                    * A2725150
*   CALLED  BY  BAL LINKF,REWIND                                      * A2725160
*                                                                     * A2725170
*   FUNCTION EXECUTED BY A SVC 13 (CONTROL PROGRAM)                   * A2725180
*                                                                     * A2725190
*   EXIT                                                              * A2725200
*     1-REWACC -'ACCEPTED RETURN'                                     * A2725210
*              -RETURN TO CALLING BY BCR 15,LINKF                     * A2725220
*               (AN INTERRUPTION IS EXPECTED AT CHANNEL END )         * A2725230
*     2-REWRET -EITHER 'NORMAL RETURN'FROM THE CONTROL PROGRAM          A2725240
*              -OR'EXCEPTIONAL RETURN'FROM THE CONTROL PROGRAM        * A2725250
*              -RETURN TO INTERRUPTION POINT BY SVC 3                 * A2725260
*                                                                     * A2725270
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725280
         CNOP  4,8                                                    A2725290
REWIND   SVC   13                                                     A2725300
REWADD   DC    X'0000'                 DEVICE ADDRESS               A2725310
         DC    A(REWCCW)               REWIND CCW ADDRESS           A2725320
```

```
         DS    C                              STATUS                          A2725330
         DS    3C                             SENSE                           A2725340
         DS    D                              OLD CSW                         A2725350
REWPSW   DS    D                              OLD PSW                         A2725360
         DC    A(REWRET)                      NORMAL RETURN ADDRESS           A2725370
         DC    A(REWRET)                      EXCEPTIONAL RETURN ADDRESS      A2725380
REWACC   BCR   15,LINKF                       REW. ACCEPTED-WAIT INTERR.      A2725390
         CNOP  2,4                                                            A2725400
REWRET   SVC   3                              RETURN TO INTERRUPT. POINT      A2725410
         DC    A(REWPSW)                      *(CALLING POINT)                A2725420
REWCCW   CCW   X'07',*,X'00',1                REWIND CCW                      A2725430
         EJECT                                                                A2725440
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725450
*                                                                             * A2725460
* PROGRAM CHECK ROUTINE                                                       * A2725470
*                                                                             * A2725480
* ITS   CHARACTERISTICS HAVE BEEN DEFINED IN THE INITIALIZATION               * A2725490
*     ROUTINE BY A SVC 6                                                      * A2725500
*                                                                             * A2725510
* THIS ROUTINE IS CALLED BY THE CONTROL PROGRAM AFTER A PROGRAM               * A2725520
*     INTERRUPTION                                                            * A2725530
*                                                                             * A2725540
* A MESSAGE IS PRINTED ON THE PRINTER                                         * A2725550
*                                                                             * A2725560
* BRANCH TO ERSTPE (EXCEPTIONAL END OF JOB)                                   * A2725570
*                                                                             * A2725580
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725590
PCHECK   LA    WORK,ERME21                     PRINT ERROR MESSAGE             A2725600
         BAL   LINKD,PRME1                     *AND                            A2725610
         BC    15,ERSTPE                       * STOP                          A2725620
         EJECT                                                                A2725630
*                                                                             A2725640
CS       EQU   8                                                             A2725650
NUM      EQU   8                                                             A2725660
*                                                                             A2725670
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725680
*                                                                             * A2725690
*  INPUT BUFFER  -  'UPDTOLD'   -  OLD RECORD                                 * A2725700
*                                                                             * A2725710
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725720
OLDBUF   DS    C              1 BYTE BEFORE DATA            1                 A2725730
OLZONE   DS    CL80           READING ZONE                 80 COL  1-80       A2725740
OLCOD    EQU   OLZONE         RECORD CODE,OR 1ST TEXT BYTE  1 COL  1- 1       A2725750
OLTXT    EQU   OLZONE+1       TEXT                         71 COL  2-72       A2725760
OLIDT    EQU   OLZONE+72      IDENTIFICATION                8 COL 73-80       A2725770
*                                                                             A2725780
OLCSD    DS    D              CS IDENT.(LEFT JUSTIFIED)     8                 A2725790
OLNUM    DS    D              RECORD NUMBER(RIGHT JUST.)    8                 A2725800
OLBIN    DS    F              BINARY VALUE OF OLNUM         4                 A2725810
OLCSL    DS    C              LENGTH OF CS IDENT IN OLIDT (LEFT PART)         A2725820
OLVALD   DS    C              BIT 7=1 FOR INVALID RECORD    1                 A2725830
*                                                                             A2725840
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725850
*                                                                             * A2725860
*  TO STORE PRECEDING OLD RECORD                                              * A2725870
```

```
*                                                                     * A2725880
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725890
OPIDT    DS    D             IDENTIFICATION                 8          A2725900
*                                                                       A2725910
OPCSD    DS    D             CS IDENT.                      8          A2725920
OPNUM    DS    D             RECORD NUMBER                  8          A2725930
OPBIN    DS    F             BINARY VALUE OF OPNUM          8          A2725940
OPCSL    DS    C             LENGTH OF CS IDENT IN OPIDT (LEFT PART)  A2725950
OPVALD   DS    C             BIT 7=1 FOR INVALID RECORD     1          A2725960
         EJECT                                                         A2725970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2725980
*                                                                     * A2725990
*  INPUT BUFFER  -  'UPDTCORR'  -  RIS CARD                           * A2726000
*                                                                     * A2726010
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726020
RISBUF   DS    C           1 BYTE BEFORE DATA             1          A2726030
RLZONE   DS    CL80        READING ZONE                  80 COL  1-80 A2726040
RLCOD    EQU   RLZONE      RIS CARD CODE                  4 COL  1- 4 A2726050
RLIDT1   EQU   RLZONE+7    1ST IDENTIFICATION             8 COL  8-15 A2726060
RLIDT2   EQU   RLZONE+23   2ND IDENTIFICATION             8 COL 24-31 A2726070
RLCNT    EQU   RLZONE+40   R,I,S,N=REQUESTED FUNCTION     1 COL 41-41 A2726080
RLMOD    EQU   RLZONE+43   C=MODIF. PER CS                1 COL 44-44 A2726090
RLNUM    EQU   RLZONE+51   INITIAL NEW NUMBER(RIGHT-JUST.)8 COL 52-59 A2726100
RLSTPZ   EQU   RLZONE+59   STEP FOR NUMBERING (RIGHT)     8 COL 60-67 A2726110
RLDUPL   EQU   RLZONE+68   D=DUPLICATE OF CS IS REQUESTED 1 COL 69-69 A2726120
*                                                                       A2726130
RLCSD1   DS    D           1ST CS IDENTIFICATION          8          A2726140
RLNUM1   DS    D           1ST NUMBER                     8          A2726150
RLBIN1   DS    F           BINARY VALUE OF RLNUM1         4          A2726160
RLCSD2   DS    D           2ND CS IDENTIFICATION          8          A2726170
RLNUM2   DS    D           2ND NUMBER                     8          A2726180
RLBIN2   DS    F           BINARY VALUE OF RLNUM2         4          A2726190
RLCSDI   DS    D           CS PART BEFORE NUMI            8          A2726200
RLNUMI   DS    D           INITIAL NEW NUMBER             8          A2726210
RLBINI   DS    F           BINARY VALUE OF RLNUMI         4          A2726220
RLSTP    DS    D           NUMBERING STEP                 8          A2726230
RLSTEP   DS    F           BINARY VALUE OF RLSTP          4          A2726240
RLCS1L   DS    C            LENGTH OF CS IDENT IN RLIDT1(LEFT PART)  A2726250
RLCS2L   DS    C            LENGTH OF CS IDENT IN RLIDT2(LEFT PART)  A2726260
RLCSNL   DS    C            LENGTH OF CS PART  IN RLNUM (LEFT PART)  A2726270
RLCSSL   DS    C            LENGTH OF CS PART  IN RLSTPZ(LEFT PART)  A2726280
RLVALD   DS    C           BIT 7=1,INVALID RIS CARD       1          A2726290
         EJECT                                                         A2726300
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726310
*                                                                     * A2726320
*  TO STORE PRECEDING RIS CARD                                        * A2726330
*                                                                     * A2726340
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726350
RPIDT1   DS    D             1ST IDENTIFICATION             8          A2726360
RPIDT2   DS    D             2ND IDENTIFICATION             8          A2726370
RPCNT    DS    C             R,I,S OR N                     1          A2726380
RPMOD    DS    C             C= MODIF.PER CS                1          A2726390
RPNUM    DS    D             INITIAL NEW NUMBER             8          A2726400
RPSTPZ   DS    D             STEP                           8          A2726410
RPDUPL   DS    C             D= DUPLICATE                   1          A2726420
```

```
*                                                                      A2726430
RPCSD1   DS    D            1ST CS IDENT                    8          A2726440
RPNUM1   DS    D            1ST NUMBER                      8          A2726450
RPBIN1   DS    F            BINARY VALUE OF RPNUM1          4          A2726460
RPCSD2   DS    D            2ND CS IDENT                    8          A2726470
RPNUM2   DS    D            2ND NUMBER                      8          A2726480
RPBIN2   DS    F            BINARY VALUE OF RPNUM2          4          A2726490
RPCSDI   DS    D            CS PART BEFORE NUMI             8          A2726500
RPNUMI   DS    D            INITIAL NEW NUMBER              8          A2726510
RPBINI   DS    F            BINARY VALUE OF RPNUMI          4          A2726520
RPSTP    DS    D            STEP                            8          A2726530
RPSTEP   DS    F            BINARY VALUE OF RPSTP           4          A2726540
RPCS1L   DS    C             LENGTH OF CS IDENT IN RPIDT1(LEFT PART)   A2726550
RPCS2L   DS    C             LENGTH OF CS IDENT IN RPIDT2(LEFT PART)   A2726560
RPCSNL   DS    C             LENGTH OF CS PART  IN RPNUM (LEFT PART)   A2726570
RPCSSL   DS    C             LENGTH OF CS PART  IN RPSTPZ(LEFT PART)   A2726580
RPVALD   DS    C            BIT 7=1,INVALID RIS CARD        1          A2726590
         EJECT                                                         A2726600
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726610
*                                                                      * A2726620
*  INPUT BUFFER  -  'UPDTCORR'  -  MODIFICATION CARD                   * A2726630
*                                                                      * A2726640
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726650
MLZONE   EQU   RLZONE        READING ZONE                80 COL  1-80 A2726660
MLCOD    EQU   MLZONE        CARD CODE,OR 1ST BYTE OF TEXT 1 COL  1- 1 A2726670
MLTXT    EQU   MLZONE+1      TEXT                        71 COL  2-72 A2726680
MLIDT    EQU   MLZONE+72     IDENTIFICATION               8 COL 73-80 A2726690
*                                                                      A2726700
MLCSD    DS    D             CS IDENT.                    8          A2726710
MLNUM    DS    D             NUMBER                       8          A2726720
MLBIN    DS    F             BINARY VALUE OF MLNUM        4          A2726730
MLNBR    DS    F             BIN.NUMBER OF MODIFS CARDS   4          A2726740
MLCSL    DS    C             LENGTH OF CS IDENT IN MLIDT (LEFT PART)  A2726750
MLVALD   DS    C             BIT 7=1,INVALID MODIF CARD   1          A2726760
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726770
*                                                                      * A2726780
*  TO STORE PRECEDING MODIFICATION CARD                                * A2726790
*                                                                      * A2726800
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726810
MPIDT    DS    D             IDENTIFICATION               8          A2726820
*                                                                      A2726830
MPCSD    DS    D             CS IDENT.                    8          A2726840
MPNUM    DS    D             NUMBER                       8          A2726850
MPBIN    DS    F             BINARY VALUE OF MPNUM        4          A2726860
MPNBR    DS    F             BIN.NUMBER OF MODIFS CARDS   4          A2726870
MPCSL    DS    C             LENGTH OF CS IDENT IN MPIDT (LEFT PART)  A2726880
MPVALD   DS    C             BIT 7=1,INVALID MODIF CARD   1          A2726890
         EJECT                                                         A2726900
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726910
*                                                                      * A2726920
*  OUTPUT BUFFER -  'UPDTNEW '  -  NEW RECORD                         * A2726930
*                                                                      * A2726940
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2726950
NEWBUF   DS    C             1 BYTE BEFORE DATA           1          A2726960
NWZONE   DS    CL80          WRITING ZONE                80 COL  1-80 A2726970
```

```
NWCOD    EQU   NWZONE        CODE,OR 1ST BYTE OF TEXT      1 COL  1- 1 A2726980
NWTXT    EQU   NWZONE+1      TEXT                         71 COL  2-72 A2726990
NWIDT    EQU   NWZONE+72     IDENTIFICATION                8 COL 73-80 A2727000
*                                                                       A2727010
NWCSD    DS    D             CS IDENT.                     8           A2727020
NWNUM    DS    D             NUMBER                        8           A2727030
NWBIN    DS    D             BIN.VALUE OF NWNUM(4 1ST BYT.)8           A2727040
MXNUM1   DS    D             HIGH VALUE OF NWNUM                       A2727050
MXBIN1   DS    F             HIGH VALUE OF NWBIN           4           A2727060
NWCSL    DS    C             LENGTH OF CS IDENT IN NWIDT               A2727070
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727080
*                                                                     * A2727090
*  OUTPUT BUFFER -  'DUPLFILE'  -  DUPLICATING OF NEW RECORD.         * A2727100
*                                                                     * A2727110
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727120
DPLBUF   EQU   NEWBUF       1 BYTE BEFORE DATA            1           A2727130
DPZONE   EQU   NWZONE       WRITING ZONE                 80           A2727140
DPLSW    DC    X'00'        A CS HAS BEEN DUPLICATED      1           A2727150
*                           BIT 7=1 DURING A FILE TREATMENT           A2727160
*                           BIT 6=1 DURING AN UPDATING                A2727170
         EJECT                                                        A2727180
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727190
*                                                                     * A2727200
*  OUTPUT BUFFER -  'PRINTER '  -  LINE OF UPDATE LISTING             * A2727210
*                                                                     * A2727220
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727230
PRTBUF   DS    C             1 BYTE BEFORE DATA            1           A2727240
PRZONE   DS    CL120         PRINTING ZONE               120 P   1-120 A2727250
ERCOD    EQU   PRZONE+3      ERROR CODES AREA             18 P   4- 21 A2727260
NBCOD    EQU   PRZONE+22     NEW RECORD AREA -CODE         1 P  23- 23 A2727270
NBTXT    EQU   PRZONE+23     *               -TEXT        71 P  24- 94 A2727280
NBIDT    EQU   PRZONE+94     *               -IDENT.       8 P  95-102 A2727290
MBIDT    EQU   PRZONE+103    MODIF CARD AREA -IDENT.       8 P 104-111 A2727300
OBIDT    EQU   PRZONE+112    OLD RECORD AREA -IDENT.       8 P 113-120 A2727310
*                                                                       A2727320
NBRLIN   DS    F             COUNT OF LINES BY PAGE        4           A2727330
MAXLIN   DC    F'56'         MAX.NUMBER OF LINES BY PAGE   4           A2727340
LIN1ST   DS    C             BIT7=1,1ST LINE OF CS PRINTED 1           A2727350
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727360
*                                                                     * A2727370
*  OUTPUT BUFFER - PRINTER-KEYBOARD    -  MESSAGE                     * A2727380
*                                                                     * A2727390
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727400
CSLBUF   DS    C             1 BYTE BEFORE DATA            1           A2727410
CLZONE   DS    CL100         WRITING ZONE                100 P   1-100 A2727420
         EJECT                                                        A2727430
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727440
*                                                                     * A2727450
*  INPUT BUFFER  -  'UPDTCORR'  -  / UPDATE CARD                      * A2727460
*                                                                     * A2727470
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727480
GLZONE   EQU   RLZONE       READING ZONE                  1 COL  1-80 A2727490
GLCOD    EQU   GLZONE       CARD CODE                     9 COL  1- 9 A2727500
GLFILE   EQU   GLZONE+10    S=SYMBOLIC FILE MODE          1 COL 11-11 A2727510
GLPRIN   EQU   GLZONE+12    A=ALL CS MUST BE LISTED       1 COL 13-13 A2727520
```

```
GLIDT    EQU   GLZONE+14      IDENTIFICATION(LEFT JUSTIF.)  8 COL 15-22 A2727530
GLFCT    EQU   GLZONE+23      I=INSERT A NEW FILE           1 COL 24-24 A2727540
*                                                                        A2727550
GLCSD    DS    D              CS IDENT.                     8           A2727560
GLCSL    DS    C              LENGTH OF CS IDENT IN GLIDT (LEFT PART)   A2727570
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727580
*                                                                       * A2727590
*  TO STORE PRECEDING / UPDATE CARD                                     * A2727600
*                                                                       * A2727610
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727620
GPZONE   DS    CL40                                        40          A2727630
GPIDT    EQU   GPZONE+14      IDENT.OF 1ST CS (LEFT )       8          A2727640
GPFCT    EQU   GPZONE+23      I=INSERT A NEW FILE           1          A2727650
*                                                                        A2727660
FILESW   DS    C              BIT6=0 BINARY,=1 SYMBOLIC FILE            A2727670
PRINSW   EQU   FILESW         BIT7=0 MODIFIED CS ONLY PRINTED,=1 ALL CS A2727680
GPCSL    DS    C              LENGTH OF CS IDENT IN GPIDT (LEFT PART)   A2727690
GPCSD    DS    D              CS IDENT.                     8           A2727700
         EJECT                                                          A2727710
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727720
*                                                                       * A2727730
* GENERAL REGISTERS                                                     * A2727740
*                                                                       * A2727750
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2727760
*                                                                       * A2727770
*        TO CALL SOME SUB-ROUTINES                                      * A2727780
*                                                                       * A2727790
* LINKA  1ST (HIGH)LEVEL                                                * A2727800
*      -REPLA (REPLACING ROUTINE )                                      * A2727810
*      -SUPPR (DELETING  ROUTINE )                                      * A2727820
*      -INSEX (INSERTION ROUTINE )                                      * A2727830
*      -COUNT (NUMBERING ROUTINE )                                      * A2727840
*                                                                       * A2727850
* LINKB  2ND LEVEL (ROUTINES CALLED FROM 1ST LEVEL )                    * A2727860
*      -SKCRDX (SKIP UPDTCORR CARDS)                                    * A2727870
*      -SKLDX  (COPY SOME OLD RECORDS,OR A MODULE,OR A FILE)            * A2727880
*                                                                       * A2727890
* LINKC  3RD LEVEL (CALLED FROM 1ST OR 2ND LEVEL)                       * A2727900
*      -RDCRD  (UPDTCORR CARD READING ROUTINE)                          * A2727910
*      -RDOLD  (UPDTOLD RECORD READING ROUTINE)                         * A2727920
*      -ENRXX  (UPDTNEW RECORD EDITING ROUTINE)                         * A2727930
*                                                                       * A2727940
* LINKD  4TH LEVEL (CALLED FROM 1ST,2ND OR 3RD LEVEL)                   * A2727950
**     -RSCAN  (TEST AN IDENTIFICATION)                                 * A2727960
*      -PRHEX  (PRINT A MESSAGE ON PRINTER)                             * A2727970
*                                                                       * A2727980
* LINKE  5TH LEVEL (CALLED FROM 1ST,2ND,3RD,OR 4TH LEVEL)               * A2727990
*      -PREDX  (EDIT A LINE OF LISTING)                                 * A2728000
*      -PRLINA (PRINT A LINE ON PRINTER)                                * A2728010
*      -EDCSLX (PRINT A MESSAGE ON PRINTER-KEYBOARD)                    * A2728020
*      -PRCXXX (STORE INPUT DATA IN SAVE AREA).                         * A2728030
*      -HEXBA  (CONVERT FROM ZONED TO BINARY)                           * A2728040
*      -BINHXA (CONVERT FROM BINARY TO ZONED)                           * A2728050
*                                                                       * A2728060
* LINKF 6TH LEVEL (CALLED FROM 1ST,2ND,3RD,4TH OR 5TH LEVEL)            * A2728070
```

```
*     I/O OPERATIONS                                                  * A2728080
*      -ICARD  (READ A CARD FROM UPDTCORR)                            * A2728090
*      -IRECD  (READ A RECORD FROM UPDTOLD)                           * A2728100
*      -OWRITE (WRITE A RECORD ON UPDTNEW)                            * A2728110
*      -ODUPL  (WRITE A DUPLICATE ON DUPLFILE)                        * A2728120
*      -OPRINT (PRINT A LINE ON PRINTER)                              * A2728130
*      -OWCSL  (WRITE A MESSAGE ON PRINTER-KEYBOARD)                  * A2728140
*      -DEVTST (REQUEST FOR DEVICE CHARACTERISTICS)                   * A2728150
*      -REWIND (REWIND   TAPE)                                        * A2728160
*                                                                     * A2728170
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728180
LINKA    EQU   1                                                      A2728190
LINKB    EQU   2                                                      A2728200
LINKC    EQU   3                                                      A2728210
LINKD    EQU   4                                                      A2728220
LINKE    EQU   5                                                      A2728230
LINKF    EQU   6                                                      A2728240
         EJECT                                                        A2728250
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728260
*                                                                     * A2728270
*        WORKING REGISTERS (CAN BE USED TO SUPPLY SUB-ROUTINES WITH   * A2728280
*                           SOME INFORMATION)                         * A2728290
* POINTR-(      HEXBA BINHXA             )                           * A2728300
* CONTR -(      HEXBA BINHXA             )                           * A2728310
* WORK  -(RSCAN       BINHXA PRMEX EDCSLX)                           * A2728320
* WORKA -(RSCAN                          )                           * A2728330
* WORKB -(RSCAN                          )                           * A2728340
* WORKC -(RSCAN                          )                           * A2728350
* WORKD                                                              * A2728360
*                                                                     * A2728370
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728380
POINTR   EQU   7                                                      A2728390
CONTR    EQU   8                                                      A2728400
WORK     EQU   9                                                      A2728410
WORKA    EQU   10                                                     A2728420
WORKB    EQU   11                                                     A2728430
WORKC    EQU   12                                                     A2728440
WORKD    EQU   13                                                     A2728450
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728460
*                                                                     * A2728470
*        BASE REGISTERS                                               * A2728480
*  BASERG,BASER2                                                      * A2728490
*                                                                     * A2728500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728510
BASER2   EQU   14                                                     A2728520
BASERG   EQU   15                                                     A2728530
         EJECT                                                        A2728540
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728550
*                                                                     * A2728560
*     VARIOUS CONSTANTS                                               * A2728570
*                                                                     * A2728580
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728590
TPMARK   DC    X'7F'                  TAPE MARK                      A2728600
SLASH    DC    C'/ UPDATE '                                          A2728610
BLANK    DC    C'         '           BLANKS                         A2728620
```

```
ZERO     DC    X'00000000'             ZEROS                          A2728630
ZEROF    DC    C'00000000'                                            A2728640
WKAREA   DS    D                       WORKING AREA                   A2728650
RISCOD   DC    X'02'                   RIS CARD CODE                  A2728660
         DC    C'RIS'                  *                              A2728670
BINCOD   DC    X'02'                   12 2 9 FOR COL.1               A2728680
IPLCOD   DC    C'IPL'                                                 A2728690
K1       DC    F'1'                                                   A2728700
KB2      DC    H'2'                                                   A2728710
KB4      DC    H'4'                                                   A2728720
KB8      DC    H'8'                                                   A2728730
K999     DC    C'999999999'                                           A2728740
NOT      DC    C'NOT'                                                 A2728750
TAPTYP   DC    C'2400'                                                A2728760
         EJECT                                                        A2728770
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728780
*                                                                     * A2728790
* 'SWD' SWITCH - RELATED TO BOTH INPUT DATA (UPDTOLD,UPDTCORR)        * A2728800
*                                                                     * A2728810
* SWDCE BIT 6=1 -'CARD  DATA END' (ALL THE 'UPTCORR' CARDS HAVE BEEN  * A2728820
*                  READ).                                             * A2728830
*                -IS SET TO 0     BY THE 'INIT' ROUTINE.             * A2728840
*                -IS SET TO 1     BY THE 'RDCRD' ROUTINE,IF THERE ARE * A2728850
*                  NO MORE CARD TO BE READ.                           * A2728860
*                -IS TESTED     BY THE 'FLTGA ' ROUTINE,TO SELECT THE * A2728870
*                  FUNCTION TO BE PERFORMED AT THE 'FILE' LEVEL.      * A2728880
*                                                                     * A2728890
* SWDOE BIT 7=1 -'OLD DATA END' (THE LAST TAPE MARK OF 'UPDTOLD' HAS  * A2728900
*                  BEEN READ).                                        * A2728910
*                -IS SET TO 0     BY THE 'INIT' ROUTINE.             * A2728920
*                -IS SET TO 1     BY THE 'RDOLD' ROUTINE,IF A TAPE MARK* A2728930
*                  HAS BEEN READ AND IF THE PRECEDING OLD RECORD IS   * A2728940
*                  ALSO A TAPE MARK.                                  * A2728950
*                -IS TESTED     BY THE 'FLTGA' ROUTINE.             * A2728960
*                                                                     * A2728970
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2728980
SWD      DC    X'00'                                                  A2728990
SWDCE    EQU   SWD             BIT 6=1   CARD DATA END                A2729000
SWDOE    EQU   SWD                 7=1   OLD  DATA END                A2729010
         EJECT                                                        A2729020
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2729030
*                                                                     * A2729040
* 'SWF' SWITCH - RELATED TO AN OLD FILE (MODULES FOLLOWED BY A TAPE   * A2729050
*                  MARK).                                             * A2729060
*              - RELATED TO A 'CARD' FILE (RIS AND MODIF.CARDS        * A2729070
*                  FOLLOWING A / UPDATE CARD).                        * A2729080
*                                                                     * A2729090
* SWF01 BIT 5=0 -'OLD FILE START' (THE FIRST RECORD OF AN OLD FILE    * A2729100
*                  HAS BEEN READ.NO OTHER FUNCTION HAS BEEN EXECUTED  * A2729110
*                  FOR THIS RECORD)                                   * A2729120
*                - IS SET TO 0     BY THE 'FLTGA' ROUTINE,AFTER THE 1ST* A2729130
*                  RECORD HAS BEEN READ.                              * A2729140
*                - IS SET TO 1     BY THE 'RDOLD' ROUTINE,  IF ANY OLD * A2729150
*                  RECORD HAS BEEN READ.                              * A2729160
*                                                                     * A2729170
```

```
* SWFCE BIT 6=1 -'CARD  FILE END' (THE CURRENT 'CARD' FILE HAS BEEN   * A2729180
*                 ENTIRELY READ).                                    * A2729190
*               - IS SET TO THE SAME VALUE AS THAT OF THE 'SWDCE'    * A2729200
*                 SWITCH.(IS ALSO 'FILE END' WHEN 'DATA END' HAS     * A2729210
*                 BEEN DETECTED).  ( 'FLTGA' ROUTINE )               * A2729220
*               - IS SET TO 1    BY  THE 'RDCRD' ROUTINE,IF A / UPDATE* A2729230
*                 CARD HAS BEEN READ (OR IF NO MORE CARDS).          * A2729240
*               - IS TESTED     BY THE 'CSTGA' ROUTINE TO SELECT THE  * A2729250
*                 FUNCTION TO BE PERFORMED AT THE 'CS' LEVEL.        * A2729260
*                                                                    * A2729270
* SWFOE BIT 7=1 -'OLD FILE END' (THE TAPE MARK FOLLOWING THE LAST    * A2729280
*                 MODULE OF AN OLD FILE HAS BEEN READ).              * A2729290
*               - IS SET TO THE SAME VALUE AS THAT OF THE 'SWDOE'    * A2729300
*                 SWITCH.(IS ALSO 'FILE END'AFTER 'DATA END' HAS     * A2729310
*                 BEEN DETECTED).  ( 'FLTGA' ROUTINE )               * A2729320
*               - IS SET TO 1   BY   THE 'RDOLD' ROUTINE,IF A TAPE   * A2729330
*                 MARK HAS BEEN READ).                               * A2729340
*               - IS TESTED     BY THE 'CSTGA' ROUTINE TO SELECT THE  * A2729350
*                 FUNCTION TO BE PERFORMED AT THE 'CS' LEVEL.        * A2729360
*                                                                    * A2729370
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2729380
SWF      DC    X'00'                                                 A2729390
SWFO1    EQU   SWF            BIT 5=1   1ST OLD RECORD IS TREATED    A2729400
SWFCE    EQU   SWF                6=1   CARD FILE END               A2729410
SWFOE    EQU   SWF                7=1   OLD  FILE END               A2729420
         EJECT                                                       A2729430
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2729440
*                                                                    * A2729450
* 'SWCS' SWITCH - RELATED TO AN OLD MODULE(CONTROL SECTION)          * A2729460
*               - RELATED TO A GROUP OF RIS AND MODIF.CARDS ,WHICH   * A2729470
*                 AFFECTS  THE SAME MODULE.                          * A2729480
*                                                                    * A2729490
* SWCSC1 BIT 4=1 -'FIRST RIS CARD ALREADY TREATED'                   * A2729500
*                - IS SET TO 0 AT THE BEGINNING OF THE 'CSTGA'       * A2729510
*                  ROUTINE (A MODULE WILL BE TREATED,AN RIS CARD HAS * A2729520
*                  ALREADY BEEN READ,NO FUNCTION HAS BEEN PERFORMED  * A2729530
*                  FOR THIS CARD).                                   * A2729540
*                - IS SET TO 1     BY THE 'RDCRD' ROUTINE,IF AN RIS   * A2729550
*                  CARD HAS BEEN READ.                               * A2729560
*                                                                    * A2729570
* SWCSO1 BIT 5=1 -'FIRST OLD RECORD ALREADY TREATED'                 * A2729580
*                - IS SET TO 0 AT THE BEGINNING OF THE 'CSTGA'ROUTINE.* A2729590
*                - IS SET TO 1 AT THE BEGINNING OF THE 'RDOLD'ROUTINE.* A2729600
*                - IS SET TO 1     BY THE 'INSEA' ROUTINE (DURING AN  * A2729610
*                  INSERTION,THE OLD RECORDS MUST NOT BE READ).      * A2729620
*                - IS TESTED     BY THE 'RCTGB' ROUTINE,IF AN INSER-  * A2729630
*                  TION IS REQUESTED AT THE BEGINNING OF THE MODULE. * A2729640
*                                                                    * A2729650
* SWCSCE BIT 6=1 -'CARD CS END'(THE CARD GROUP RELATED TO A MODULE   * A2729660
*                  HAS BEEN ENTIRELY READ).                          * A2729670
*                - IS SET TO 0 AT THE BEGINNING OF THE 'CSTGA'ROUTINE.* A2729680
*                - IS SET TO 1     BY THE 'RDCRD' ROUTINE,IF A      * A2729690
*                  CARD RELATED TO  ANOTHER MODULE HAS BEEN READ.    * A2729700
*                - IS TESTED     BY THE 'RCTGB' ROUTINE TO SELECT THE * A2729710
*                  FUNCTION TO BE PERFORMED AT THE 'RECORD' LEVEL.   * A2729720
```

```
*                - IS TESTED AT THE END OF THE 'RISN' ROUTINE.        * A2729730
*                                                                     * A2729740
*                                                                     * A2729750
* SWCSOE BIT 7=1 -'OLD CS END' (THE CURRENT OLD MODULE HAS BEEN       * A2729760
*                  ENTIRELY READ).                                    * A2729770
*                - IS SET TO 0 AT THE BEGINNING OF THE 'CSTGA'ROUTINE.* A2729780
*                - IS SET TO 1     BY THE 'RDOLD' ROUTINE,IF THE READ * A2729790
*                  OLD RECORD BELONGS TO THE NEXT MODULE.             * A2729800
*                - IS SET TO 1     BY THE 'INSEA' ROUTINE(SEE SWCS01).* A2729810
*                - IS TESTED     BY THE 'RCTGB' ROUTINE TO SELECT THE * A2729820
*                  FUNCTION TO BE PERFORMED AT THE 'RECORD' LEVEL.    * A2729830
*                - IS TESTED AT THE EXCEPTIONAL END OF THE 'RISN'     * A2729840
*                  ROUTINE.                                           * A2729850
*                                                                     * A2729860
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2729870
SWCS     DC    X'00'                                                  A2729880
SWCSC1   EQU   SWCS          BIT 4=1   1ST RIS CARD OF CS IS TREATED  A2729890
SWCS01   EQU   SWCS              5=1   1ST OLD RECORD OF CS IS TREATED A2729900
SWCSCE   EQU   SWCS              6=1   CARD CS END                     A2729910
SWCSOE   EQU   SWCS              7=1   OLD  CS END                     A2729920
         EJECT                                                        A2729930
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2729940
*                                                                     * A2729950
* 'SWSET' SWITCH -RELATED TO A SET OF OLD RECORDS TO BE CORRECTED     * A2729960
*                                                                     * A2729970
*              - RELATED TO A SET OF 'UPDTCORR' (AN RIS CARD AND ITS  * A2729980
*                MODIFICATION CARDS WHICH REQUESTS A CORRECTION IN    * A2729990
*             ALL OR PART OF AN OLD MODULE)                           * A2730000
*                                                                     * A2730010
* SWSECL BIT 2=1 -'LAST MODIF.CARD TREATED'(CARD SET END)             * A2730020
*                - IS SET TO 0 BY THE 'REPLA' AND 'INSEA' ROUTINES.   * A2730030
*                  (DURING A REPLACEMENT OR AN INSERTION THE MODIFICA-* A2730040
*                   TION CARDS MUST BE READ).                         * A2730050
*                - IS SET TO 1 BY THE 'SUPPR' AND 'COUNT' ROUTINES.   * A2730060
*                  (DURING A DELETION OR A NUMBERING THE MODIFICATION * A2730070
*                   CARDS MUST NOT BE READ).                          * A2730080
*                - IS SET TO 1     BY THE 'RISN' ROUTINE   IF THE CARD* A2730090
*                  FOLLOWING THE LAST MODIF.CARD TO BE TREATED HAS    * A2730100
*                  BEEN READ .                                        * A2730110
*                - IS TESTED     BY THE 'RISN' ROUTINE TO STOP OR TO  * A2730120
*                  CONTINUE THE INSERTING PHASE OF THE CURRENT FUNCTION* A2730130
*                                                                     * A2730140
* SWSEOL BIT 3=1 -'LAST OLD RECORD TREATED' (OLD SET END)             * A2730150
*                - IS SET TO 0     BY THE 'REPLA','SUPPR' AND 'COUNT' * A2730160
*                  ROUTINES(DURING A REPLACEMENT,A DELETION OR A NUMBE-* A2730170
*                  RING,THE OLD RECORDS MUST BE READ).                * A2730180
*                - IS SET TO 1     BY THE 'INSEA' ROUTINE.(DURING AN  * A2730190
*                  INSERTION,THE OLD RECORDS MUST NOT BE READ).       * A2730200
*                - IS SET TO 1     BY THE'RISN' ROUTINE   IF THE OLD  * A2730210
*                  RECORD FOLLOWING THE LAST OLD RECORD TO BE CORRECTED* A2730220
*                  HAS BEEN READ.                                     * A2730230
*                - IS TESTED     BY THE 'RISN' ROUTINE TO STOP OR TO  * A2730240
*                  CONTINUE READING OF THE OLD RECORDS.               * A2730250
*                                                                     * A2730260
* SWSEC1 BIT 4=1 -'THE 1ST MODIF.CARD OF THE CURRENT SET HAS BEEN     * A2730270
```

```
*                TREATED'                                              * A2730280
*              - IS SET TO 0     BY THE 'REPLA' AND 'INSEA' ROUTINES.* A2730290
*              - IS SET TO 1     BY THE 'SUPPR' AND 'COUNT' ROUTINES.* A2730300
*              - IS SET TO 1     BY THE 'RISN' ROUTINE,  IF THE 1ST  * A2730310
*                MODIFICATION CARD HAS BEEN TREATED.                 * A2730320
*              - IS TESTED     BY THE  'RISN ROUTINE' TO EXAMINE THE * A2730330
*                IDENTIFICATION OF THIS FIRST CARD.                  * A2730340
*                                                                    * A2730350
* SWSEO1 BIT 5=1 -'THE 1ST OLD RECORD OF THE CURRENT SET HAS BEEN    * A2730360
*                 TREATED'                                           * A2730370
*              - IS SET TO 0     BY THE 'REPLA','SUPPR' AND 'COUNT'  * A2730380
*                ROUTINES.                                           * A2730390
*              - IS SET TO 1     BY THE 'INSEA' ROUTINE.             * A2730400
*              - IS SET TO 1     BY THE 'RISN' ROUTINE,AFTER THE 1ST * A2730410
*                OLD RECORD HAS BEEN TREATED.                        * A2730420
*              - IS TESTED     BY THE 'RISN' ROUTINE  TO SELECT THE  * A2730430
*                APPROPRIATE VALUE TO BE ASSIGNED TO THE NEW NUMBER  * A2730440
*                OF THIS FIRST RECORD.                               * A2730450
*                                                                    * A2730460
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730470
SWSET    DC    X'00'                                                  A2730480
SWSECL   EQU   SWSET          BIT 2=1   LAST MODIF TREATED           A2730490
SWSEOL   EQU   SWSET              3=1   LAST OLD RECORD TREATED      A2730500
SWSEC1   EQU   SWSET              4=1   1ST MODIF CARD OF SET TREATED A2730510
SWSEO1   EQU   SWSET              5=1   1ST OLD RECORD OF SET IS TREATD A2730520
SWSECE   EQU   SWSET              6=1   CARD SET END                  A2730530
SWSEOE   EQU   SWSET              7=1   OLD  SET END                  A2730540
         EJECT                                                        A2730550
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730560
*                                                                    * A2730570
* 'SWSK' SWITCH -                                                    * A2730580
*                                                                    * A2730590
* SWSKF BIT  5  - USED     BY THE 'SKLDN' ROUTINE.                   * A2730600
*               - IS SET TO 0 AT 'SKLDN' (DELETE AN OLD FILE)        * A2730610
*               - IS SET TO 1 AT 'SKLDM' (COPY AN OLD FILE)          * A2730620
*                                                                    * A2730630
* SWSKRD BIT 6                                                       * A2730640
* SWSKAM BIT 7  - USED     BY THE 'SKLDA' ROUTINE.                   * A2730650
*               - ARE SET TO 00 AT 'SKLDA'(COPY AN OLD MODULE).      * A2730660
*               - ARE SET TO 11 AT 'SKLDB'(COPY AN OLD RECORD).      * A2730670
*               - ARE SET TO 01 AT 'SKLDC'(COPY THE REMAINDER OF AN  * A2730680
*                 OLD MODULE).                                       * A2730690
*                                                                    * A2730700
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730710
SWSK     DC    X'00'                                                  A2730720
SWSKF    EQU   SWSK           BIT 5,0 DELETE ,1 COPY,AN OLD FILE     A2730730
SWSKRD   EQU   SWSK           BIT 6=1   ONE RECORD TREATMENT         A2730740
SWSKAM   EQU   SWSK           BIT 7=1   SEVERAL RECORDS TREATMENT    A2730750
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730760
*                                                                    * A2730770
* 'SWSK1' SWITCH - USED     BY THE 'SKCRD' ROUTINE.                  * A2730780
*  (BIT 7)       - IS SET TO 0 AT'SKCRDA' (SKIP UP TO NEXT MODULE).  * A2730790
*                - IS SET TO 1 AT'SKCRDM' (SKIP UP TO NEXT FILE).    * A2730800
*                                                                    * A2730810
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730820
```

```
SWSK1    DC    X'00'          BIT 7,0 UP TO RIS CARD,1 UP TO / UPDATE   A2730830
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730840
*                                                                       * A2730850
* 'SWENR' SWITCH - USED     BY THE 'ENRMX' ROUTINE.                     * A2730860
*  (BIT 7)       - IS SET TO 0 AT 'ENRON','ENRO1','ENROA' (NEW RECORD * A2730870
*                  ISSUED FROM OLD RECORD).                             * A2730880
*                - IS SET TO 1 AT 'ENRMA'(NEW RECORD ISSUED FROM       * A2730890
*                  MODIFICATION CARD).                                  * A2730900
*                                                                       * A2730910
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730920
SWENR    DC    X'00'          BIT7 ,0 FROM OLD RECORD,1 FROM MODIF CARD A2730930
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2730940
*                                                                       * A2730950
* 'SWIDT' SWITCH - USED     BY THE EDITION OF A NEW RECORD.             * A2730960
*  (BIT 7)       - SET TO 0     BY THE 'REPLA','SUPPR','INSEA'         * A2730970
*                  ROUTINES.(DURING A REPLACEMENT,A DELETION OR AN     * A2730980
*                  INSERTION,THE IDENTIFICATION OF THE MODULE TO BE   * A2730990
*                  WRITTEN WILL NOT BE CHANGED).                        * A2731000
*                - SET TO 0     BY THE 'COUNT' ROUTINE.                 * A2731010
*                - SET TO 1     BY THE 'COUNT' ROUTINE,IF THE MODULE    * A2731020
*                  IDENTIFICATION MUST BE CHANGED,AS REQUESTED IN THE * A2731030
*                  RIS CARD.                                            * A2731040
*                - SET TO 0 AT 'ENROA'.                                 * A2731050
*                                                                       * A2731060
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731070
SWIDT    DC    X'00'          BIT 7=1 CHANGE CS.ID IN NUMBERING ROUTINE A2731080
         EJECT                                                          A2731090
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731100
*                                                                       * A2731110
* 'SWERR' SWITCH - 'AN ERROR HAS BEEN DETECTED'                         * A2731120
*                                                                       * A2731130
*   BIT 7 = 1    -'ERROR DETECTED DURING THE TREATMENT OF A FILE'       * A2731140
*                - IS SET TO 0     BY THE 'INIT' AND 'FLTGA' ROUTINES.* A2731150
*                - IS SET TO 1     BY THE 'PRME1' AND 'EDCSL1' ROUTI- * A2731160
*                  NES.(AN ERROR MESSAGE IS PRINTED)                    * A2731170
*                - IS TESTED     BY THE 'CSTGC' ROUTINE TO PRINT A      * A2731180
*                  MESSAGE AT THE END OF A FILE TREATMENT.              * A2731190
*                                                                       * A2731200
*   BIT 6 =1     -'ERROR DETECTED DURING AN UPDATING'(JOB LEVEL).       * A2731210
*                - IS SET TO 0     BY THE 'INIT' ROUTINE.               * A2731220
*                - IS SET TO 1     BY THE'PRME1' AND'EDCSL1' ROUTINES.* A2731230
*                - IS SET TO 1     BY THE 'CSTGC' ROUTINE IF AN ERROR * A2731240
*                  MESSAGE (I,O,N OR D) IS PRINTED.                     * A2731250
*                - IS TESTED     BY THE 'NLSTOP' ROUTINE TO PRINT A     * A2731260
*                  MESSAGE AT UPDATING END.                             * A2731270
*                                                                       * A2731280
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731290
SWERR    DC    X'00'          ERROR(BIT 6'JOB'LEVEL)(BIT 7'FILE'LEVEL)  A2731300
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731310
*                                                                       * A2731320
* 'SWRISN' SWITCH -'FUNCTION TO BE EXECUTED DURING THE 'RISN' ROUTINE'* A2731330
* (BITS 4,5,6 AND 7)                                                    * A2731340
*                                                                       * A2731350
*                -ARE SET TO 1000     BY THE 'REPLA' ROUTINE.          * A2731360
*                -ARE SET TO 0100     BY THE 'INSEA' ROUTINE.          * A2731370
```

```
*              -ARE SET TO 0010     BY THE 'SUPPR' ROUTINE.          * A2731380
*              -ARE SET TO 0001     BY THE 'COUNT' ROUTINE.          * A2731390
*                                                                    * A2731400
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731410
SWRISN   DC    X'00'          BITS 4 5 6 7                           A2731420
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731430
*                                                                    * A2731440
* 'SWOLD' SWITCH - 'TAPE MARKS FROM  UPDTOLD' - SET     BY THE 'IRECD.* A2731450
*                  ROUTINE.                                          * A2731460
*                                                                    * A2731470
*   BIT 6 (FILE) - SET TO 0  IF THE READ OLD RECORD IS NOT A TAPE MARK* A2731480
*                - SET TO 1  IF IT IS A TAPE MARK.                   * A2731490
*                                                                    * A2731500
*   BIT 7 (DATA) - SET TO 0  IF IT IS NOT A TAPE MARK.               * A2731510
*                - SET TO 1  IF IT IS A TAPE MARK FOLLOWING  ANOTHER * A2731520
*                  TAPE MARK.                                        * A2731530
*                                                                    * A2731540
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731550
SWOLD    DC    X'00'          00 NOT ANY END,01 FILE END,11 DATA END  A2731560
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731570
*                                                                    * A2731580
*'SWPRER' SWITCH - USED     BY THE 'ERSTPX' ROUTINE.                * A2731590
*  (BIT 7)       - IS SET TO 1 IF THE ERROR IS A PRINTER ERROR       * A2731600
*                  (PRINT THE LAST MESSAGES ON THE PRINTER-KEYBOARD  * A2731610
*                   ONLY)                                            * A2731620
*                                                                    * A2731630
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731640
SWDUMP   DC    X'00'          BIT 7=1 DUMP REQUESTED                  A2731650
SWPRER   DC    X'00'          BIT 7=1 STOP FOR PRINTER ERROR          A2731660
         EJECT                                                       A2731670
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731680
*                                                                    * A2731690
* UPDATING MESSAGES  TYPE = AX9XX  FOR PRINTER AND PRINTER-KEYBOARD  * A2731700
*                    TYPE = UPXXX  FOR PRINTER ONLY                  * A2731710
*                                                                    * A2731720
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2731730
*                                                                    * A2731740
* END OF JOB - MESSAGES COMMON TO PRINTER AND PRINTER-KEYBOARD       * A2731750
*                    - 4 POSSIBLE CASES                              * A2731760
*                                                                    * A2731770
* 1 *************************                                      * A2731780
*   * AX90A END OF UPDATING *                                        * A2731790
*   *************************                                        * A2731800
*     - NORMAL UPDATING END                                          * A2731810
*     - (REFER TO 'INFORMATION MESSAGES' LISTING )                   * A2731820
*                                                                    * A2731830
* 2 ************************************************               * A2731840
*   * AX91W CONTROL CARD ERROR,UPDTXXX NOT ASSIGNED *                * A2731850
*   * AX92W UPDATING ERROR,CANNOT CONTINUE         *                * A2731860
*   ************************************************               * A2731870
*     - PROGRAM INTERRUPTED BEFORE NORMAL END AT ERROR DETECTION     * A2731880
*     - CORRECT ERRONEOUS DEV360 OR DEVSUP CARD(UPDTOLD,..NEW,..CORR) A2731890
*     - RESUME UPDATING                                              * A2731900
*                                                                    * A2731910
* 3 ***************************************                        * A2731920
```

```
*   * AX92W UPDATING ERROR,CANNOT CONTINUE *                          * A2731930
*   ****************************************                          * A2731940
*     - PROGRAM INTERRUPTED BEFORE NORMAL END AT ERROR DETECTION      * A2731950
*     - 2 CAUSES POSSIBLE (IN BOTH CASES,UPDATING HAS TO BE RESUMED)  * A2731960
*         -EITHER- AN ERROR ON A UNIT HAS BEEN INDICATED TO PRINTER-  * A2731970
*              KEYBOARD BY THE CONTROL-PROGRAM,AND THE OPERATOR HAS   * A2731980
*              ANSWERD 'STOP'                                         * A2731990
*           -OR- CATASTROPHIC ERROR HAS BEEN DETECTED BY THE UPDATING * A2732000
*              PROGRAM                                                * A2732010
*     - (REFER TO 'CATASTROPHIC ERROR MESSAGES' LISTING )             * A2732020
*                                                                     * A2732030
* 4 ****************************************                          * A2732040
*   * AX90A END OF UPDATING                *                          * A2732050
*   * AX93W POSSIBLE UPDATING ERROR DETECTED *                        * A2732060
*   ****************************************                          * A2732070
*     - THE PROGRAM HAS BEEN EXECUTED TO DATA END                     * A2732080
*     - ERRORS HAVE BEEN DETECTED DURING PROCESSING                   * A2732090
*     - (REFER TO 'ERROR MESSAGES - AT FILE LEVEL')                   * A2732100
*     - WHEN ERROR HAS BEEN FOUND,CHECK IF UPDATING IS CORRECT,       * A2732110
*      OTHERWISE CORRECT ERROR AND RESUME UPDATING.                   * A2732120
*                                                                     * A2732130
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732140
        EJECT                                                          A2732150
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732160
*                                                                     * A2732170
* PRINTER INFORMATION MESSAGES - NORMAL FONCTIONNING                  * A2732180
*              ( SEE 'END OF JOB' - 1ST CASE )                        * A2732190
*                                                                     * A2732200
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732210
*                                                                     * A2732220
* 1 AT START OF A FILE PROCESSING                                     * A2732230
*                                                                     * A2732240
*    ************************************                             * A2732250
*    * / UPDATE (CARD-IMAGE)            *                             * A2732260
*    * UP13I THE NEXT FILE WILL BE COPIED *                           * A2732270
*    ************************************                             * A2732280
*     - ALL MODULES OF THE FILE WILL BE COPIED,TAPE-MARK INCLUDED     * A2732290
*                                                                     * A2732300
*    ************************************************************     * A2732310
*    * UP12I THE NEXT FILE WILL BE DELETED.ITS FIRST CS IS XXXX *     * A2732320
*    ************************************************************     * A2732330
*     - NONE OF THE MODULES (CONTROL-SECTIONS) OF THE FILE WILL BE    * A2732340
*       COPIED ON THE NEW TAPE. THE TAPE-MARK IS NOT COPIED           * A2732350
*                                                                     * A2732360
*    ********************************************                     * A2732370
*    * / UPDATE (CARD-IMAGE)                      *                   * A2732380
*    * UP14I (NOT)ALL CS LISTED. THE 1ST CS IS XXXX *                 * A2732390
*    ********************************************                     * A2732400
*     - A CORRECTION,TO ONE OF THE MODULES OF THE FILE STARTING BY    * A2732410
*       THE MODULE XXXX,IS REQUESTED BY AN RIS CARD                   * A2732420
*                                                                     * A2732430
*    ************************************                             * A2732440
*    * UP15I ALL OLD FILES HAVE BEEN TREATED *                        * A2732450
*    ************************************                             * A2732460
*     - THE LAST TAPE MARK OF UPDTOLD HAS BEEN READ. THE ONLY         * A2732470
```

```
*       ACCEPTABLE TYPE OF CORRECTION IS THEN THE INSERTION OF A NEW  * A2732480
*       FILE AT TAPE END.                                             * A2732490
*                                                                     * A2732500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732510
        EJECT                                                           A2732520
* * * * * * * * * * * *  CONTINUED  * * * * * * * * * * * * * * * * * A2732530
*                                                                     * A2732540
* 2 DURING A FILE PROCESSING (CORRECTIONS REQUESTED BY RIS CARDS )    * A2732550
*                                                                     * A2732560
*    ********************************************                     * A2732570
*    * UP24I THE NEXT CS,XXXXXXXX WILL BE COPIED *                     * A2732580
*    ********************************************                     * A2732590
*     - THE OLD MODULE WHOSE THE 1ST RECORD IS IDENTIFIED BY XXXXXXXX * A2732600
*       WILL BE COPIED ON THE NEW TAPE. NO CORRECTION REQUEST HAS     * A2732610
*       BEEN MADE FOR THIS MODULE. LISTING OF ALL  ITS  RECORDS WILL  * A2732620
*       BE PRINTED IF THE CURRENT / UPDATE CARD REQUESTS IT.          * A2732630
*                                                                     * A2732640
*    ******************************************************           * A2732650
*    *  RIS (CARD IMAGE)                                 *           * A2732660
*    * UP19I (REPLACE,INSERT,SUPPRESS OR NUMBERING)REQUEST *          * A2732670
*    ******************************************************           * A2732680
*     - A CORRECTION IS REQUESTED FOR A MODULE AND ALL ITS NEW RECORDS* A2732690
*       WILL BE PRINTED                                               * A2732700
*                                                                     * A2732710
* 3 AT THE END OF FILE PROCESSING                                     * A2732720
*                                                                     * A2732730
*    *********************************                                * A2732740
*    * UP23I FILE TREATMENT FINISHED *                                * A2732750
*    *********************************                                * A2732760
*     - NO ERROR HAS BEEN DETECTED DURING PROCESSING OF THIS FILE     * A2732770
*     - EXCEPT WHEN THE FILE HAS BEEN DELETED,A TAPE MARK IS WRITTEN  * A2732780
*       ON UPDTNEW (AND ON DUPLFILE IF AT LEAST ONE MODULE OF THE     * A2732790
*       FILE HAS BEEN DUPLICATED)                                     * A2732800
*                                                                     * A2732810
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732820
        EJECT                                                           A2732830
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732840
*                                                                     * A2732850
* PRINTER CATASTROPHIC ERROR MESSAGES (UPDATING IS STOPPED)           * A2732860
*              ( SEE 'END OF JOB' - 3RD CASE )                        * A2732870
*                                                                     * A2732880
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2732890
*                                                                     * A2732900
*    ***************************************                          * A2732910
*    * UP10W FIRST '/ UPDATE' CARD MISSING *                          * A2732920
*    ***************************************                          * A2732930
*     - THE FIRST CARD FROM UPDTCORR IS NOT A / UPDATE CARD           * A2732940
*                                                                     * A2732950
*    ********************************************                     * A2732960
*    * UP11W THE FIRST OLD RECORD IS A TAPE MARK *                     * A2732970
*    ********************************************                     * A2732980
*     - UPDTOLD IS INCORRECT OR IS NOT PROPERLY POSITIONED            * A2732990
*                                                                     * A2733000
*    *********************************                                * A2733010
*    * UP21W PROGRAM CHECK,LOAD DUMP *                                * A2733020
```

```
*    ********************************                                  * A2733030
*     - THE UPDATING PROGRAM HAS BEEN ALTERED ON ITS MEDIA             * A2733040
*       (CARDS OR TAPE),OR DURING THE LOADING                          * A2733050
*                                                                      * A2733060
*                                                                      * A2733070
* IN THE ABOVE 3 CASES UPDATING MUST BE RESUMED AFTER CORRECTION       * A2733080
*                                                                      * A2733090
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733100
        EJECT                                                           A2733110
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733120
*                                                                      * A2733130
* PRINTER ERROR MESSAGES   - ' AT FILE LEVEL'                          * A2733140
*                  ( SEE 'END OF JOB' - 4TH CASE )                     * A2733150
*                                                                      * A2733160
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733170
*                                                                      * A2733180
* A ERROR  OR A DISCREPANCY HAS BEEN DETECTED DURING THE TREATMENT     * A2733190
*                       OF A FILE                                      * A2733200
* THE UP23I MESSAGE IS ACCOMPAGNED BY ONE OR BOTH TYPES                * A2733210
*                     OF THE FOLLOWING MESSAGES                        * A2733220
*                                                                      * A2733230
* 1  ***************************************************               * A2733240
*    * I DECREASING OR NON NUMERICAL NUMBER IN MODIF CARD *            * A2733250
*    * O NEW RECORD NUMBER OVERFLOW(SET TO 1)             *            * A2733260
*    * N INCORRECT OR OMITTED INITIAL NEW NUMBER          *            * A2733270
*    *         (SET TO PRECEDING NEW NUMBER+1)            *            * A2733280
*    * D DECREASING OR NON NUMERICAL NUMBER IN OLD RECORD *            * A2733290
*    *         (IS CORRECTED)                             *            * A2733300
*    * UP23I FILE TREATMENT FINISHED                      *            * A2733310
*    ***************************************************               * A2733320
*     - A DISCREPANCY HAS BEEN DETECTED IN THE NUMERIC PART OF A CARD  * A2733330
*       OR RECORD IDENTIFICATION                                       * A2733340
*     - THE CORRESPONDING LINE OF THE LISTING CONTAINS THE SAME        * A2733350
*       LETTER ( I,O,N OR D )                                          * A2733360
*     - ANALYZE THE DISCREPANCY AND,IF NECESSARY,RESUME UPDATING       * A2733370
*                                                                      * A2733380
*      I  THE DISCREPANCY HAS NOT AFFACT THE PROCESSING,BUT IT IS      * A2733390
*         RECOMMANDED TO CHECK THE ORDER OF THE MODIFICATION CARD.     * A2733400
*                                                                      * A2733410
*      O  IT IS RECOMMANDED TO RENUMBER THE MODULE USING A LOWER STEP  * A2733420
*         NUMBER                                                       * A2733430
*                                                                      * A2733440
*      N  THE PROGRAM HAS DETERMINATED THE NUMBER OF A NEW RECORD TO   * A2733450
*         BE WRITTEN IN A MODULE.                                      * A2733460
*           - THIS NUMBER IS                                           * A2733470
*                - EITHER-THAT OF AN OLD RECORD TO BE COPIED           * A2733480
*                - OR - THE INITIAL NUMBER REQUESTED BY AN RIS CARD    * A2733490
*                  FOR THE FIRST RECORD RESULTING FROM THE CORRECTION* A2733500
*                  TO BE MADE.                                         * A2733510
*           - AND THIS NUMBER EXCEEDS NOT THE NUMBER OF THE PRECEDING  * A2733520
*             RECORD.                                                  * A2733530
*           - THE PROGRAM THEN ASSIGNS THE PRECEDING NUMBER+1 TO THE   * A2733540
*             RECORD TO BE WRITTEN                                     * A2733550
*                                                                      * A2733560
*      D  THE PROGRAM ASSIGNS THE PRECEDING NUMBER+1 TO THE RECORD TO  * A2733570
```

```
*         BE WRITTEN                                                    * A2733580
*                                                                       * A2733590
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733600
         EJECT                                                           A2733610
* * * * * * * * * * * *  CONTINUED  * * * * * * * * * * * * * * * * * * A2733620
*                                                                       * A2733630
* 2  ************************************                               * A2733640
*    * UP22I ERRORS DETECTED IN THIS FILE *                            * A2733650
*    * UP23I FILE TREATMENT FINISHED      *                            * A2733660
*    ************************************                               * A2733670
*     - AN ERROR HAS BEEN DETECTED IN THE UPDTCORR CARDS                * A2733680
*     - ONE OF THE REQUESTED FUNCTIONS COULD NOT BE EXECUTED            * A2733690
*     - UPDATING MUST PROBABLY BE RESUMED                               * A2733700
*     - THE ERROR ITSELF IS INDICATED DURING THE LISTING OF THE FILE    * A2733710
*                                                                       * A2733720
*     ( REFER TO ' ERROR MESSAGES - LEVEL OF CORRECTIONS' )             * A2733730
*                                                                       * A2733740
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733750
         EJECT                                                           A2733760
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733770
*                                                                       * A2733780
* PRINTER ERROR MESSAGES   - LEVEL OF CORRECTIONS                       * A2733790
*               ( SEE  ERROR MESSAGES'AT FILE LEVEL' - 2ND CASE)        * A2733800
*                                                                       * A2733810
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2733820
*                                                                       * A2733830
*    ********************************                                   * A2733840
*    * UP16I FIRST RIS CARD MISSING *                                   * A2733850
*    ********************************                                   * A2733860
*     - AFTER A / UPDATE CARD THE PROGRAM HAS READ A MODIFICATION CARD* A2733870
*       (OTHER THAN / UPDATE OR RIS CARD)                               * A2733880
*     - THE OLD FILE CORRESPONDING TO THE / UPDATE CARD WILL BE COPIED* A2733890
*     - THE MODIFICATION AND RIS CARDS FOLLOWING THIS / UPDATE CARD     * A2733900
*       WILL BE PRINTED WITHOUT INVOLVING CORRECTIONS.                  * A2733910
*                                                                       * A2733920
*    ********************************************************************* A2733930
*    * UP18I THE FUNCTION SPECIFY IN THIS RIS CARD CANNOT BE PERFOR-* * A2733940
*    *       MED,AS THE EXPECTED OLD RECORD HAS NOT BEEN FOUND      * * A2733950
*    *  RIS (CARD-IMAGE)                                            * * A2733960
*    ********************************************************************* A2733970
*                                                                       * A2733980
*     - THE PROGRAM HAS READ                                            * A2733990
*       - EITHER - AN RIS CARD REQUESTING AN INSERTION AT THE           * A2734000
*                  BEGINNING OF A FILE OR OF A MODULE WHOSE FIRST RECORD * A2734010
*                  HAS BEEN ALREADY PROCESSED                           * A2734020
*       - OR - AN RIS CARD RELATED TO A MODULE OF AN OLD FILE,AFTER     * A2734030
*                  READING OF ALL THE MODULES OF THIS FILE.             * A2734040
*       - OR - AN RIS CARD (RECORDS PROCESSING MODE) RELATED TO A       * A2734050
*                  RECORD OF AN OLD MODULE,AFTER READING OF ALL THE     * A2734060
*                  RECORDS OF THIS MODULE.                              * A2734070
*                                                                       * A2734080
*     - ALL THE MODIFICATION AND RIS CARDS RELATED TO THE SAME MODULE * A2734090
*                  WILL BE PRINTED WITHOUT INVOLVING CORRECTIONS.       * A2734100
*                                                                       * A2734110
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2734120
```

```
        EJECT                                                          A2734130
* * * * * * * * * * * * *  CONTINUED  * * * * * * * * * * * * * * * * * A2734140
*                                                                      * A2734150
*     ****************************************                         * A2734160
*     * UP20I END OF CARDS SET,NOT THAT EXPECTED- *                    * A2734170
*     *       FUNCTION INTERRUPTED               *                     * A2734180
*     ****************************************                         * A2734190
*     - 1ST CASE - DURING AN INSERTION OR A REPLACING                   * A2734200
*       - EITHER - THERE IS NOT A MODIFICATION CARD AFTER A RIS CARD    * A2734210
*       - OR - THE FIRST MODIFICATION CARD DOES NOT CONTAIN THE SAME    * A2734220
*              MODULE IDENTIFICATION AS THAT OF THE RIS CARD (RECORDS   * A2734230
*              PROCESSING MODE)                                        * A2734240
*       - OR - A MODIFICATION CARD DOES NOT CONTAIN THE SAME IDENTI-    * A2734250
*              FICATION THAT OF THE PRECEDING MODIFICATION CARD         * A2734260
*              (BEWARE OF BLANK CARDS...)                              * A2734270
*                                                                      * A2734280
*     - 2ND CASE - AFTER THE EXECUTION OF A CORRECTION                 * A2734290
*       - EITHER - A MODIFICATION CARD FOLLOWS AN RIS CARD WHICH       * A2734300
*              REQUESTS A SUPPRESSION OR A NUMBERING                   * A2734310
*       - OR - THE PROCESSED RIS CARD(MODULE PROCESSING MODE) IS       * A2734320
*              FOLLOWED BY A RIS CARD REQUESTING AN OTHER CORRECTION   * A2734330
*              ON THE SAME MODULE.                                     * A2734340
*                                                                      * A2734350
*     - IN ALL  ABOVE CASE                                             * A2734360
*       - THE REMAINDER OF THE PROCESSED OLD MODULE WILL BE            * A2734370
*         COPIED ON THE NEW TAPE                                       * A2734380
*       - THE RIS AND MODIFICATIONS CARDS RELATED TO THIS MODULE WILL  * A2734390
*         BE PRINTED WITHOUT INVOLVING CORRECTIONS                     * A2734400
*                                                                      * A2734410
*     ****************************************                         * A2734420
*     * UP17I INVALID RIS CARD,CORRECTION IGNORED *                    * A2734430
*     *  RIS (CARD IMAGE)                        *                     * A2734440
*     ****************************************                         * A2734450
*     - AN RIS CARD IS INVALID IF                                      * A2734460
*       - EITHER - BOTH MODULE IDENTIFICATIONS CONTAINED IN  A RIS     * A2734470
*              CARD (RECORDS PROCESSING MODE) ARE DIFFERENT            * A2734480
*       - OR - THE NUMBER OF THE 2ND IDENTIFICATION FIELD IS LESS      * A2734490
*              THAN THE NUMBER OF THE 1ST ONE                          * A2734500
*       - OR - ONE OF THE FOLLOWING FIELDS IS NOT ENTIRELY NUMERICAL   * A2734510
*              OR BLANK                                                * A2734520
*                 -NUMBER IN THE 1ST IDENTIFICATION FIELD              * A2734530
*                 -NUMBER IN THE 2ND IDENTIFICATION FIELD              * A2734540
*                 -INITIAL NUMBER                                      * A2734550
*                 -NUMBERING STEP                                      * A2734560
*       - OR - THE 'FUNCTION REQUEST CODE' IS DIFFERENT FROM R,I,S,N   * A2734570
*                                                                      * A2734580
*     IN ALL ABOVE CASES THE RIS AND MODIFICATION CARDS RELATED TO     * A2734590
*     THE SAME MODULE AS THAT OF THE INVALID RIS CARD WILL BE PRINTED  * A2734600
*     WITHOUT INVOLVING CORRECTIONS                                    * A2734610
*                                                                      * A2734620
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * A2734630
        EJECT                                                          A2734640
*                                                                      A2734650
* ERROR CODES AND CORRESPONDING MESSAGES          -        PRINTER     A2734660
*                                                                      * A2734670
```

```
COD      EQU   2                 CODE LENGTH          BI,BO          A2734680
*                    STORAGE OF ERRORS CODES                         A2734690
ERTBG     DS   16C               * X X X X X D N O*                  A2734700
          DS   2C                * I*                                A2734710
*                  MESSAGES                                          A2734720
MSGTBL    DC   C' I '                                                A2734730
          DC   C'DECREASING OR NO'                                   A2734740
          DC   C'N NUMERICAL NUMB'                                   A2734750
          DC   C'ER IN MODIF CARD'                                   A2734760
          DC   C'                '                                   A2734770
          DC   C'                '                                   A2734780
MSGTBM    DC   C' O '                                                A2734790
          DC   C'NEW RECORD NUMBE'                                   A2734800
          DC   C'R OVERFLOW      '                                   A2734810
          DC   C'(SET TO 1)      '                                   A2734820
          DC   C'                '                                   A2734830
          DC   C'                '                                   A2734840
MSGTBN    DC   C' N '                                                A2734850
          DC   C'INCORRECT OR OMI'                                   A2734860
          DC   C'TTED INITIAL NEW'                                   A2734870
          DC   C' NUMBER(SET TO P'                                   A2734880
          DC   C'RECEDING NEW NUM'                                   A2734890
          DC   C'BER +1)         '                                   A2734900
MSGTBP    DC   C' D '                                                A2734910
          DC   C'DECREASING OR NO'                                   A2734920
          DC   C'N NUMERICAL NUMB'                                   A2734930
          DC   C'ER IN OLD RECORD'                                   A2734940
          DC   C'(IS CORRECTED)  '                                   A2734950
          DC   C'                '                                   A2734960
          EJECT                                                      A2734970
*      'UXNNY TEXT'                                                  A2734980
SIMCOD    DC   C'A2'                        BEFORE CONSOLE MESSAGE   A2734990
UPDCOD    DC   C'UP'                        BEFORE PRINTER MESSAGE   A2735000
*                                                                    A2735010
*  PRINTER               INFORMATIVE MESSAGES                        A2735020
*                                                                    A2735030
MESS03    DC   FL1'39'                                               A2735040
          DC   C'24I THE NEXT CS,'                                   A2735050
          DC   C'XXXXXXXX,WILL BE'                                   A2735060
          DC   C' COPIED'                                            A2735070
MESS22    DC   FL1'18'                                               A2735080
          DC   C'19I INSERT REQUE'                                   A2735090
          DC   C'ST'                                                 A2735100
MESS23    DC   FL1'19'                                               A2735110
          DC   C'19I REPLACE REQU'                                   A2735120
          DC   C'EST'                                                A2735130
MESS24    DC   FL1'20'                                               A2735140
          DC   C'19I SUPPRESS REQ'                                   A2735150
          DC   C'UEST'                                               A2735160
MESS25    DC   FL1'21'                                               A2735170
          DC   C'19I NUMBERING RE'                                   A2735180
          DC   C'QUEST'                                              A2735190
MESS27    DC   FL1'27'                                               A2735200
          DC   C'23I FILE TREATME'                                   A2735210
          DC   C'NT FINISHED'                                        A2735220
```

```
MESS29   DC    FL1'41'                                                  A2735230
         DC    C'14I XXX ALL CS  '                                      A2735240
         DC    C'LISTED-1ST CS IS'                                      A2735250
         DC    C' XXXXXXXX'                                             A2735260
MESS30   DC    FL1'58'                                                  A2735270
         DC    C'12I THE NEXT FIL'                                      A2735280
         DC    C'E WILL BE DELETE'                                      A2735290
         DC    C'D.ITS FIRST CS I'                                      A2735300
         DC    C'S XXXXXXXX'                                            A2735310
MESS32   DC    FL1'35'                                                  A2735320
         DC    C'15I ALL OLD FILE'                                      A2735330
         DC    C'S HAVE BEEN TREA'                                      A2735340
         DC    C'TED'                                                   A2735350
MESS33   DC    FL1'32'                                                  A2735360
         DC    C'13I THE NEXT FIL'                                      A2735370
         DC    C'E WILL BE COPIED'                                      A2735380
         EJECT                                                          A2735390
*                                                                       A2735400
*  PRINTER                    ERROR MESSAGES  (CONTINUE)                A2735410
*                                                                       A2735420
ERME04   DC    FL1'26'                                                  A2735430
         DC    C'16I FIRST RIS CA'                                      A2735440
         DC    C'RD MISSING'                                            A2735450
ERME06   DC    FL1'39'                                                  A2735460
         DC    C'17I'                                                   A2735470
         DC    C' '                                                     A2735480
         DC    C'INVALID RIS CARD'                                      A2735490
         DC    C',CORRECTION IGNO'                                      A2735500
         DC    C'RED'                                                   A2735510
ERME07   DC    FL1'103'                                                 A2735520
         DC    C'18I THE FUNCTION'                                      A2735530
         DC    C' SPECIFY IN THIS'                                      A2735540
         DC    C' RIS CARD CANNOT'                                      A2735550
         DC    C' BE PERFORMED,AS'                                      A2735560
         DC    C' EXPECTED OLD RE'                                      A2735570
         DC    C'CORD HAS NOT BEE'                                      A2735580
         DC    C'N FOUND'                                               A2735590
ERME08   EQU   ERME07                                                   A2735600
ERME09   EQU   ERME07                                                   A2735610
ERME10   EQU   ERME07                                                   A2735620
ERME12   DC    FL1'59'                                                  A2735630
         DC    C'20I END OF CARDS'                                      A2735640
         DC    C' SET,NOT THAT EX'                                      A2735650
         DC    C'PECTED.FUNCTION '                                      A2735660
         DC    C'INTERRUPTED'                                           A2735670
ERME14   EQU   ERME12                                                   A2735680
ERME26   DC    FL1'32'                                                  A2735690
         DC    C'22I ERRORS DETEC'                                      A2735700
         DC    C'TED IN THIS FILE'                                      A2735710
         EJECT                                                          A2735720
*                                                                       A2735730
*  PRINTER                    ERROR MESSAGES  (STOP)                    A2735740
*                                                                       A2735750
ERME02   DC    FL1'29'                                                  A2735760
         DC    C'10W 1ST / UPDATE'                                      A2735770
```

```
         DC    C' CARD MISSING'                                          A2735780
ERME05   DC    FL1'31'                                                   A2735790
         DC    C'11W 1ST OLD RECO'                                       A2735800
         DC    C'RD IS TAPE-MARK'                                        A2735810
ERME21   DC    FL1'27'                                                   A2735820
         DC    C'21W PROGRAM CHEC'                                       A2735830
         DC    C'K.LOAD DUMP'                                            A2735840
         EJECT                                                           A2735850
*                                                                        A2735860
*  PRINTER AND CONSOLE INFORMATIVE MESSAGES                             A2735870
*                                                                        A2735880
MES534   DC    FL1'20'                                                   A2735890
         DC    C'90A END OF UPDAT'                                       A2735900
         DC    C'ING'                                                    A2735910
         DC    X'15'                                                     A2735920
         EJECT                                                           A2735930
*                                                                        A2735940
*  PRINTER AND CONSOLE       ERROR MESSAGES  (STOP)                     A2735950
*                                                                        A2735960
ERME28   DC    FL1'35'                                                   A2735970
         DC    C'92W UPDATING ERR'                                       A2735980
         DC    C'OR,CANNOT CONTIN'                                       A2735990
         DC    C'UE'                                                     A2736000
         DC    X'15'                                                     A2736010
ERME37   DC    FL1'37'                                                   A2736020
         DC    C'93W POSSIBLE UPD'                                       A2736030
         DC    C'ATING ERROR DETE'                                       A2736040
         DC    C'CTED'                                                   A2736050
         DC    X'15'                                                     A2736060
         EJECT                                                           A2736070
*                                                                        A2736080
*              CONSOLE       ERROR MESSAGES  (STOP)                     A2736090
*                                                                        A2736100
ERME18   DC    FL1'61'                                                   A2736110
         DC    C'91W CONTROL CARD'                                       A2736120
         DC    C' ERROR,XXXXXXXX '                                       A2736130
         DC    C'NOT ASSIGNED,CAN'                                       A2736140
         DC    C'NOT CONTINUE'                                           A2736150
         DC    X'15'                                                     A2736160
         EJECT                                                           A2736170
         END                                                             A2736180
         AOPTN CROSSREF                                                  A2E00010
A2EB     TITLE ' EDITOR V-1,L-2 '                                  V1L2  A2E00015
EDITOR   START 0                                                         A2E00020
*************************************************************************A2E00025
*                                                                       *A2E00030
*                            SIM20 EDITOR                               *A2E00035
*                            INTRODUCTION                               *A2E00040
*                                                                       *A2E00045
*  THE SIM20 EDITOR IS A PROGRAM WHICH, FOR A GIVEN SYSTEM/360 CONFI-   *A2E00050
*  GURATION AND A GIVEN 1620 CONFIGURATION, EDITS THE 1620 SIMULATOR    *A2E00055
*  TO INCLUDE ONLY THE ROUTINES REQUIRED TO SIMULATE THE  DESIGNATED    *A2E00060
*  1620 CONFIGURATION.                                                  *A2E00065
*                                                                       *A2E00070
*  THE EDITOR TAKES INTO ACCOUNT THE 1620 CONFIGURATION TO BE  SIMU-    *A2E00075
```

```
*  LATED, THE SYSTEM/360 CONFIGURATION ON WHICH THE SIMULATOR  IS TO  * A2E00080
*  BE RUN, AS WELL AS ALL THE FEATURES AND DEVICES REQUIRED  BY  THE  * A2E00085
*  SIMULATED 1620.                                                    * A2E00090
*                                                                     * A2E00095
*  USING CONTROL CARDS, THE EDITOR SELECTS FROM THE SIMULATOR   ONLY  * A2E00100
*  THOSE ROUTINES WHICH ARE NEEDED TO SIMULATE THE GIVEN 1620 CONFIG- * A2E00105
*  URATION.                                                           * A2E00110
*  IT EDITS THE SYMBOLIC VERSION OF THE SIMULATOR. THEREFORE,  IT IS  * A2E00115
*  NECESSARY TO ASSEMBLE THE EDITED PROGRAM.                          * A2E00120
*  THE SIMULATOR CAN BE ON CARDS OR ON PAPER TAPE,  AND  THE  EDITED  * A2E00125
*  PROGRAM WILL BE ON THE SAME MEDIUM AS THE ORIGINAL SIMULATOR. THE  * A2E00130
*  EDITOR IS ALWAYS ON CARDS AND IS RELOCATABLE.                      * A2E00135
*                                                                     * A2E00140
*                            CONTROL CARDS                            * A2E00145
*  THERE ARE FIVE DIFFERENT TYPES OF CONTROL CARDS, DEFINED AS  FOL-  * A2E00150
*  LOWS =                                                             * A2E00155
*  1) CPU1                                                            * A2E00160
*    THIS CONTROL CARD DEFINES THE MODEL AND THE CORE STORAGE CAPAC-  * A2E00165
*    ITY OF THE 1620 TO BE SIMULATED. ONLY ONE CPU1 CARD IS REQUIRED  * A2E00170
*    FOR EACH EDITOR RUN. IF THERE ARE SEVERAL CARDS, ONLY THE  LAST  * A2E00175
*    ONE ENCOUNTERED WILL BE TAKEN INTO CONSIDERATION.                * A2E00180
*                                                                     * A2E00185
*    FORMAT =                                                         * A2E00190
*           COLUMNS 3 TO 17 CONTAIN = CPU1 1620/X,YYK                 * A2E00195
*               WHERE  X = 1 OR 2 - 1620 MODEL TO BE SIMULATED        * A2E00200
*                      YY = 20, 40 OR 60 - 1620 CORE STORAGE          * A2E00205
*                           (IN THOUSANDS OF POSITIONS)               * A2E00210
*                                                                     * A2E00215
*********************************************************************** A2E00220
        EJECT                                                          A2E00225
*********************************************************************** A2E00230
*                                                                     * A2E00235
*  2) CPU2                                                            * A2E00240
*    THIS CONTROL CARD DEFINES THE MODEL AND THE MAIN STORAGE CAPAC-  * A2E00245
*    ITY OF THE SYSTEM/360 TO BE USED. ONLY ONE CPU2 CARD IS REQUIRED * A2E00250
*    FOR EACH EDITOR RUN. IF THERE ARE SEVERAL CARDS, ONLY THE  LAST  * A2E00255
*    ONE ENCOUNTERED WILL BE TAKEN INTO CONSIDERATION.                * A2E00260
*                                                                     * A2E00265
*                                                                     * A2E00270
*    FORMAT =                                                         * A2E00275
*           COLUMNS 3 TO 17 CONTAIN = CPU2 360/XX,YYK OR              * A2E00280
*           COLUMNS 3 TO 18 CONTAIN = CPU2 360/XX,YYYK                * A2E00285
*               WHERE XX = 30 OR 40 - SYSTEM/360 MODEL USED           * A2E00290
*                     YY = 32, 64 - SYSTEM/360 MAIN STORAGE CAPACITY  * A2E00295
*                    YYY = 128 AND ABOVE - SYSTEM/360 MAIN STORAGE    * A2E00300
*                          CAPACITY                                   * A2E00305
*  3) FEATURE                                                         * A2E00310
*    THIS CONTROL CARD IS NOT REQUIRED IF THE PROGRAM TO  BE  EDITED  * A2E00315
*    DOES NOT USE ANY NON-STANDARD FEATURES. IF SUCH FEATURES ARE US- * A2E00320
*    ED, EACH ONE REQUIRES A CARD.                                    * A2E00325
*                                                                     * A2E00330
*    FORMAT =                                                         * A2E00335
*           COLUMNS 3 TO 9 OF EACH CARD CONTAIN = FEATURE             * A2E00340
*           COLUMN 10 CONTAINS A BLANK                                * A2E00341
*            COLUMNS 11 TO 15 DEPEND ON THE FEATURE USED, I.E.        * A2E00345
```

```
*               INDEX = 1620 MODEL 2 INDEX REGISTERS                  * A2E00350
*               INDAD = 1620 MODEL 1 INDIRECT ADDRESSING              * A2E00355
*               FLOAT = FLOATING-POINT ARITHMETIC                     * A2E00360
*               TRANS = ADDITIONAL INSTRUCTIONS                       * A2E00365
*               DIVID = AUTOMATIC DIVIDE                              * A2E00370
*                DISKV = I/O PROGRAM MUST BE DISK-RESIDENT (FOR IN-   * A2E00375
*                       STANCE MODEL E WITH ALL DEVICES)              * A2E00380
*               144LN = FOR A 1443 WITH 144 POSITIONS OF PRINT        * A2E00385
*                                                                     * A2E00390
*                                                                     * A2E00395
********************************************************************** A2E00400
        EJECT                                                          A2E00405
********************************************************************** A2E00410
*                                                                     * A2E00415
*  4) DEVICE                                                          * A2E00420
*      THIS CONTROL CARD IS REQUIRED FOR EACH DEVICE  USED BY 1620.   * A2E00425
*    EACH CONTROL CARD IS USED TO CREATE CONTROL BLOCKS FOR CONTROL   * A2E00430
*    PROGRAM.                                                         * A2E00435
*             TWO VERIFICATIONS ARE MADE =                            * A2E00440
*                1) IF TWO DEVICES ARE SAME DEVICE ADDRESSES THE 2ND  * A2E00445
*                   IS IGNORED                                        * A2E00450
*                2) THE TYPE OF DEVICE/360 ARE CHECKED     EX = 2540R * A2E00455
*            NO VERIFICATION ARE MADE ON CHANNEL , DEVICE ADDRESS     * A2E00460
*            OF EACH DEVICE                                           * A2E00465
*            IF AN ERROR OCCURS SIMULATOR WILL BE NOT AVALAIBLE       * A2E00470
*                                                                     * A2E00475
*    FORMAT =                                                         * A2E00480
*           COLUMNS 3 TO 8 = DEVICE                                   * A2E00485
*           COLUMN 9 CONTAINS A BLANK                                 * A2E00486
*            COLUMNS 10 TO 15 CONTAIN THE 1620 DEVICE (I.E.           * A2E00490
*               1622R, = 1622 CARD READER                             * A2E00495
*               1622P, = 1622 CARD PUNCH                              * A2E00500
*               1443W, = 1443 PRINTER                                 * A2E00505
*               1621R, = 1621 PAPER TAPE READER                       * A2E00510
*               1624P = 1624 PAPER TAPE PUNCH                         * A2E00515
*               1311N, = 1311 DISK STORAGE DRIVE                      * A2E00520
*               1620C, = 1052 TYPEWRITER                              * A2E00525
*                                                                     * A2E00530
*            COLUMNS 16 TO 29 CONTAIN THE SYSTEM/360 DEVICE, THE CON-* A2E00535
*            SOLE AND THE STANDARD ADDRESS USED BY THE SIMULATOR      * A2E00540
*     EX =      1442R,0,X'00A' = 1442 CARD READER                     * A2E00545
*               2540R,0,X'00C' = 2540 CARD READER                     * A2E00550
*               2520R,0,X'00C' = 2520 CARD READER                     * A2E00555
*               2501R,0,X'00C' = 2501 CARD READER                     * A2E00560
*               1442P,0,X'00A' = 1442 CARD PUNCH                      * A2E00565
*               2540P,0,X'00D' = 2540 CARD PUNCH                      * A2E00570
*               2520P,0,X'00D' = 2520 CARD PUNCH                      * A2E00575
*  THESE THREE LAST DEVICES ARE ALSO USED FOR A PAPER TAPE PUNCH IF    * A2E00580
*  CARD PUNCH DEVICE DOES NOT EXIST.                                  * A2E00585
*               1443W,0,X'00B' = 1443 PRINTER                         * A2E00590
*               1403W,0,X'00E' = 1403 PRINTER                         * A2E00595
*               2671R,0,X'004' = 2671 PAPER TAPE READER               * A2E00600
*               ....P,.,X'...' = PAPER TAPE PUNCH .....UNDETERMINED   * A2E00605
*               2311N,C,X'CMM' = 2311 DISK STORAGE DRIVE              * A2E00610
*               1052T,0,X'009' = 1052 TYPEWRITER                      * A2E00615
```

```
*                                                                     * A2E00620
*  NOTE = IF CARD READER/PUNCH IS PRESENT AND IF PAPER TAPE (VERSION  * A2E00621
*         CARD) IS NEEDED,THIS NEXT CARD IS USED  =                   * A2E00622
*                      / DEVICE 1621P                                 * A2E00623
*         /            IN COLUMN 1                                    * A2E00624
*         BLANK        IN COLUMN 2                                    * A2E00625
*         DEVICE       IN COLUMNS 3 TO 8                              * A2E00626
*         BLANK        IN COLUMN 9                                    * A2E00627
*         1621P        IN COLUMNS 10 TO 14                            * A2E00628
*         BLANKS       IN OTHER COLUMNS                               * A2E00629
*                                                                     * A2E00630
*********************************************************************** A2E00631
         EJECT                                                        A2E00632
*********************************************************************** A2E00635
*                                                                     * A2E00640
*  5) START                                                           * A2E00645
*    THIS CONTROL CARD IS ALWAYS REQUIRED AND IS THE LAST ONE.        * A2E00650
*                                                                     * A2E00655
*    FORMAT =                                                         * A2E00660
*              COLUMNS 3 TO 7 CONTAIN START                           * A2E00665
*                                                                     * A2E00670
*  ALL CONTROL CARD DECKS MUST RESPECT THE ABOVE LISTED ORDER OF CARD * A2E00675
*  TYPE. NO GIVEN CARD ORDER NEED BE RESPECTED WITHIN  ONE  TYPE  OF  * A2E00680
*  CARDS.                                                             * A2E00685
*  EXAMPLE = THE FIRST CARD MUST BE A CPU1 CARD, FOLLOWED  BY  CPU2,  * A2E00690
*            FEATURE, DEVICE, AND START CARDS, IN THAT ORDER.         * A2E00695
*                                                                     * A2E00700
*********************************************************************** A2E00705
         EJECT                                                        A2E00710
*                                                                     A2E00715
*              WORKING REGISTERS                                      A2E00720
*              EQUIVALENCE FOR CODE CONDITION OF BRANCHES             A2E00725
*                                                                     A2E00730
*                                                                     A2E00735
         SPACE 2                                                      A2E00740
         SPACE 2                                                      A2E00745
BE       EQU   8                                                      A2E00750
BH       EQU   2                                                      A2E00755
B        EQU   15                                                     A2E00760
BR       EQU   15                                                     A2E00765
BO       EQU   1                                                      A2E00770
BZ       EQU   8                                                      A2E00775
BL       EQU   4                                                      A2E00780
BNE      EQU   7                                                      A2E00785
NOP      EQU   0                                                      A2E00790
RZ       EQU   2                                                      A2E00795
R1       EQU   1                                                      A2E00800
R3       EQU   3                                                V1L2  A2E00802
WR1      EQU   10                                                     A2E00805
WR2      EQU   11                                                     A2E00810
WR3      EQU   13                                                     A2E00815
WR4      EQU   14                                                     A2E00820
WR5      EQU   5                                                      A2E00825
         EJECT                                                        A2E00830
BEGIN    BALR  7,0                                                    A2E00835
```

```
         USING *,7                                                      A2E00840
         USING *+4096,8                                                 A2E00845
         LA    8,2048                                                   A2E00850
         AR    8,8                                                      A2E00855
         LA    8,0(8,7)                                                 A2E00860
*                                                                       A2E00865
*                                                                       A2E00870
*              PROGRAM INITIALIZATION                                   A2E00875
*                                                                       A2E00880
RETRY    MVI   DEVTAB,X'FF'                                             A2E00885
         MVI   DEVSHT,X'FF'                                             A2E00890
         MVI   FEATAB,X'FF'                                             A2E00895
         MVI   IOBUFF,X'00'                                             A2E00900
         MVC   IOBUFF+1(120),IOBUFF                                     A2E00905
         MVI   CP1A,X'00'                                               A2E00910
         MVC   CP1A+1(11),CP1A                                          A2E00915
         EJECT                                                          A2E00920
*                                                                       A2E00925
*                                                                       A2E00930
*      READ CONTROL INFORMATION                                         A2E00935
*                                                                       A2E00940
*                                                                       A2E00945
*                                                                       A2E00950
READ1    MVI   IOBUFF-2,X'40'                                    V1L1 A2E00955
         MVC   IOBUFF-1(80),IOBUFF-2                             V1L1 A2E00956
         BAL   15,READ                 SEND A READ COMMAND       V1L1 A2E00957
         CLC   IOBUFF-2(2),CTN         COMPARE IDENTIFICATION          A2E00960
         BC    7,ER1A                  NOT RIGHT BRANCH                A2E00965
*                                                                       A2E00970
*              COMPARE CONTROL INFORMATION WITH DIFFERENT TABLES         A2E00975
*                                                                       A2E00980
*                                                                       A2E00985
*                                                                       A2E00990
         LA    WR2,CPU1                LOAD FIRST TABLE                A2E00995
         LA    WR1,15                                                   A2E01000
         BAL   15,COMPAR               COMPARE CONTROL CARD            A2E01005
         BC    15,CP1                  IT IS A CPU1 CARD               A2E01010
         LA    WR2,CPU2                LOAD SECOND TABLE               A2E01015
         LA    WR1,16                                                   A2E01020
         BAL   15,COMPAR               COMPARE CONTROL CARD            A2E01025
         BC    15,CP2                  IT IS A CPU2 CARD               A2E01030
         LA    WR2,FEAT                LOAD THIRD TABLE                A2E01035
         LA    WR1,13                                                   A2E01040
         BAL   15,COMPAR               COMPARE CONTROL CARD            A2E01045
         BC    15,FEAT1                IT IS A FEATURE CARD            A2E01050
         LA    WR2,DEVICE              LOAD FOURTH TABLE               A2E01055
         LA    WR1,19                                                   A2E01060
         BAL   15,COMPAR               COMPARE CONTROL CARD            A2E01065
         BC    15,DEV1                 IT IS A DEVICE CARD             A2E01070
         LA    WR2,START               LOAD FIFTH TABLE                A2E01075
         LA    WR1,5                                                    A2E01080
         BAL   15,COMPAR               COMPARE CONTROL CARD            A2E01085
         BC    15,START1               IT IS A START CARD              A2E01090
ER1A     BAL   15,MESSAG                                               A2E01095
         DC    YL2(PASTR)              SEND MESSAGE INV. CARD          A2E01100
```

```
         BAL   15,MESSAG                                        V1L1 A2E01105
         DC    Y(NCONTI)                                        V1L1 A2E01110
         BC    15,WAI12              WAIT STATE                 V1L1 A2E01115
CTN      DC    C'/ '                                                 A2E01116
         EJECT                                                       A2E01117
*                                                                    A2E01120
*              CONTROL INFORMATION IS 'START'                        A2E01125
*              TREATMENT BEGINS                                      A2E01130
*                                                                    A2E01135
START1   SR    WR1,WR1                                               A2E01140
         IC    WR1,CP2A+1                                            A2E01145
         BCTR  WR1,0                                                 A2E01150
         STC   WR1,CP2A+1                                            A2E01155
         IC    WR1,CP2A+2                                            A2E01160
         BCTR  WR1,0                                                 A2E01165
         BCTR  WR1,0                                                 A2E01170
         STC   WR1,CP2A+2            IS 1620 CPU SIZE HIGHER       A2E01175
         CLC   CP2B(3),CP2A          THAN 360 CPU SIZE - 12K       A2E01180
         BC    BH,CPER               YES,BRANCH                    A2E01185
         MVC   CBIT(1),CP1A+5        MOVE MODEL                    A2E01190
         LA    WR1,DEVSHT            SEARCH FOR '13110' DEVICE     A2E01195
AAA33    CLI   0(WR1),X'FF'          IS TABLE EXHAUSTED            A2E01200
         BC    BE,AAA55              YES,BRANCH                    A2E01205
         CLC   0(5,WR1),A13110       COMPARE IF DISK ARE PRESENT   A2E01210
         BC    BE,AAA22              YES,BRANCH                    A2E01215
         LA    WR1,5(WR1)            NO,LOOP                       A2E01220
         BC    15,AAA33              *                             A2E01225
AAA22    LA    WR1,FEATAB            SEARCH FOR 'DISKV'            A2E01230
AAA44    CLI   0(WR1),X'FF'          IS TABLE EXHAUSTED            A2E01235
         BC    8,START3              YES,BRANCH                    A2E01236
         CLC   0(5,WR1),DISKV        IS 'DISKV' PRESENT            A2E01240
         BC    BE,SEARCH             YES,BRANCH                    A2E01245
         LA    WR1,5(WR1)            NO,INCREMENT WR1 BY FIVE      A2E01250
         BC    15,AAA44              LOOP                          A2E01255
AAA55    MVC   0(5,WR1),NODSK        MOVE NODISK                   A2E01260
         MVI   5(WR1),X'FF'          SET END OF TABLE              A2E01265
         BC    15,SEARCH             BRANCH                        A2E01270
START3   SR    WR4,WR4               WR4=0                         A2E01275
         SR    WR5,WR5               WR5=0                         A2E01276
         LA    WR2,DEVSHT            SEARCH FOR DEVSHT TABLE       A2E01277
EDX1A    LA    WR3,SEATB             WR3=A(SEATB)                  A2E01278
         CLI   0(WR2),X'FF'          IS TABLE EXHAUSTED            A2E01279
         BC    8,EDX3A               YES,BRANCH                    A2E01280
EDX2A    CLC   0(5,WR2),0(WR3)       COMPARE ARG WITH TABLE        A2E01281
         BC    8,ADDTAB              EQUAL BRANCH                  A2E01282
         LA    WR3,6(WR3)            INCREMENT WR3 BY 6            A2E01283
         BC    15,EDX2A              LOOP                          A2E01284
ADDTAB   IC    WR4,5(WR3)            INSERT INCREMENT IN WR4       A2E01285
         AR    WR5,WR4               ADD WR5 AND WR4               A2E01286
         LA    WR2,5(WR2)            INCREMENT WR2 BY 5            A2E01287
         BC    15,EDX1A              BRANCH                        A2E01288
EDX3A    LA    WR2,FEATAB            LOAD WR2 WITH FEATAB          A2E01289
EDX6A    LA    WR3,SEATB             LOAD WR3 WITH SEATB           A2E01290
         CLI   0(WR2),X'FF'          IS TABLE EXHAUSTED            A2E01291
         BC    8,EDX4A               YES,BRANCH                    A2E01292
```

```
EDX5A    CLC   0(5,WR2),0(WR3)      COMPARE ARG WITH TABLE                A2E01293
         BC    8,ADTAB1             EQUAL BRANCH                          A2E01294
         LA    WR3,6(WR3)           INCREMENT WR3 BY 6                    A2E01295
         BC    15,EDX5A             LOOP                                  A2E01296
ADTAB1   IC    WR4,5(WR3)           INSERT IN WR4 INCREMENT               A2E01297
         AR    WR5,WR4              ADD WR4 AND WR5                       A2E01298
         LA    WR2,5(WR2)           INCREMENT WR2 BY 5                    A2E01299
         BC    15,EDX6A             LOOP                                  A2E01300
EDX4A    LA    WR4,483              WR4 = 483 MAXIMUM FOR A 32K           A2E01301
         CR    WR5,WR4              IS WR5 HIGHER THAN LIMIT              A2E01302
         BC    BH,EDX7A             YES,BRANCH                            A2E01303
         MVC   0(5,WR1),NASKV       MOVE NASKV                            A2E01304
         MVI   5(WR1),X'FF'         MOVE END OF TABLE                     A2E01305
         BC    15,SEARCH            BRANCH -CONTINUE                      A2E01306
EDX7A    CLC   CP2A(3),K32          TEST FOR 32K                    V1L2  A2E01307
         BC    8,EDX7B                                              V1L2  A2E01308
*EDX7A   MVC   0(5,WR1),NISKV       MOVE NISKV          (DELETED)   V1L2  A2E01309
         MVC   0(5,WR1),NISKV       MOVE NISKV                      V1L2  A2E01310
         MVI   5(WR1),X'FF'         MOVE END OF TABLE                     A2E01311
         BC    15,SEARCH            BRANCH - CONTINUE                     A2E01312
EDX7B    BAL   15,MESSAG            SEND MESSAGE                    V1L2  A2E01313
         DC    YL2(NODSKV)          DISKV NEEDED                    V1L2  A2E01314
         BAL   R1,READTY            READ COMMAND                    V1L2  A2E01315
         BC    B,RETRY              RETRY                           V1L2  A2E01316
*                                                                   V1L2  A2E01317
K32      DC    C'020'               CP2A EQUAL 32K-12K              V1L2  A2E01318
NODSKV   DC    FL1'28'                                              V1L2  A2E01319
         DC    C' A243A '                                           V1L2  A2E01320
         DC    C'DISKV FEATURE '                                    V1L2  A2E01321
         DC    C'NEEDED'                                            V1L2  A2E01322
         DC    X'15'                                                V1L2  A2E01323
         EJECT                                                            A2E01324
NASKV    DC    C'NASKV'             *                                   A2E01325
SEATB    DC    C'INDAD'             TABLE FOR LIMIT OF A 32K            A2E01326
         DC    X'04'                LENGTH                              A2E01327
         DC    C'TRANS'             ADDITIONAL INSTRUCTIONS             A2E01328
         DC    X'15'                *                                   A2E01329
         DC    C'DIVID'             DIVIDE                              A2E01330
         DC    X'2B'                *                                   A2E01331
         DC    C'FLOAT'             FLOATING                            A2E01332
         DC    X'78'                *                                   A2E01333
         DC    C'INDEX'             INDEX INSTRUCTIONS                  A2E01334
         DC    X'3E'                *                                   A2E01335
         DC    C'1622R'             1622 CARD READER                    A2E01336
         DC    X'1E'                *                                   A2E01337
         DC    C'1622P'             1622 CARD PUNCH                     A2E01338
         DC    X'1E'                *                                   A2E01339
         DC    C'1443W'             1443 PRINTER                        A2E01340
         DC    X'41'                *                                   A2E01341
         DC    C'1621R'             1621 PAPER TAPE READER              A2E01342
         DC    X'21'                *                                   A2E01343
         DC    C'1621P'             1621 PAPER TAPE PUNCH               A2E01344
         DC    X'32'                *                                   A2E01345
         DC    C'13110'             1311 DISK                           A2E01346
         DC    X'C6'                *                                   A2E01347
```

```
         DC    C'13111'              1311 DISK 2ND                    A2E01348
         DC    X'03'                 *                                A2E01349
         DC    C'13112'              1311 DISK THIRD                  A2E01350
         DC    X'03'                 *                                A2E01351
         DC    C'13113'              1311 DISK 4TH                    A2E01352
         DC    X'03'                 *                                A2E01353
         DC    C'1620C'                                               A2E01354
         DC    X'00'                                                  A2E01355
         DC    C'144LN'                                               A2E01356
         DC    X'00'                                                  A2E01357
         DC    X'FF'                 END OF TABLE                     A2E01358
SEARCH   BAL   15,READSM             NO,READ DATA                     A2E01359
         BAL   15,COMPSM                                              A2E01360
         DC    YL2(CPUTAB)                                            A2E01361
         DC    YL2(IOBUFF)                                            A2E01362
         DC    X'0006'                                                A2E01363
         MVI   CPUTAB+5,C'*'                                          A2E01364
         MVI   CP1A+5,C'*'           IGNORE MODEL                     A2E01365
         BAL   15,COMPSM                                              A2E01366
         DC    YL2(FEATAB)                                            A2E01367
         DC    YL2(IOBUFF)                                            A2E01368
         DC    X'0005'                                                A2E01369
         MVI   CPUTAB+5,C'0'                                          A2E01370
         MVC   CP1A+5(1),CBIT        RESET MODEL                      A2E01371
         CLI   IOBUFF+6,X'40'                                         A2E01372
         BC    BE,SEAR2                                               A2E01373
         MVI   COMPS1+1,X'04'        SET COMPARE = 4                  A2E01374
         BAL   15,COMPSA             BRANCH TO COMPARE                A2E01375
         DC    YL2(DEVTAB)                                            A2E01376
         DC    YL2(IOBUFF+6)                                          A2E01377
         DC    X'000B'                                                A2E01380
         BC    15,SEAR1                                               A2E01385
SEAR2    MVI   CPUTAB+5,C'*'                                          A2E01390
         BAL   15,COMPSM                                              A2E01395
         DC    YL2(DEVSHT)                                            A2E01400
         DC    YL2(IOBUFF)                                            A2E01405
         DC    X'0005'                                                A2E01410
         MVI   CPUTAB+5,C'0'                                          A2E01415
SEAR1    CLC   IOBUFF(6),CTLPRG      COMPARE IF UCB                   A2E01420
         BC    BE,CTP1U              YES,BRANCH                       A2E01425
SEAR4    BAL   15,READSM             UNKNOWN IGNORE                   A2E01430
         CLI   IOBUFF,C'*'           IS A TEXT CARD                   A2E01435
         BC    BE,SEAR4              YES,IGNORE                       A2E01440
         CLC   IOBUFF+9(3),END       IS END                           A2E01445
         BC    BE,SEARCH             YES,BRANCH                       A2E01450
         BC    15,SEAR4              BRANCH                           A2E01455
         EJECT                                                        A2E01460
*                                                                     A2E01465
*        SUBROUTINE 'COMPSM'                                          A2E01470
*        WHICH COMPARES IF ARGUMENT IS PRESENT ON INPUT TAPE          A2E01475
*                                    IF ARGUMENT IS NOT PRESENT       A2E01480
*                                    THE SECTION ON INPUT TAPE IS     A2E01485
*                                    IGNORED                          A2E01490
*                                                                     A2E01495
*                                                                     A2E01500
```

```
*                                       IF ARGUMENT IS PRESENT ON INPUT  A2E01505
*                                       TAPE RECORDS ARE WRITEN          A2E01510
*                                       ON OUTPUT TAPE                   A2E01515
*                                                                        A2E01520
        SPACE 4                                                          A2E01525
*                                                                        A2E01530
*                                                                        A2E01535
COMPSM  LH    WR2,4(15)                                                  A2E01540
        BCTR  WR2,0                     *                                A2E01545
        STC   WR2,COMPS1+1              *                                A2E01550
COMPSA  LH    WR2,4(15)                 STORE LENGTH                     A2E01555
        LH    WR1,0(15)                 STORE ADDRESS                    A2E01560
        LH    WR3,2(15)                 *                                A2E01565
COMPS2  CLI   0(WR1),X'FF'                                               A2E01570
        BC    BE,6(15)                                                   A2E01575
COMPS1  CLC   0(0,WR3),0(WR1)                                            A2E01580
        BC    BE,COMPS3                                                  A2E01585
        LA    WR1,0(WR1,WR2)                                             A2E01590
        BC    15,COMPS2                                                  A2E01595
COMPS3  BAL   15,READSM                                                  A2E01600
        CLI   IOBUFF,C'*'               IS A TEXT CARD                   A2E01605
        BC    BE,SUITA                  YES,IGNORE                       A2E01610
        CLC   IOBUFF+9(3),END                                            A2E01615
        BC    BE,COMPS4                                                  A2E01620
SUITA   BAL   15,PUNCH                                                   A2E01625
        BC    15,COMPS3                                                  A2E01630
COMPS4  CLC   IOBUFF+15(6),F40           COMPARE WITH BLANKS              A2E01635
        BC    BNE,SUIT1                 NOT EQUAL BRANCH                 A2E01640
        BC    15,SEARCH                 EQUAL BRANCH                     A2E01645
SUIT1   BAL   15,PUNCH                  GO TO OUTPUT RECORD              A2E01650
        BAL   15,MESSAG                 SEND MESSAGE END OF SIMULATOR    A2E01655
        DC    YL2(ENDSM1)               *                                A2E01660
        BAL   15,MVT4                   GO TO WRITE A TM                 A2E01665
        BAL   15,MESSAG                 SEND MESSAGE END OF EDITING      A2E01670
        DC    YL2(END16)                *                                A2E01675
        BAL   R1,READTY                                                  A2E01680
        CLC   IOBUFF(3),YES                                              A2E01685
        BC    BE,EDAC1                  GO TO RETRY                      A2E01690
        BAL   15,MESSAG                 SEND MESSAGE ' END OF EDITING '  A2E01695
        DC    YL2(ENBEG)                *                                A2E01700
        NI    SWTC+1,X'0F'              SET RETRY SWITCH OFF             A2E01705
        BC    15,RW1A                   GO TO REWIND                     A2E01710
EDAC1   OI    SWTC+1,X'F0'              SET SWITCH FOR RETRY ON          A2E01715
        BC    15,EDAC                   BRANCH TO REWIND INPUT TAPE      A2E01720
        EJECT                                                            A2E01725
*                                                                        A2E01730
*                                                                        A2E01735
*             CREATION OF UCB AND CCB FOR CONTROL PROGRAM                A2E01740
*                                                                        A2E01745
*                                                                        A2E01750
        SPACE 2                                                          A2E01755
*       CREATION OF CCB                                                  A2E01760
*                                                                        A2E01765
CTP1U   MVC   CONTRL(8),IOBUFF+72       MOVE IDENTIFICATION              A2E01770
        MVI   F0,X'F0'                  SET CHANNEL NUMBER = 0           A2E01775
```

```
         OI    CTABE+1,X'F0'          SET SWITCH ON                          A2E01780
CTABA    LA    WR1,DEVTAB             WR1= A(DEVICE TABLE)                   A2E01785
         CLI   F0,X'F7'               IS CHANNEL NUMBER=7          V1L2      A2E01790
*        CLI   F0,X'F3'               IS CHAN NUMBER=3  (DELETED) V1L2       A2E01795
         BC    BE,CTABC               YES,BRANCH                             A2E01796
CTABU    CLI   0(WR1),X'FF'           IS TABLE EXHAUSTED                     A2E01800
         BC    BE,CTABD               YES,BRANCH                             A2E01805
         CLC   6(1,WR1),F0            IS CHANNEL NUMBER EQUAL TO TABLE       A2E01810
         BC    BE,CTPABJ              YES,BRANCH                             A2E01815
CTABF    LA    WR1,11(WR1)            WR1 = WR1+11                           A2E01820
         BC    15,CTABU               BRANCH                                 A2E01825
CTPABJ   MVC   IOBUFF(80),CTAB+80     MOVE RECORD                            A2E01830
         MVC   IOBUFF+17(6),CTAB      MOVE ADDRESS                           A2E01835
         MVC   IOBUFF+19(1),F0        MOVE CHANNEL NUMBER                    A2E01840
         MVI   IOBUFF+39,X'40'        MOVE BLANKS                            A2E01845
         MVC   IOBUFF+40(39),IOBUFF+39 *                                     A2E01850
CTABE    BC    15,CTABK               SWITCH                                 A2E01855
         BAL   15,PUNCH1              GO TO WRITE                            A2E01860
         SR    WR5,WR5                WR5=0                                  A2E01865
         IC    WR5,F0                 INSERT CHANNEL NUMBER                  A2E01870
         LA    WR5,1(WR5)             INCREMENT CHANNEL NUMBER BY ONE        A2E01875
         STC   WR5,F0                 STORE CHANNEL NUMBER                   A2E01880
         BC    15,CTABA               LOOP                                   A2E01885
*                                                                            A2E01890
*                                                                            A2E01895
CTABK    NI    CTABE+1,X'0F'          SSET SWTCH OFF                         A2E01900
         MVC   IOBUFF(5),CHTAB        MOVE LABEL                             A2E01905
         BC    15,CTABE               BRANCH                                 A2E01910
*                                                                            A2E01915
CTABD    MVC   IOBUFF(80),FULL        MOVE FULL WORD                         A2E01920
         BC    15,CTABE               BRANCH                                 A2E01925
*                                                                            A2E01930
CTABC    MVC   IOBUFF(80),FULL        MOVE FULL WORD                         A2E01935
         BAL   15,PUNCH1              GO TO WRITE                            A2E01940
*        BAL   15,PUNCH1              GO TO WRITE       (DELETED) V1L2       A2E01945
*        BAL   15,PUNCH1              GO TO WRITE       (DELETED) V1L2       A2E01950
*        BAL   15,PUNCH1              GO TO WRITE       (DELETED) V1L2       A2E01955
*        BAL   15,PUNCH1              GO TO WRITE       (DELETED) V1L2       A2E01960
         MVI   F0,X'F0'                                                      A2E01965
CTP1     OI    CTP0A+1,X'F0'          SET SWITCH FOR CHAN LIST ON            A2E01970
         OI    CTP1A+1,X'F0'          SET SWITCH FOR CHANNEL OFF             A2E01975
         LA    WR1,DEVTAB             WR1 = A(DEVICE TABLE )                 A2E01980
CTP0B    CLI   F0,X'F7'               COMPARE CHANNEL TO 7         V1L2      A2E01985
*CTP0B   CLI   F0,X'F3'               COMPARE CHAN TO 3 (DELETED) V1L2       A2E01986
         BC    BE,CTPC                EQUAL BRANCH TO CREATE UCB LIST        A2E01990
         CLI   0(WR1),X'FF'           IS DEVICE LIST FINISHED                A2E01995
         BC    BE,CTP1A               YES,BRANCH                             A2E02000
         CLC   6(1,WR1),F0            COMPARE WITH CHANNEL                   A2E02005
         BC    BE,CTP0A               EQUAL BRANCH                           A2E02010
CTP0D    LA    WR1,11(WR1)            NOT EQUAL INCREMENT WR1 BY 11          A2E02015
         BC    15,CTP0B               LOOP                                   A2E02020
CTP0A    BC    0,CTP0C                SWITCH FOR CCB                         A2E02025
         NI    CTP1A+1,X'0F'          SET SWITCH FOR CHANNEL ON              A2E02030
         MVC   CTAB1+17(2),9(WR1)     MOVE DEVICE ADDRESS                    A2E02035
         MVC   IOBUFF(80),CTAB1       *                                      A2E02040
```

```
         BAL   15,PUNCH1              GO TO OUTPUT RECORD                A2E02045
         MVC   CTAB2+20(4),0(WR1)     *                                  A2E02050
         MVC   CTAB2+19(1),4(WR1)     *                                  A2E02055
         MVC   IOBUFF(80),CTAB2       MOVE RECORD TO OUTPUT BUFFER       A2E02060
         BAL   15,PUNCH1              *                                  A2E02065
         BC    15,CTP0D               LOOP                               A2E02070
CTP1A    BC    0,CTP2A                SWITCH FOR CHANNEL                 A2E02075
         MVC   IOBUFF(80),FULL        MOVE LAST FULLWORD                 A2E02080
         BAL   15,PUNCH1              MOVE OUTPUT RECORD                 A2E02085
CTP2A    SR    WR1,WR1                WR1=0                              A2E02090
         IC    WR1,F0                 INSERT CHANNEL NUMBER              A2E02095
         LA    WR1,1(WR1)             INCREMENT IT BY ONE                A2E02100
         STC   WR1,F0                 STORE IT                           A2E02105
         BC    15,CTP1                LOOP                               A2E02110
*                                                                         A2E02115
*             TREATMENT OF FIRST CCB LIST                                 A2E02120
*                                                                         A2E02125
CTP0C    MVI   IOBUFF,X'40'           MOVE BLANKS IN OUTPUT BUFFER       A2E02130
         MVC   IOBUFF+1(79),IOBUFF    *                                  A2E02135
         MVI   IOBUFF,C'*'            *                                  A2E02140
         BAL   15,PUNCH1              GO TO WRITE                        A2E02145
         MVC   IOBUFF+15(21),IDENT    *                                  A2E02150
         MVC   IOBUFF+37(1),F0        MOVE CHANNEL NUMBER                A2E02155
         BAL   15,PUNCH1              GO TO WRITE                        A2E02160
         MVI   IOBUFF+1,X'40'         MOVE BLANKS TO BUFFER              A2E02165
         MVC   IOBUFF+2(75),IOBUFF+1  *                                  A2E02170
         BAL   15,PUNCH1              GO TO WRITE                        A2E02175
         MVC   CTAB+2(1),F0           MOVE CHANNEL NUMBER                A2E02180
         MVC   IOBUFF(80),CTAB        MOVE RECORD TO OUTPUT BUFFER       A2E02185
         BAL   15,PUNCH1              GO TO WRITE                        A2E02190
         MVC   CTAB+97(6),CTAB        MOVE ADDRESS                       A2E02195
         MVC   IOBUFF(80),CTAB+80     MOVE RECORD TO OUTPUT BUFFER       A2E02200
         BAL   15,PUNCH1              GO TO WRITE                        A2E02205
         NI    CTP0A+1,X'0F'          SET SWITCH OFF                     A2E02210
         BC    15,CTP0A               BRANCH                             A2E02215
*                                                                         A2E02220
*                                                                         A2E02225
*             CREATION OF UCB BLOCKS                                      A2E02230
*                                                                         A2E02235
         SPACE 2                                                         A2E02240
CTPC     MVI   IOBUFF,X'40'           SET OUTPUT BUFFER TO BLANKS        A2E02245
         MVC   IOBUFF+1(79),IOBUFF    *                                  A2E02250
         MVC   IOBUFF+7(5),EJECT      MOVE CONTROL RECORD                A2E02255
*                                     FOR ASSEMBLY                       A2E02260
         BAL   15,PUNCH1              GO TO WRITE                        A2E02265
         MVI   IOBUFF,C'*'            MOVE A NEW RECORD FOR ASSEMBLY     A2E02270
         MVI   IOBUFF+1,X'40'         MOVE BLANKS TO OUTPUT BUFFER       A2E02275
         MVC   IOBUFF+2(78),IOBUFF+1  *                                  A2E02280
         BAL   15,PUNCH1              GO TO WRITE                        A2E02285
         MVC   IOBUFF+15(19),LISCCB   MOVE IDENTIFICATION                A2E02290
         BAL   15,PUNCH1              GO TO WRITE                        A2E02295
         OI    CTPC0+1,X'F0'          SET SWITCH ON                      A2E02300
         LA    WR1,DEVTAB             WR1 = ADDRESS OF DEVICE TABLE      A2E02305
CTPCA    CLI   0(WR1),X'FF'           IS TABLE FINISHED                  A2E02310
         BC    BE,SEAR4               YES,CONTINUE TREATMENT             A2E02315
```

```
         LA    WR2,DEVTAB                                          V1L2 A2E02316
CTPC1    CLR   WR1,WR2                                             V1L2 A2E02317
         BC    BE,CTPC2                                            V1L2 A2E02318
         CLC   0(11,WR1),0(WR2)       IS DEV ALREADY DEFINED      V1L2 A2E02319
         BC    8,CTPCD                YES-BYPASS                  V1L2 A2E02320
         LA    WR2,11(WR2)            NO                          V1L2 A2E02321
         BC    15,CTPC1               LOOP                        V1L2 A2E02322
CTPC2    MVC   CCBTAB+1(4),0(WR1)     MOVE DEVICE ADDRESS         V1L2 A2E02323
*        CLI   0(WR1),X'40'           IS A BLANK IN TABL(DELETED) V1L2 A2E02324
*        BC    BE,CTPCD               YES,IGNORE        (DELETED) V1L2 A2E02325
*        MVC   CCBTAB+1(4),0(WR1)     NOT YET MOVE ADDR (DELETED) V1L2 A2E02326
         MVC   CCBTAB(1),4(WR1)       *                           A2E02327
         MVC   CCBTAB+17(4),0(WR1)    *                           A2E02330
         MVC   IOBUFF(80),CCBTAB      MOVE RECORD TO OUTPUT BUFFER A2E02335
         MVC   IOBUFF+39(11),DEVTYP   MOVE DEVICE TYPE            A2E02340
         MVC   IOBUFF+46(4),0(WR1)    *                           A2E02345
         BAL   15,PUNCH1              GO TO WRITE                 A2E02350
         MVI   CCBTAB+97,X'F0'        MOVE A 'F0' TO DEVICE ADDRESS A2E02355
         MVC   CCBTAB+98(1),6(WR1)    MOVE DEVICE ADDRESS         A2E02360
         MVC   CCBTAB+99(2),9(WR1)    *                           A2E02365
         MVC   IOBUFF(80),CCBTAB+80   MOVE OUTPUT RECORD TO BUFFER A2E02370
         MVC   IOBUFF+39(6),DEV360    MOVE DEVICE                 A2E02375
         BAL   15,PUNCH1              GO TO WRITE                 A2E02380
         MVC   IOBUFF(80),CCBX00      *                           A2E02385
         MVC   IOBUFF+39(6),DEVSPF    MOVE DEVSPF                 A2E02390
         BAL   15,PUNCH1              GO TO WRITE                 A2E02395
         MVC   IOBUFF+39(6),BORCH     MOVE BORCH                  A2E02400
         BAL   15,PUNCH1              GO TO WRITE                 A2E02405
         MVC   IOBUFF(80),CCBA00      MOVE RECORD TO OUTPUT BUFFER A2E02410
         MVC   IOBUFF+39(6),DEVSV     MOVE DEVSVC                 A2E02415
         BAL   15,PUNCH1              GO TO WRITE                 A2E02420
         MVC   IOBUFF+39(6),DEVCH     MOVE DEVCHN                 A2E02425
         BAL   15,PUNCH1              GO TO WRITE                 A2E02430
         MVC   IOBUFF+39(6),DEVIN     MOVE DEVINT                 A2E02435
         BAL   15,PUNCH1              GO TO WRITE                 A2E02440
         MVC   IOBUFF(80),CCBXL3      MOVE RECORD TO OUTPUT BUFFER A2E02445
         BAL   15,PUNCH1              GO TO WRITE                 A2E02450
CTPC0    BC    15,CONSLE              SWITCH FOR CONSOLE          A2E02455
         BC    15,DISA                BRANCH FOR DISK             A2E02460
CTPCB    MVC   IOBUFF(80),CCBSPF      MOVE RECORD TO OUTPUT BUFFER A2E02465
         BAL   15,PUNCH1              GO TO WRITE                 A2E02470
CTPCD    LA    WR1,11(WR1)            INCREMENT DEVICE TABLE ADDR BY A2E02475
*                                     ELEVEN                      A2E02480
         BC    15,CTPCA               LOOP                        A2E02485
CONSLE   CLC   0(4,WR1),F1052         COMPARE IF 1052 PRESENT     A2E02490
*        BC    7,CTPCB                NO BRANCH         (DELETED) V1L2 A2E02495
         BC    7,DISA                 NO BRANCH                   V1L2 A2E02496
         MVC   IOBUFF(80),CCBSPF      MOVE OUTPUT RECORD          A2E02500
         MVC   IOBUFF+17(2),E60       *                           A2E02505
         BAL   15,PUNCH1              GO TO WRITE                 A2E02510
         MVC   IOBUFF(80),CCBA00      MOVE RECORD                 A2E02515
         MVC   IOBUFF(6),REQUST                                   A2E02520
         BAL   15,PUNCH1              GO TO WRITE                 A2E02525
         NI    CTPC0+1,X'0F'          SET SWITCH OFF              A2E02530
         BC    15,CTPCD               LOOP                        A2E02535
```

```
DISA     CLC   0(4,WR1),F2311          COMPARE IF DISK                  A2E02540
         BC    7,CTPCB                 NO,BRANCH                        A2E02545
         MVC   IOBUFF(80),CCBSPF       MOVE RECORD                      A2E02550
         MVC   IOBUFF+17(2),E80        *                                A2E02555
         BAL   15,PUNCH1               GO TO WRITE                      A2E02560
         BC    15,CTPCD                BRANCH                           A2E02565
F2311    DC    C'2311'                                                  A2E02570
E80      DC    C'80'                                                    A2E02575
CBIT     DC    C' '                                                     A2E02580
NODSK    DC    C'NODSK'                BUFFER                           A2E02585
CHTAB    DC    C'CHTAB'                LABEL FOR CHANNEL                A2E02590
REQUST   DC    C'REQUST'                                                A2E02595
F1052    DC    C'1052'                 CONSTANCE FOR 1052 TYPE WRITER   A2E02600
         EJECT                                                          A2E02605
*                                                                       A2E02610
*             RECORDS FOR CONTROL PROGRAM                               A2E02615
*                                                                       A2E02620
DEVTYP   DC    C'DEVTYP-     '         DEVICE TYPE                      A2E02625
NISKV    DC    C'NISKV'                NO 'DISKV' VERSION               A2E02630
DISKV    DC    C'DISKV'                DISKV VERSION                    A2E02635
A13110   DC    C'13110'                DISKS                            A2E02640
DEV360   DC    C'DEV360'               DEVICE 360                       A2E02645
DEVSPF   DC    C'DEVSPF'               *                                A2E02650
BORCH    DC    C'BORCH '               *                                A2E02655
LISCCB   DC    C'UNIT CONTROL '                                         A2E02660
         DC    C'BLOCKS'                                                A2E02665
EJECT    DC    C'EJECT'                CONTROL CARD FOR ASSEMBLY        A2E02670
DEVSV    DC    C'DEVSVC'               *                                A2E02675
DEVCH    DC    C'DEVCHN'               *                                A2E02680
DEVIN    DC    C'DEVINT'               *                                A2E02685
CTAB     DC    C'CH0LST DC     A(0)'                                    A2E02690
         DC    C'          '                                            A2E02695
         DC    C'          '                                            A2E02700
         DC    C'IOQBGG-ADD.OF 1ST '                                    A2E02705
         DC    C'UCB ON CHAIN'                                          A2E02710
         DC    C'          '                                            A2E02715
         DC    C' '                                                     A2E02720
         DC    C'       DC'                                             A2E02725
         DC    C'      A(      )'                                       A2E02730
         DC    C'          '                                            A2E02735
         DC    C'     '                                                 A2E02740
         DC    C'ADDR OF POINTER'                                       A2E02745
         DC    C' TO NEXT UCB'                                          A2E02750
         DC    C'          '                                            A2E02755
         DC    C'    '                                                  A2E02760
CTAB1    DC    C'       DC'                                             A2E02765
         DC    C'      X''00'''                                         A2E02770
         DC    C'          '                                            A2E02775
         DC    C'          '                                            A2E02780
         DC    C'          '                                            A2E02785
         DC    C'          '                                            A2E02790
         DC    C'          '                                            A2E02795
         DC    C'          '                                            A2E02800
CTAB2    DC    C'       DC'                                             A2E02805
         DC    C'      AL3(     )'                                      A2E02810
```

```
         DC    C'          '                                    A2E02815
         DC    C'          '                                    A2E02820
         DC    C'          '                                    A2E02825
         DC    C'          '                                    A2E02830
         DC    C'          '                                    A2E02835
         DC    C'     '                                         A2E02840
CCBTAB   DC    C'       DC'                                     A2E02845
         DC    C'      C''    '''                               A2E02850
         DC    C'          '                                    A2E02855
         DC    C'          '                                    A2E02860
         DC    C'          '                                    A2E02865
         DC    C'          '                                    A2E02870
         DC    C'          '                                    A2E02875
         DC    C'        '                                      A2E02880
         DC    C'       DC'                                     A2E02885
         DC    C'      X''    '''                               A2E02890
         DC    C'          '                                    A2E02895
         DC    C'        '                                      A2E02900
         DC    C'DEV360'                                       A2E02905
         DC    C'          '                                    A2E02910
         DC    C'          '                                    A2E02915
         DC    C'          '                                    A2E02920
         DC    C'     '                                         A2E02925
CCBX00   DC    C'       DC'                                     A2E02930
         DC    C'      X''00'''                                 A2E02935
         DC    C'          '                                    A2E02940
         DC    C'          '                                    A2E02945
         DC    C'          '                                    A2E02950
         DC    C'          '                                    A2E02955
         DC    C'          '                                    A2E02960
         DC    C'          '                                    A2E02965
CCBA00   DC    C'       DC'                                     A2E02970
         DC    C'      A(0)'                                    A2E02975
         DC    C'          '                                    A2E02980
         DC    C'          '                                    A2E02985
         DC    C'          '                                    A2E02990
         DC    C'          '                                    A2E02995
         DC    C'          '                                    A2E03000
         DC    C'          '                                    A2E03005
         DC    C' '                                            A2E03010
CCBXL3   DC    C'       DC'                                     A2E03015
         DC    C'      XL3''0'''                                A2E03020
         DC    C'          '                                    A2E03025
         DC    C'        '                                      A2E03030
         DC    C'LAST THREE BYTES '                            A2E03035
         DC    C'OF SENSE'                                     A2E03040
         DC    C'          '                                    A2E03045
         DC    C'     '                                         A2E03050
CCBSPF   DC    C'       DC'                                     A2E03055
         DC    C'      X''E0'''                                 A2E03060
         DC    C'          '                                    A2E03065
         DC    C'         '                                     A2E03070
         DC    C'INVST-INVALID DEVICE'                         A2E03075
         DC    C' STATUS BITS'                                 A2E03080
         DC    C'          '                                    A2E03085
```

```
FULL     DC    C'       DC'                                          A2E03090
         DC    C'      F''0'''                                       A2E03095
         DC    C'           '                                        A2E03100
         DC    C'           '                                        A2E03105
         DC    C'           '                                        A2E03110
         DC    C'           '                                        A2E03115
         DC    C'           '                                        A2E03120
         DC    C'           '                                        A2E03125
         DC    C' '                                                  A2E03130
E60      DC    C'60'                                                 A2E03135
CTLPRG   DC    C'CTLPRG'              CONSTANTE FOR CONTROL PROGRAM  A2E03140
CONTRL   DS    8C                     CONSTANTE FOR IDENTIFICATION   A2E03145
IDENT    DC    C'CHANNEL CONTROL '                                   A2E03150
         DC    C'BLOCK'                                              A2E03155
F0       DC    C'0'                   F0 = 0                         A2E03160
         EJECT                                                       A2E03165
*                                                                    A2E03170
*                                                                    A2E03175
*             SUBROUTINE TO REWIND TAPES                             A2E03180
*                                                                    A2E03185
RW1A     SR    WR1,WR1                                               A2E03190
         STH   WR1,DEVA1              STORE DEVICE ADDRESS = 0       A2E03195
         MVI   DEV24,X'40'            SET DEVICE TYPE  = BLANKS      A2E03200
         MVC   DEV24+1(4),DEV24       *                              A2E03205
         CNOP  0,4                                                   A2E03210
         SVC   17                     SEARCH FOR DEVICE ADDRESS      A2E03215
         DC    C'SIM2OUT '                                           A2E03220
DEVA1    DC    X'0000'                *                              A2E03225
DEV24    DC    C'    '                *                              A2E03230
         DC    C' '                                                  A2E03235
         DC    AL3(MVT3)              GO TO ERROR                    A2E03240
         CLC   DEV24(4),DEV24A        COMPARE IF TAPE                A2E03245
         BC    7,EDAC                 NO BRANCH                      A2E03250
         MVC   DEVRW(2),DEVA1         SET DEVICE ADDR IN REWIND      A2E03255
*                                     SUBROUTINE                     A2E03260
         BAL   15,FIRSRW              GO TO REWIND                   A2E03265
EDAC     MVI   EDI2,X'00'             SET DEVICE ADDRESS TO ZEROS    A2E03270
         MVI   EDI2+1,X'00'           *                              A2E03275
         MVI   EDI1,C' '              SET TYPE OF UNIT TO BLANKS     A2E03280
         MVC   EDI1+1(4),EDI1         *                              A2E03285
         CNOP  0,4                    SEARCH FOR DEVICE ADDRESS      A2E03290
         SVC   17                     FOR INPUT DEVICE               A2E03295
         DC    C'SIM2IN  '                                           A2E03300
EDI2     DC    X'0000'                *                              A2E03305
EDI1     DC    C'    '                                               A2E03310
         DC    C' '                   *                              A2E03315
         DC    AL3(MVT1)              *                              A2E03320
         CLC   EDI1(4),DEV24A         COMPARE IF DEVICE IS A TAPE    A2E03325
         BC    7,SWTC                 NO,BRANCH                      A2E03330
         MVC   DEVRW(2),EDI2          SET DEVICE ADDRESS FOR REWIND  A2E03335
         BAL   15,FIRSRW              GO TO SEND A REWIND COMMAND    A2E03340
SWTC     BC    0,RETRY                RETRY SWITCH                   A2E03345
         BAL   15,MESSAG              SEND A MESSAGE                 A2E03350
         DC    YL2(WAI)               *                              A2E03355
         SVC   9                      ENABLE INTERRUPTS              A2E03360
```

```
         SVC   19                      WAIT STATE                        A2E03365
         BC    15,*-2                  LOOP                              A2E03370
         EJECT                                                           A2E03375
*                                                                        A2E03380
*                                                                        A2E03385
*              SUBROUTINE TO SEND A REWIND COMMAND                       A2E03390
*                                                                        A2E03395
*                                                                        A2E03400
         SPACE 4                                                        A2E03405
FIRSRW   SVC   9                       ENABLE INTERRUPTS                 A2E03410
         CNOP  4,8                                                      A2E03415
         SVC   13                      I/O REQUEST FOR REWIND COMMAND    A2E03420
DEVRW    DC    X'0000'                 DEVICE ADDRESS                    A2E03425
         DC    A(CCWREW)               ADDR OF CCW FOR REWIND COMMAND    A2E03430
         DS    4C                      *                                 A2E03435
         DS    D                       *                                 A2E03440
SVCPSW   DS    D                       PSW OF INTERRUPTION               A2E03445
         DC    A(NRMRET)               NORMAL RETURN                     A2E03450
         DC    A(NRMRET)               *                                 A2E03455
         BCR   15,15                   RETURN TO CALLER                  A2E03460
         CNOP  2,4                                                       A2E03465
NRMRET   SVC   3                       RETURN TO POINT OF INTERRUPTION   A2E03470
         DC    A(SVCPSW)               *                                 A2E03475
*                                                                        A2E03480
*              CCW FOR REWIND                                            A2E03485
*                                                                        A2E03490
CCWREW   CCW   X'07',*,X'00',1                                           A2E03495
DEV24A   DC    C'2400'                 TYPE OF UNIT FOR TAPES            A2E03500
END      DC    C'END'                                                    A2E03505
F40      DC    C'      '                                                 A2E03510
YES      DC    C'YES'                                                    A2E03515
         EJECT                                                           A2E03520
*                                                                        A2E03525
*              SUBROUTINE TO READ CONTROL CARDS                          A2E03530
*              USING I/O PACKAGE                                         A2E03535
*                                                                        A2E03540
         CNOP  0,4                                                      A2E03545
READ     SVC   18                                                       A2E03550
         DC    C'SIM2INF '                                              A2E03555
         DC    FL2'80'                                                  A2E03560
         DC    A(IOBUFF-3)                                              A2E03565
         CLI   IOBUFF-3,X'07'                                           A2E03570
         BC    8,READPR                                                 A2E03575
         BAL   15,MESVC4                                                A2E03580
         DC    YL2(REA2)                                                A2E03585
WAI12    BAL   15,MESVC4                                                A2E03590
         DC    YL2(WAI)                                                 A2E03595
WAI1     SVC   9                       ENABLE INTERRUPTS                 A2E03600
         SVC   19                      WAIT STATE                        A2E03605
         BC    15,*-2                  *                                 A2E03610
READPR   ST    15,REG15                                                 A2E03615
         MVI   IOBUFF+27,X'15'                                          A2E03620
         MVI   IOBUFF-3,X'40'          *                                 A2E03625
         MVI   IOBUFF-4,X'1F'          SET LENGTH OF MESSAGE             A2E03630
         BAL   15,MESSAG                                                A2E03635
```

```
         DC    YL2(IOBUFF-4)                                          A2E03640
         MVI   IOBUFF+27,X'40'       RESET BLANK IN INPUT BUFFER      A2E03645
         MVI   IOBUFF-4,X'00'        *                                A2E03650
         L     15,REG15              RESTORE R15 REGISTER             A2E03655
         BCR   15,15                 RETURN TO CALLER                 A2E03660
         EJECT                                                        A2E03665
*                                                                     A2E03670
*              SEND MESSAGE IF CARD 'EDITCTL' NOT DEFINED             A2E03675
*                                                                     A2E03680
MESVC4   LH    WR1,0(15)             LOAD ADDRESS OF MESSAGE IN WR1   A2E03685
         LA    R2,0                  R2 = 0                           A2E03690
         IC    R2,0(WR1)             INSERT IN R2 LENGTH OF MESSAGE   A2E03695
         BCTR  R2,0                  DECREMENT R2 BY ONE              A2E03700
         STC   R2,DL                 STORE R2 IN DATA LENGTH          A2E03705
         LA    WR1,1(WR1)            INCREMENT ADDRESS BY ONE         A2E03710
         STH   WR1,DL+2              STORE ADDRESS IN SVC 18          A2E03715
         SVC   9                     ENABLE INTERRUPTS                A2E03720
         CNOP  2,4                                                    A2E03725
         SVC   4                     SEND MESSAGE                     A2E03730
DL       DC    FL1'0'                LENGTH                           A2E03735
         DC    AL3(0)                ADDRESS                          A2E03740
         TM    0(WR1),X'07'          IS MESSAGE WRITEN CORRECTLY      A2E03745
         BC    8,*-4                 NOT YET LOOP                     A2E03750
         BC    1,2(15)               YES,RETURN TO CALLER             A2E03755
         TM    0(WR1),X'03'          TEST IF ANY ERROR                A2E03760
         BC    1,WAI1                YES,GO TO WAIT STATE             A2E03765
         BC    15,2(15)              NO,RETURN TO CALLER              A2E03770
         EJECT                                                        A2E03775
*                                                                     A2E03780
*                                                                     A2E03785
*              SUBROUTINE OUTPUT OF SIM-20                            A2E03790
*              USING I/O PACKAGE                                      A2E03795
*                                                                     A2E03800
*                                                                     A2E03805
PUNCH1   MVC   IOBUFF+72(3),CONTRL   MOVE IDENTIFICATION              A2E03810
         PACK  CONT(8),CONTRL+3(5)   *                                A2E03815
         CVB   WR5,CONT              *                                A2E03820
         LA    WR5,1(WR5)            *                                A2E03825
         CVD   WR5,CONT              *                                A2E03830
         NI    CONT+7,X'F0'          *                                A2E03835
         UNPK  IOBUFF+75(7),CONT+5(4) *                               A2E03840
         MVC   CONTRL+3(5),IOBUFF+75 SAVE IDENTIFICATION              A2E03845
         BC    15,PUNCH              GO TO WRITE                      A2E03850
CONT     DS    1D                    COUNTER FOR IDENTIFICATION       A2E03855
         DC    X'0F'                 *                                A2E03860
         CNOP  0,4                                                    A2E03865
PUNCH    SVC   18                    REQUEST ON EDO1FILE DEVICE       A2E03870
         DC    C'SIM2OUT '           SYMBOL                           A2E03875
         DC    FL2'80'               LENGTH                           A2E03880
         DC    A(IOBUFF-1)           ADDRESS                          A2E03885
         CLI   IOBUFF-1,X'07'        IS WRITE CORRECT                 A2E03890
         BC    8,PRINT               YES,GO TO PRINT                  A2E03895
         CLI   IOBUFF-1,X'01'        NO,TEST ERROR                    A2E03900
         BC    8,MVT3                BRANCH TO ERROR                  A2E03905
         CLI   IOBUFF-1,X'03'        TEST UE                          A2E03910
```

```
         BC    8,MVT4                  YES,GO TO WRITE A TAPE MARK      A2E03915
         BC    15,WAI12                LOOP                             A2E03920
MVT3     BAL   15,MESVC4               SEND A MESSAGE ERROR             A2E03925
         DC    YL2(MVT31)              *                                A2E03930
         BC    15,WAI12                BRANCH TO WAIT STATE             A2E03935
         EJECT                                                          A2E03940
*                                                                       A2E03945
*                                                                       A2E03950
*             SUBROUTINE TO WRITE A TAPE MARK ON 'EDO1FILE' DEVICE      A2E03955
*                                                                       A2E03960
MVT4     MVI   IOBUFF,X'7F'            MOVE CHARACTER TM IN OUTPUT BUF  A2E03965
         MVC   IOBUFF+1(79),IOBUFF     MOVE 7F IN BUFFER                A2E03970
         ST    15,REG15                SAVE REGISTER R15                A2E03975
         CNOP  0,4                                                      A2E03980
         SVC   18                      REQUEST ON 'EDO1FILE' DEVICE     A2E03985
         DC    C'SIM2OUT '             SYMBOL                           A2E03990
         DC    FL2'1'                  LENGTH                           A2E03995
         DC    A(IOBUFF-1)             ADDRESS                          A2E04000
         BAL   15,MESVC4               SEND A MESSAGE                   A2E04005
         DC    YL2(ENDTA)              *                                A2E04010
         L     15,REG15                RESTORE R15 REGISTER             A2E04015
         BCR   15,15                   GO TO PRINT                      A2E04020
REG15    DS    F                       BUFFER TO SAVE R15 REGISTER      A2E04025
         SPACE 4                                                        A2E04030
*                                                                       A2E04035
*                                                                       A2E04040
*             SUBROUTINE TO PRINT EDIT INFORMATION                      A2E04045
*             IF 'ED2PRINT' IS PRESENT                                  A2E04050
*                                                                       A2E04055
PRINT    MVC   IOBUFF+81(80),IOBUFF                                     A2E04060
         MVI   IOBUFF+80,X'00'                                          A2E04065
         MVI   IOBUFF+79,X'00'                                          A2E04070
         CNOP  0,4                                                      A2E04075
         SVC   18                      REQUEST ON PRINTER               A2E04080
         DC    C'SIM2PRNT'                                              A2E04085
         DC    FL2'81'                 LENGTH                           A2E04090
         DC    A(IOBUFF+79)            ADDRESS                          A2E04095
         BCR   15,15                                            V1L2    A2E04100
*        BCR   8,15                    RETURN TO CALLER  (DELETED) V1L2 A2E04101
         EJECT                                                          A2E04105
*             TREATMENT OF CPU1 CONTROL CARD                            A2E04110
*                                                                       A2E04115
CP1      MVC   CP1A(6),IOBUFF+5        MOVE MODEL OF 1620               A2E04120
         IC    WR3,IOBUFF+12           IN SEARCH TABLE                  A2E04125
         LA    WR3,1(WR3)                                               A2E04130
         STC   WR3,IOBUFF+10                                            A2E04135
         MVC   CAPACT(6),IOBUFF+5      MOVE CAPACITY OF 1620            A2E04140
         MVC   CP2B+1(2),IOBUFF+12     IN SEARCH TABLE                  A2E04145
         CLI   CP1A+5,C'2'                                              A2E04150
         BC    BE,CP11                                                  A2E04155
         BC    15,READ1                GO TO READ                       A2E04160
CP11     OI    SWITC1+1,X'80'                                           A2E04165
         OI    SWITC2+1,X'80'                                           A2E04170
         OI    SWITC3+1,X'F0'                                           A2E04175
         OI    SWITC4+1,X'F0'                                           A2E04180
```

```
         OI    SWITC5+1,X'F0'                                    A2E04185
         MVC   IOBUFF(13),FEAT                                   A2E04190
         BAL   9,FEAT1                                           A2E04195
         MVC   IOBUFF(13),FEAT+13                                A2E04200
         BAL   9,FEAT1                                           A2E04205
CP1ACF   MVC   IOBUFF(13),FEAT+26      MOVE RECORD 'DIVID'       A2E04210
         BAL   9,FEAT1                                           A2E04215
         NI    SWITC1+1,X'0F'                                    A2E04220
         NI    SWITC2+1,X'0F'                                    A2E04225
         NI    SWITC3+1,X'0F'                                    A2E04230
         NI    SWITC4+1,X'0F'                                    A2E04235
         NI    SWITC5+1,X'0F'                                    A2E04240
         BC    15,READ1                                          A2E04245
CP2B     DC    X'F00000'           CAPACITY OF 1620              A2E04250
CP2A     DC    X'000000'            CAPACITY OF 360              A2E04255
CPU1     DC    C'CPU1 1620/1,20K'                                A2E04260
         DC    C'CPU1 1620/1,40K'                                A2E04265
         DC    C'CPU1 1620/1,60K'                                A2E04270
         DC    C'CPU1 1620/2,60K'                                A2E04275
         DC    C'CPU1 1620/2,20K'                                A2E04280
         DC    C'CPU1 1620/2,40K'                                A2E04285
         DC    X'FF'                                             A2E04290
         EJECT                                                   A2E04295
*                                                                A2E04300
*                                                                A2E04305
*             TREATMENT OF CPU2 CONTROL CARD                     A2E04310
*                                                                A2E04315
         SPACE 4                                                 A2E04320
CP2      CLI   IOBUFF+14,C'K'          IS CPU2 HIGHER THAN 64K   A2E04325
         BC    7,CP21                  YES,BRANCH                A2E04330
         MVI   CP2A,X'F0'                                        A2E04335
         MVC   CP2A+1(2),IOBUFF+12                               A2E04340
         BC    15,READ1                GO TO READ                A2E04345
CP21     MVC   CP2A(3),IOBUFF+12                                 A2E04350
         BC    15,READ1                GO TO READ                A2E04355
*                                                                A2E04360
CPU2     DC    C'CPU2 360/30,32K '                               A2E04365
         DC    C'CPU2 360/30,64K '                               A2E04366
         DC    C'CPU2 360/30,128K'                               A2E04367
         DC    C'CPU2 360/40,32K '                               A2E04368
         DC    C'CPU2 360/40,64K '                               A2E04369
         DC    C'CPU2 360/40,128K'                               A2E04370
         DC    C'CPU2 360/40,256K'                               A2E04371
         DC    X'FF'                                             A2E04395
         EJECT                                                   A2E04400
*                                                                A2E04405
*             TREATMENT OF FEATURE CONTROL CARD                  A2E04410
*                                                                A2E04415
FEAT1    LA    R1,FEATAB                                         A2E04420
FEAT3    CLI   0(R1),X'FF'                                       A2E04425
         BC    BE,FEAT2                                          A2E04430
         CLC   0(3,R1),IOBUFF+8        SCAN FEATAB TO BE SURE    A2E04435
         BC    BE,FEAT4                CONTROL CARD IS SINGLE    A2E04440
         LA    R1,5(R1)                                          A2E04445
         BC    15,FEAT3                                          A2E04450
```

```
FEAT4    CLC   3(2,R1),IOBUFF+11                                          A2E04455
SWITC1   BCR   0,9                                                       A2E04460
         BC    BE,READ1                 *                               A2E04465
         CLI   IOBUFF+12,C'D'           IS A CARD INDAD                 A2E04470
SWITC2   BCR   0,9                                                       A2E04475
         BC    BE,READ1                 YES,IGNORE                      A2E04480
         CLI   CP1A+5,X'F2'                                             A2E04485
         BC    BNE,FEAT7                INVALID,SEND A MESSAGE          A2E04490
         MVC   0(5,R1),IOBUFF+8         VALID MOVE IT                   A2E04495
SWITC3   BCR   0,9                                                       A2E04500
         BC    15,READ1                                                 A2E04505
FEAT2    CLI   IOBUFF+12,C'X'                                           A2E04510
         BC    BE,FEAT6                                                 A2E04515
FEAT8    MVC   0(5,R1),IOBUFF+8                                         A2E04520
         MVI   5(R1),X'FF'                                              A2E04525
         CLI   IOBUFF+12,C'T'           COMPARE IF 'FLOAT' IS PRESENT   A2E04530
         BC    BE,CP1ACF                YES,GO TO MOVE 'DIVID'          A2E04535
SWITC4   BCR   0,9                                                       A2E04540
         BC    15,READ1                                                 A2E04545
FEAT6    CLI   CP1A+5,X'F2'                                             A2E04550
         BC    BNE,FEAT7                                                A2E04555
         BC    15,FEAT8                                                 A2E04560
FEAT7    BAL   15,MESSAG                                                A2E04565
         DC    YL2(FEAT9)                                               A2E04570
SWITC5   BCR   0,9                                                       A2E04575
         BC    15,READ1                                                 A2E04580
FEAT     DC    C'FEATURE INDAD'                                         A2E04585
         DC    C'FEATURE TRANS'                                         A2E04590
         DC    C'FEATURE DIVID'                                         A2E04595
         DC    C'FEATURE INDEX'                                         A2E04600
         DC    C'FEATURE FLOAT'                                         A2E04605
         DC    C'FEATURE 144LN'                                         A2E04610
         DC    C'FEATURE DISKV'                                         A2E04615
         DC    X'FF'                                                    A2E04625
         EJECT                                                          A2E04630
*                TREATMENT OF DEVICE CONTROL CARD                        A2E04635
*                                                                       A2E04636
DEV1     CLI   IOBUFF+19,C'0'           TEST CHANNEL NUMBER       V1L2 A2E04637
         BC    BL,ER1A                  BR IF INVALID             V1L2 A2E04638
         CLI   IOBUFF+19,C'6'           TEST CHANNEL NUMBER       V1L2 A2E04639
         BC    BH,ER1A                  BR IF INVALID             V1L2 A2E04640
         CLC   IOBUFF+19(1),IOBUFF+23   TEST CHANNEL COMPATIBILITY V1L2 A2E04641
         BC    BNE,ER1A                 BR IF INCOMPATIBLE        V1L2 A2E04642
*                                                                 V1L2 A2E04643
         LA    R2,DEVTAB                GET SHORT TABLES.         V1L2 A2E04644
         LA    R1,DEVSHT                .ADDRESSES                V1L2 A2E04645
DEV2     CLI   0(R2),X'FF'              TEST END OF TABLE         V1L2 A2E04646
         BC    BE,DEV7                                          V1L2 A2E04647
*                        DEVICE TABLE NOT EMPTY                   V1L2 A2E04648
         CLC   IOBUFF+7(5),0(R1)        TEST DEVICE 1620          V1L2 A2E04649
         BC    BE,DEV6                  BR ON DEVICE ALREADY ASSIGNV1L2 A2E04650
         CLC   IOBUFF+23(3),8(R2)       TEST DEVICE ADDRESS       V1L2 A2E04651
         BC    BE,DEV5                  BR ON SAME S/360 ADDRESS  V1L2 A2E04652
         CLC   IOBUFF+13(5),0(R2)       TEST S/360 DEVICE TYPE    V1L2 A2E04653
         BC    BE,DEV4                  BR ON SAME TYPE           V1L2 A2E04654
```

```
DEV3     LA    R2,11(R2)              GET NEXT ELEMENT.                V1L2 A2E04655
         LA    R1,5(R1)               .IN TABLES                       V1L2 A2E04656
         BC    B,DEV2                 VERIFY NEXT ELEMENT              V1L2 A2E04657
*                                                                       V1L2 A2E04658
DEV4     CLI   4(R2),C'P'             IS TYPE 'PUNCH'                  V1L2 A2E04659
         BC    BE,DEV3                BR IF YES, OK                    V1L2 A2E04660
         BC    B,ER1A                 IF NO, ERROR                     V1L2 A2E04661
*                                                                       V1L2 A2E04662
*                                                                       V1L2 A2E04663
DEV5     CLC   IOBUFF+13(4),0(R2)     TEST DEVICE TYPE                 V1L2 A2E04664
         BC    BE,DEV3                BR ON SAME TYPE                  V1L2 A2E04665
         BC    B,ER1A                 BR ON ERROR                      V1L2 A2E04666
*                                                                       V1L2 A2E04667
DEV6     BAL   15,MESSAG              SEND MESSAGE                     V1L2 A2E04668
         DC    YL2(READ13)            DUPLICATE INFORMATION            V1L2 A2E04669
         BC    B,DEV9                 BR TO ACCEPT NEW INFORMATIO      V1L2 A2E04670
*              TEST 1403/144LN COMPATIBILITY                            V1L2 A2E04671
DEV7     CLC   IOBUFF+13(4),K1403     TEST FOR COMPATIBILITY 1403      V1L2 A2E04672
         BC    7,DEV8                 WITH 144LN                       V1L2 A2E04673
         LA    R3,FEATAB              1403 IS SPCIFIED                 V1L2 A2E04674
DEV71    CLI   0(R3),X'FF'            IS END OF TABLE REACHED          V1L2 A2E04675
         BC    8,DEV8                 YES, 144LN IS NOT PRESENT        V1L2 A2E04676
         CLC   0(5,R3),K144LN                                          V1L2 A2E04677
         BC    8,ER1A                 144LN IS PRESENT-ERROR           V1L2 A2E04678
         LA    R3,5(R3)                                                V1L2 A2E04679
         BC    15,DEV71                                                V1L2 A2E04680
*              SET NEW DEVICE IN TABLES                                 V1L2 A2E04681
DEV8     MVI   5(R1),X'FF'            SET PROTECTIONS IN.              V1L2 A2E04682
         MVI   11(R2),X'FF'           .TABLES                          V1L2 A2E04683
DEV9     CLI   IOBUFF+20,C','         TEST.                            V1L2 A2E04684
         BC    BNE,ER1A               .                                V1L2 A2E04685
         CLI   IOBUFF+21,C'X'         .VALIDITY.                       V1L2 A2E04686
         BC    BNE,ER1A               .                                V1L2 A2E04687
         CLI   IOBUFF+22,C''''        .OF CARD.                        V1L2 A2E04688
         BC    BNE,ER1A               .                                V1L2 A2E04689
         CLI   IOBUFF+26,C''''        .FORMAT                          V1L2 A2E04690
         BC    BNE,ER1A               .                                V1L2 A2E04691
         MVC   0(8,R2),IOBUFF+13      MOVE S/360 DEVICE TYPE           V1L2 A2E04692
         MVC   8(3,R2),IOBUFF+23      MOVE S/360 DEVICE ADDRESS        V1L2 A2E04693
         MVC   0(5,R1),IOBUFF+7       MOVE 1620 DEVICE TYPE            V1L2 A2E04694
         BC    B,READ1                GO TO NEXT CARD                  V1L2 A2E04695
K1403    DC    C'1403'                                                 V1L2 A2E04696
K144LN   DC    C'144LN'                                                V1L2 A2E04697
*DEV1    CLI   IOBUFF+19,X'F0'                          (DELETED) V1L2 A2E04698
*        BC    10,DEV1B                                 (DELETED) V1L2 A2E04699
*        BC    15,DEV1C                                 (DELETED) V1L2 A2E04700
*DEV1B   CLI   IOBUFF+19,X'F2'                          (DELETED) V1L2 A2E04701
*        BC    12,DEV1A                                 (DELETED) V1L2 A2E04702
*        BC    15,ER1A                                  (DELETED) V1L2 A2E04703
*DEV1C   CLI   IOBUFF+12,X'40'                          (DELETED) V1L2 A2E04704
*        BC    8,DEV1A                                  (DELETED) V1L2 A2E04705
*        BC    15,ER1A                                  (DELETED) V1L2 A2E04706
*DEV1A   LA    R2,DEVTAB                                (DELETED) V1L2 A2E04707
*        LA    R1,DEVSHT                                (DELETED) V1L2 A2E04708
*DEV3    CLI   0(R2),X'FF'                              (DELETED) V1L2 A2E04709
```

```
*         BC    BE,DEV4                                    (DELETED) V1L2 A2E04710
*         CLC   8(3,R2),IOBUFF+23      DEVICE ALREADY FOU(DELETED) V1L2 A2E04711
*         BC    BE,DEV54A                                  (DELETED) V1L2 A2E04712
*DEV84A   CLC   0(5,R1),IOBUFF+7       *                  (DELETED) V1L2 A2E04713
*         BC    BE,READ12              *                  (DELETED) V1L2 A2E04714
*         LA    R2,11(R2)                                 (DELETED) V1L2 A2E04715
*         LA    R1,5(R1)                                  (DELETED) V1L2 A2E04716
*         BC    15,DEV3                                   (DELETED) V1L2 A2E04717
*DEV4     CLI   IOBUFF+12,X'40'        IS A 1624P PRESENT(DELETED) V1L2 A2E04718
*         BC    BE,DEV64A              YES,BRANCH         (DELETED) V1L2 A2E04719
*         CLI   IOBUFF+20,C','         IS FORMAT VALID    (DELETED) V1L2 A2E04720
*         BC    7,ER1A                 NO SEND MESSAGE    (DELETED) V1L2 A2E04721
*         CLI   IOBUFF+21,C'X'         *                  (DELETED) V1L2 A2E04722
*         BC    7$ER1A                 INV SEND MESSAGE   (DELETED) V1L2 A2E04723
*         CLI   IOBUFF+22,X'7D'        *                  (DELETED) V1L2 A2E04724
*         BC    7,ER1A                 GO TO SEND MESSAGE(DELETED) V1L2 A2E04725
*         CLI   IOBUFF+26,X'7D'        *                  (DELETED) V1L2 A2E04726
*         BC    7,ER1A                                    (DELETED) V1L2 A2E04727
*         MVC   0(8,R2),IOBUFF+13      MOVE DEVICE ADDRES(DELETED) V1L2 A2E04728
*         MVC   8(3,R2),IOBUFF+23      *                  (DELETED) V1L2 A2E04729
*         MVC   0(5,R1),IOBUFF+7                          (DELETED) V1L2 A2E04730
*         MVI   5(R1),X'FF'                               (DELETED) V1L2 A2E04731
*         MVI   11(R2),X'FF'                              (DELETED) V1L2 A2E04732
*         BC    15,READ1                                  (DELETED) V1L2 A2E04733
*DEV64A   MVC   0(5,R1),IOBUFF+7       MOVE 1624P TO SHOR(DELETED) V1L2 A2E04734
*         MVI   5(R1),X'FF'            *                  (DELETED) V1L2 A2E04735
*         MVI   0(R2),X'40'            *                  (DELETED) V1L2 A2E04736
*         MVC   1(10,R2),0(R2)         *                  (DELETED) V1L2 A2E04737
*         MVI   11(R2),X'FF'           *                  (DELETED) V1L2 A2E04738
*         BC    15,READ1               GOTO READ          (DELETED) V1L2 A2E04739
*READ12   BAL   15,MESSAG                                 (DELETED) V1L2 A2E04740
*         DC    YL2(READ13)                               (DELETED) V1L2 A2E04741
*         BC    15,READ1               GO TO READ         (DELETED) V1L2 A2E04742
*DEV54A   CLC   IOBUFF+13(4),B1442     TEST FOR 1442      (DELETED) V1L2 A2E04743
*         BC    BE,DEV84A              YES BRANCH         (DELETED) V1L2 A2E04744
*         BC    15,READ12              NO,SEND MESS DUPL  (DELETED) V1L2 A2E04745
*B1442    DC    C'1442'                *                  (DELETED) V1L2 A2E04746
*                                                                     A2E04790
DEVICE   DC    C'DEVICE 1622R,1442R,'                              A2E04795
         DC    C'DEVICE 1622R,2540R,'                              A2E04800
         DC    C'DEVICE 1622R,2520R,'                              A2E04805
         DC    C'DEVICE 1622R,2501R,'                              A2E04810
         DC    C'DEVICE 1622P,2540P,'                              A2E04815
         DC    C'DEVICE 1622P,1442P,'                              A2E04820
         DC    C'DEVICE 1622P,2520P,'                              A2E04825
         DC    C'DEVICE 1443W,1403W,'                              A2E04830
         DC    C'DEVICE 1443W,1443W,'                              A2E04835
         DC    C'DEVICE 1621R,2671R,'                              A2E04840
         DC    C'DEVICE 1621P,1442P,'                              A2E04845
         DC    C'DEVICE 1621P,2540P,'                              A2E04850
         DC    C'DEVICE 1621P,2520P,'                              A2E04855
*        DC    C'DEVICE 1621P       '                (DELETED) V1L2 A2E04856
         DC    C'DEVICE 13110,2311A,'                              A2E04860
         DC    C'DEVICE 13111,2311B,'                              A2E04865
         DC    C'DEVICE 13112,2311C,'                              A2E04870
```

```
         DC    C'DEVICE 13113,2311D,'                                  A2E04875
         DC    C'DEVICE 1620C,1052T,'                                  A2E04880
         DC    X'FF'                                                   A2E04885
START    DC    C'START'                                                A2E04890
         DC    X'FF'                                                   A2E04895
         EJECT                                                         A2E04900
*                                                                      A2E04905
*             MESSAGES ERROR                                           A2E04910
*                                                                      A2E04915
*                                                                      A2E04920
CPER     BAL   15,MESSAG                                               A2E04925
         DC    YL2(CPER1)                                              A2E04930
         BAL   R1,READTY              SEND A READ COMMAND              A2E04935
         BC    15,RETRY                                                A2E04940
CPER1    DC    FL1'40'                                                 A2E04945
         DC    C' A235A '                                              A2E04950
         DC    C'CPU1 AND CPU2 '                                       A2E04955
         DC    C'ARE NOT COMPATIBLE'                                   A2E04960
         DC    X'15'                                                   A2E05000
PASTR    DC    FL1'35'                                                 A2E05005
         DC    C' A234I '                                              A2E05010
         DC    C'INVALID '                                             A2E05015
         DC    C'CONTROL INFORMATION'                                  A2E05020
         DC    X'15'                                                   A2E05025
NCONTI   DC    FL1'23'                                           V1L1  A2E05026
         DC    C'       CANNOT CONTINUE'                         V1L1  A2E05027
         DC    X'15'                                             V1L1  A2E05028
ENDSM1   DC    FL1'20'                                                 A2E05030
         DC    C' A232I '                                              A2E05035
         DC    C'END OF SIM20'                                         A2E05040
         DC    X'15'                                                   A2E05045
ENBEG    DC    FL1'22'                                                 A2E05050
         DC    C' A241I '                                              A2E05055
         DC    C'END OF '                                              A2E05060
         DC    C'EDITING'                                              A2E05065
         DC    X'15'                                                   A2E05070
END16    DC    FL1'32'                                                 A2E05075
         DC    C' A233D '                                              A2E05080
         DC    C'DO YOU WANT '                                         A2E05085
         DC    C'ANOTHER FILE'                                         A2E05090
         DC    X'15'                                                   A2E05095
WAI      DC    FL1'18'                                                 A2E05100
         DC    C' A242W '                                              A2E05105
         DC    C'WAIT STATE'                                           A2E05110
         DC    X'15'                                                   A2E05115
READ13   DC    FL1'42'                                                 A2E05120
         DC    C' A230I '                                              A2E05125
         DC    C'DUPLICATION'                                          A2E05130
         DC    C' OF CONTROL '                                         A2E05135
         DC    C'INFORMATION'                                          A2E05140
         DC    X'15'                                                   A2E05145
FEAT9    DC    FL1'30'                                                 A2E05150
         DC    C' A231I '                                              A2E05155
         DC    C'NO INDEX ON '                                         A2E05160
         DC    C'THIS MODEL'                                           A2E05165
```

```
         DC    X'15'                                                    A2E05170
REA2     DC    FL1'34'                                                  A2E05171
         DC    C' A239A '                                               A2E05175
         DC    C'SIM2INF DEVICE NOT DEFINED'                            A2E05180
         DC    X'15'                    *                               A2E05185
MVT31    DC    FL1'34'                                                  A2E05195
         DC    C' A236A '                                               A2E05200
         DC    C'SIM2OUT DEVICE NOT DEFINED '                           A2E05205
         DC    X'15'                    *                               A2E05210
ENDTA    DC    FL1'30'                                                  A2E05215
         DC    C' A240I '                                               A2E05220
         DC    C'END OF TAPE ON SIM2OUT'                                A2E05225
         DC    X'15'                    *                               A2E05230
MVT2     DC    FL1'33'                                                  A2E05235
         DC    C' A237A '                                               A2E05240
         DC    C'SIM2IN DEVICE NOT DEFINED'                             A2E05245
         DC    X'15'                                                    A2E05250
REA19    DC    FL1'34'                                                  A2E05255
         DC    C' A238A '                                               A2E05260
         DC    C'SIM2COM DEVICE NOT DEFINED'                            A2E05265
         DC    X'15'                                                    A2E05270
         EJECT                                                          A2E05275
*                                                                       A2E05280
*             COMPARE CONTROL CARDS WITH TABLES                         A2E05285
*                                                                       A2E05290
COMPAR   LR    WR3,WR1                                                  A2E05295
         BCTR  WR3,0                                                    A2E05300
         STC   WR3,COMPA1+1                                             A2E05305
COMPA1   CLC   IOBUFF(0),0(WR2)                                         A2E05310
         BCR   8,15                                                     A2E05315
         LA    WR2,0(WR2,WR1)                                           A2E05320
         CLI   0(WR2),X'FF'                                             A2E05325
         BC    8,4(15)                                                  A2E05330
         BC    15,COMPA1                                                A2E05335
*             TABLES FOR COMPARISON WITH 'EDI1FILE' INFORMATION         A2E05340
*                                                                       A2E05345
*                                                                       A2E05350
CPUTAB   DC    C'1620/0'                1620 STANDARD                   A2E05355
CP1A     DS    6C                       1620/1 OR 1620/2                A2E05360
CAPACT   DS    6C                       1620/3,1620/5,1620/7            A2E05365
         DC    X'FF'                    END OF 'CPUTAB'                 A2E05370
FEATAB   DC    X'FF'                    FEATURES                        A2E05375
         DS    4C                       INDEX                           A2E05380
         DS    5C                       INDAD                           A2E05385
         DS    5C                       FLOAT                           A2E05390
         DS    5C                       TRANS                           A2E05395
         DS    20C                                                      A2E05400
         DS    5C                       DIVID                           A2E05405
         DS    5C                       DISKV                           A2E05410
         DS    5C                       144LN                           A2E05415
         DS    5C                                                       A2E05420
         DS    C                                                        A2E05425
DEVTAB   DC    X'FF'                     DEVICE                         A2E05430
         DS    10C                      CARD READER                     A2E05435
         DS    11C                      CARD PUNCH                      A2E05440
```

```
         DS    11C                    CONSOLE                    A2E05445
         DS    11C                    PRINTER                    A2E05450
         DS    11C                    PAPER TAPE READER          A2E05455
         DS    11C                    PAPER TAPE PUNCH           A2E05460
         DS    11C                    DISK 1                     A2E05465
         DS    11C                    DISK 2                     A2E05470
         DS    11C                    DISK 3                     A2E05475
         DS    11C                    DISK 4                     A2E05480
         DS    89C                                               A2E05485
DEVSHT   DC    X'FF'                                             A2E05490
         DS    4C                                                A2E05495
         DS    5C                                                A2E05500
         DS    5C                                                A2E05505
         DS    5C                                                A2E05510
         DS    5C                                                A2E05515
         DS    5C                                                A2E05520
         DS    56C                                               A2E05525
         DC    C' '                                              A2E05530
*                                                                 A2E05535
IOBUFF   DS    161C                   BUFFER I/O                 A2E05540
         EJECT                                                   A2E05545
*                                                                 A2E05550
*              SUBROUTINE TO READ SIM-20 FILE                     A2E05555
*                                                                 A2E05560
         CNOP  0,4                                               A2E05565
READSM   SVC   18                                                A2E05570
         DC    C'SIM2IN  '                                       A2E05575
         DC    FL2'80'                                           A2E05580
         DC    A(IOBUFF-1)                                       A2E05585
         CLI   IOBUFF-1,X'07'                                    A2E05590
         BCR   8,15                                              A2E05595
         CLI   IOBUFF-1,X'01'                                    A2E05600
         BC    8,MVT1                                            A2E05605
         CLI   IOBUFF-1,X'03'                                    A2E05610
         BC    8,READSM                                          A2E05615
         BC    15,WAIT2                                          A2E05620
MVT1     BAL   15,MESVC4                                         A2E05625
         DC    YL2(MVT2)                                         A2E05630
         BC    15,WAIT2                                          A2E05635
         EJECT                                                   A2E05640
*                                                                 A2E05645
*                                                                 A2E05650
*              MESSAG SUBROUTINE WHICH SENDS MESSAGES             A2E05655
*                                                                 A2E05660
*                                                                 A2E05665
MESSAG   LH    R1,0(15)                                          A2E05670
         LA    R2,0                                              A2E05675
         IC    R2,0(R1)                                          A2E05680
         BCTR  R2,0                                              A2E05685
         STC   R2,MESSIO+1                                       A2E05690
         LA    R1,1(R1)                                          A2E05695
         STH   R1,MESSIO+4                                       A2E05700
         CNOP  0,4                                               A2E05705
         SVC   18                                                A2E05710
         DC    C'SIM2MES '                                       A2E05715
```

```
MESSIO   DC    FL2'0'                                          A2E05720
         DC    A(0)                                            A2E05725
         CLI   0(R1),X'07'                                     A2E05730
         BC    8,2(15)                                         A2E05735
         BC    15,MESVC4                                       A2E05740
         CNOP  0,4                                             A2E05745
READTY   SVC   18                                              A2E05750
         DC    C'SIM2COM '                                     A2E05755
         DC    FL2'80'                                         A2E05760
         DC    A(IOBUFF-1)                                     A2E05765
         CLI   IOBUFF-1,X'07'                                  A2E05770
         BCR   8,R1                                            A2E05775
         CLI   IOBUFF-1,X'01'                                  A2E05780
         BC    8,REA18                                         A2E05785
         BC    15,WAI12                                        A2E05790
REA18    BAL   15,MESVC4                                       A2E05795
         DC    YL2(REA19)                                      A2E05800
         BC    15,WAI12                                        A2E05805
         END   BEGIN                                           A2E05810
*                                                              A2E05815
*                                                              A2E05820
*                                                              A2E05825
*                                                              A2E05830
*                                                              A2E05835
*                                                              A2E05840
```