```
    NAM    PDS-SYS2N
    OPT    O,NOG
*
*   PDS SYSTEM 2N  CASSETTE DRIVERS  (SYS2NF)
*
*   PROGRAMMED BY ERIC JAMESON
*
*
*
*   COPYRIGHT 1976 SPHERE CORPORATION
*   940 N.   400 EA.;  NORTH SALT LAKE, UTAH  84054
*
*   P. O.  BOX 213;  BOUNTIFUL, UTAH  84010
*
*
*
*
*   SPHERE RESERVES ALL RIGHTS FOR THE REPRODUCTION,
*   DISTRIBUTION AND USE OF THE PDS SOFTWARE.
*   NO COPIES MAY BE MADE OR DISTRIBUTED WITHOUT THE
*   WRITTEN PERMISSION OF SPHERE CORP.
*
*
*
*
*
* THE PROGRAM DEVELOPMENT SYSTEM (PDS SYS2N) IS A SET OF
* PROGRAMS RESIDING ON ERASABLE PROGRAMMABLE READ ONLY
* MEMORY WHICH ALLOW EVEN THE SMALLEST USER TO USE HIS
* SPHERE SYSTEM AS A COMPLETE COMPUTER SYSTEM FOR THE
* DEVELOPMENT OF COMPUTER PROGRAMS.
*   TOWARD THIS END, THE 5 PDS EPROMS CONTAIN A CURSOR
* BASED EDITOR, A MINI-ASSEMBLER, AND THE SPHERE DEBUGGING
* AID (SDA), AS WELL AS A SET OF UTILITY ROUTINES TO DO 16
* BIT MULTIPLY AND DIVIDE, ASCII-TO-BINARY, AND
* BINARY-TO-ASCII ROUTINES, AND ROUTINES TO DO
* INPUT AND OUTPUT TO THE AUDIO CASSETTE.
*
*
*
*   THE SYS2N SOFTWARE IS AN UPGRADE OF THE PDS V3A
* (V3N & V3D) SOFTWARE DESIGNED TO RUN WITH THE CASSETTE
* SYSTEM.  THERE ARE TWO VERSIONS OF THE SYS2 SOFTWARE:
* SYS2N WHICH RUNS WITH THE NEW KEYBOARD AND THE SYS2A
* WHICH RUNS WITH THE ORIGIONAL (KBD/1A) KEYBOARD.  THE
* MAIN DIFFERENCE BETWEEN THE V3A AND SYS2 VERSIONS ARE
* THAT A FIFTH EPROM HAS BEEN ADDED AND THAT THE MINI-
* ASSEMBLER HAS BEEN DELEATED AND REPLACED WITH A SET OF
* COMMANDS TO DO LOADING AND DUMPING OF CASSETTE TAPES.
```

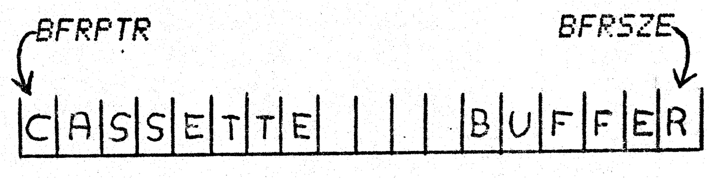
The SYS2NF prom is a version of the SYS2N cassette prom with a software bug fixed. This bug would cause the next block of a multiblock read to be skipped if the checksum on the preceeding block was a 16 and the tape had not been previously used. The change is on page 1Ø where the ESC test and branch now branches to RDHDR1 instead of RDHDR.

```
00051               *              MEMORY MAP
00052               *
00053        0000   TMP     EQU    $00
00054        0002   TMP1    EQU    $02
00055        0004   ARB     EQU    $04        16 BIT ACC.  PSEUDO REG B.
00056        0004   AR3     EQU    $04        _HI BYTE OF ARB.
00057        0005   AR2     EQU    $05        _LO BYTE OF ARB.
00058        0006   ARA     EQU    $06        16 BIT ARITH PSEUDO REG A.
00059        0006   AR1     EQU    $06        _HI BYTE OF ARA.
00060        0007   AR0     EQU    $07        _LO BYTE OF ARA.
00061        0008   DIGIT   EQU    $08        BYTE USED BY ASCBIN FOR TMP.
00062        0009   CSTATS  EQU    $09        CASSETTE I-O STATUS BYTE.
00063        000A   OUTEND  EQU    $0A        END OF OUTPUT BUFFER TEXT.
00064        000C   BUFADR  EQU    $0C        START OF I/O BUFFER (PTR)
00065        000E   BUFEND  EQU    $0E        PTR.  TO END OF I/O BUFFR.
00066        0011   OUTBUF  EQU    $11        START OF OUTPUT BUFFER.
00067        0013   CASNUM  EQU    $13        PHYSICAL CASSETTE NUMBR.
00068        0014   SRCADR  EQU    $14        SOURCE FOR TEXT MOVES.
00069        0016   DSTADR  EQU    $16        DEST.  ADDR.  FOR TEXT MOVE.
00070        001A   ENDMEM  EQU    $1A        LAST ADDRES OF REAL MEMORY.
00071        001C   CSRPTR  EQU    $1C        PTR TO CURSER ON SCREEN.
00072        001E   BUFPTR  EQU    $1E        TEMP PTR USED BY OUTSTR.
00073        0020   BUFFLO  EQU    $20        PTR TO END OF LOW EDIT TXT.
00074        0022   BUFFHI  EQU    $22        PTR TO START OF HI TEXT.
00075        0024   SCNPTR  EQU    $24        PTR.  TO BUFFRD TXT START.
00076        0026   SRCASM  EQU    $26        PTR TO ASSMBLR SOURCE CODE.
00077        002A   ONDVAL  EQU    $2A        HAS ASSMBLR OPERND VALUE.
00078        002C   SYMVAL  EQU    $2C        VALUE PUT IN ASSM.  SYMTBL.
00079        002E   BRKSAV  EQU    $2E        TEMP SAVE FOR BRKPT DATA.
00080        0030   BRKADR  EQU    $30        ADDDRESS OF BREAKPOINT.
00081        0032   EDIT    EQU    $32        0 IF EDITOR IS NOT RUNNING.
00082        0033   BLKNAM  EQU    $33        CASSETTE BLOCK NAME.
00083        0035   IOBUFF  EQU    $35        I-O BUFFER FOR DEBUGGER.
00084        0038   ACIANO  EQU    $38        SYS2N CASSETTE ACIA ADDR.
00085        003A   NOPRNT  EQU    $3A        CASSETTE NAME PRINT FLAG.
00086        003B   BLKTYP  EQU    $3B        CASSETTE BLOCK TYPE CODE.
00087        003C   BFRPTR  EQU    $3C        ADDR.  OF I-O BUFF. FOR CASS.
00088        003E   BFRSZE  EQU    $3E        LENGTH OF CASS.  BUFFER.
00089        0040   PCVAL   EQU    $40        PROGRAM COUNTER FOR ASSM.(Debugger)



00091               *       FOLLOWING ARE VARIABLE VALUES.
00092               *
00093        26F0   TIMEA   EQU    $26F0      TIMER COUNTER.
00094        00B1   ON      EQU    $B1        ACIA VALUE TO TURN ON CASS.
00095        0051   OFF     EQU    $51        ACIA VALUE TO STOP CASS.
00096        0003   ETX     EQU    $03        END-OF-TEXT.
00097        0016   SYN     EQU    $16        SYNCHRONISE.
00098        0017   ETB     EQU    $17        END-OF-TRANSMISSION-BLOCK.
00099        001B   ESC     EQU    $1B        EXCAPE TO NONSTANDARD HDR.
00100        0054   ERR4    EQU    $'T        SET  FOR TRAILER ERROR.
00101        0043   ERR5    EQU    $'C        SET  FOR CHECKSUM ERR.
00102        5161   TIME    EQU    20833      TIME CNTR FOR 1/4 SEC.
00103        0009   TIMCNT  EQU    9          TIMES FOR 2 & 1/4 SECONDS.
0
```

```
               *
FB00                  ORG     $FB00
               *
               *
               *          CASSETTE I-O DRIVERS
               *
               *      THE CASSETTE DRIVERS LOAD AND DUMP A BLOCK OF DATA
               * TO AND FROM THE CASSETTE.   THEY HANDLE BOTH THE HEADER
               * AND TRAILER FORMATTING.
               *      THE DRIVERS ARE SET UP AR A SET OF SUBROUTINES
               * CALLED BY THE EXECUTIVE OR THE USER'S PROGRAMS.
               *
               *
               *
               *       LOW MEMORY ADDRESSES USED BY THE DRIVERS ARE:
               *
               *       CSTATS  AT  09     USED TO STORE INPUT ERR CODE.
               *       BLKNAM  AT  33     2 CHAR.  NAME OF BLOCK.
               *       ACIANO  AT  38     PTR.  TO THE ACIA CURRENTLY USED.
               *       NOPRNT  AT  3A     PRINT FLAG FOR BLOCK NAME.
               *       BFRPTR  AT  3C     START OF CASSETTE I-O BUFFER.
               *       BFRSZE  AT  3E     END OF CASSETTE DATA BUFFER.
               *
               *
               *      THE DRIVERS CAN BE RUN FROM THE PDS-SYS2N SOFTWARE
               * SYSTEM, WHICH CONTAIN THE CASSETTE LOAD AND DUMP
               * COMMANDS IN THE EXEC, OR FROM THE PDS-V3A, WHERE THE
               * DRIVERS ARE CALLED THROUGH THE DEBUGGER.
               *
               *      THE DRIVERS DISPLAY PERTINENT DATA ON THE SCREEN
               * WHEN WRITING OR READING FROM THE CASSETTES.   ON A WRITE,
               * THE CHARACTERS BEING WRITTEN ONTO THE CASSETTE ARE
               * DISPLAYED IN THE SECOND CHARACTER POSITION FROM THE
               * UPPER RIGHT HAND CORNER OF THE SCREEN.   ON A READ, THE
               * CHARACTERS BEING READ IN ARE DISPLAYED IN THE UPPER
               * RIGHT HAND CORNER OF THE SCREEN.   THE NAME OF THE BLOCK
               * CURRENTLY BEING READ OR SEARCHED OVER IS DISPLAYED ON THE
               * RIGHT HAND SIDE OF THE SECOND LINE.   THE ERROR CODE FOR
               * A READ IS DISPLAYED ON THE RIGHT SIDE OF THE THIRD LINE.
               * NO CHANGE IN CHARACTER MEANS THAT THE READ WAS O.K.
               * A "C" MEANS THERE WAS A CHECKSUM ERROR ON THE BLOCK.
               * A "T" MEANS THAT THE WRONG NUMBER OF BYTES WERE READ
               *  INTO THE BUFFER (TRAILER ERROR).   THIS WOULD OF COURSE
               *  IMPLY A CHECKSUM ERROR ALONG WITH THE TRAILER ERROR.
               *
               *      THE CSTATS (CASSETTE STATUS) BYTE WILL CONTAIN A 0
               * IF THE BLOCK READ IN WAS O.K.   IF IT WAS A BAD READ, IT
               * WILL CONTAIN A 54 FOR TRAILER ERRROR OR A 43 FOR A
               * CHECKSUM ERROR CODE UPON EXIT FROM THE ROUTINE.
               *
               *      SETTING THE NOPRNT (NO PRINTING) FLAG TO A 0 WILL
               * STOP THE DISPLAY OF CHARACTERS ON THE SCREEN DURING
               * CASSETTE READ AND WRITE, EXCEPT FOR THE T & C ERROR
               * CODES, WHICH ARE ALWAYS DISPLAYED WHEN THEY OCCUR.
*     IF THE FIRST BYTE OF BLKNAM (BLOCK NAME) IS A 0
* WHEN THE READ BLOCK ROUTINE IS ENTERED, THE NEXT BLOCK
* WILL BE READ FROM TAPE NO MATTER WHAT THE NAME OF THE
* TAPE BLOCK IS.  ON THE SYS2N EXEC, A CONTROL SPACE
* CHARACTER CAN BE TYPED IN AS THE FIRST CHARACTER OF THE
* NAME IN THE LOAD BLOCK COMMAND.  THUS, A (CNTL L)(CNTL
* SPACE)(X) WOULD READ IN THE NEXT BLOCK ON THE TAPE.
*
*
*
*     THE TAPE FORMAT IS:
*       SYN  16
*       SYN  16
*       SYN  16
* SOH   ESC  1B
*       HI BYTE OF 16 BIT BLOCK LENGTH } 1 less than # data char on tape
*       LOW BYTE OF BLOCK LENGTH       }
*       FIRST CHAR OF BLOCK NAME
* STX   SECOND CHARACTER OF NAME
*       DATA
*       .
*       .
*       .
*       DATA
* ETX   ETB  17
*       CHECKSUM
*       CHECKSUM
*       CHECKSUM
*       CHECKSUM
*
*
*
*     THE CHECKSUM IS CALCULATED FROM THE DATA, WHICH
* IS READ IN FROM THE CASSETTE BUFFER RESIDING IN MEMORY.
*
*
*     THE FORMAT FOR THE CASSETTE BUFFER IS:
*
```



```
*
*
*     WHEN DATA IS READ IN, THE READ ROUTINE SETS 'BFRSZE'
* TO POINT TO THE LAST CHARACTER READ INTO THE BUFFER.
* NOTE THAT THERE IS NO OVERFLOW CHECK WHEN DATA IS READ
* INTO MEMORY.  ON OUTPUT TO THE CASSETTE, THE BLOCK
* LENGTH IS CALCULATED FROM THE BFRPTR AND BFRSZE POINTERS.
*
*
*     THE READ BLOCK AND WRITE BLOCK ROUTINES AUTOMATICALLY
* TURN ON AND OFF THE CASSETTE.
```

```
*
*       THE ADDRESSES OF THE ROUTINES FOR CONTROLLING
*       THE CASSETTE ARE AS FOLLOWS:
*
*       INTLZ  - FB00 : INITIALIZES THE ACIA FOR USE.
*       WRTBLK - FB2D : WRITES A  FORMATTED BLOCK TO CASSETTE
*       RDBLK  - FB91 : READS IN A FORMATTED BLOCK OF TAPE
*       WRTMOD - FB2F : WRITES A BLOCK TO A RUNNING CASSETTE
*       RDMOD  - FB93 : READS A BLOCK FROM A RUNNING TAPE.
*       CASOUT - FB62 : WRITES OUT ONTO TAPE THE BYTE IN A.
*       CASIN  - FB7E : READS IN A BYTE INTO ACCUM.  A
*       TURNON - FB77 : TURNS ON THE CASSETTE DRIVE.
*       TRNOFF - FBB0 : TURNS OFF THE CASSETTE DRIVE WHEN THE
*                       POINTR TO ACIA IS PASSED IN THE X REG.
*
*      NOTE THAT ALL THE ABOVE ROUTINES ARE SUBROUTINES THAT
*       ARE ENTERED BY A JSR OR BSR CALL.
*
*
*
*
*                USING THE CASSETTE DRIVERS.
*
*       IF THE EPROMS ON THE CPU BOARD ARE THE SYS2N OR THE
* SYS2D PROMS, THE DRIVERS ARE USED BY THE I (INITIALIZE),
* L (LOAD FROM CASSETTE) AND S (STORE ONTO CASSETTE)
* COMMANDS THAT ARE A PART OF THE SYS-2 EXECUTIVE.
*
*       IF THE CPU PROMS BEING USED ARE THE PDS-V3A OR THE
* PDS-V3N OR V3D PROM SETS, THEN THE USER MUST SET UP THE
* POINTERS AND CALL THE ROUTINES HIMSELF, USING THE
* DEBUGGER ON THE CPU PROMS.  (NOTE THAT THE V3A AND V3N
* DEBUGGERS VARY, FOR INSTANCE THE V3A GO COMMAND IS A 'G'
* WHILE THE V3N GO COMMAND IS A 'CNTL G'.  REFER TO THE
* USERS MANUAL FOR DETAILES.)
*       THE FIRST THING TO DO IS OPEN THE LOW MEMORY LOCATIONS
* USED BY THE DRIVERS AS FLAGS AND POINTERS AND INITIALIZE
* THE LOCATIONS.  THE ACIANO, BLKNAM, BFRPTR AND BFRSZE
* SHOULD NOW BE GIVEN VALUES.
*       ACIANO WOULD POINT TO F050 FOR THE FIRST CASSETTE AND
* TO F060 FOR THE SECOND CASSETTE DRIVE.
*       A SIMPLE PROGRAM TO CALL THE ROUTINES SHOULD NOW BE
* WRITTEN.  OPEN LOCATION 900.  TYPE IN THROUGH THE DEBUGGER
* THE INSTRUCTIONS JSR BDXX, JMP FE4F.  XX IS THE SECOND BYTE
* OF THE DESIRED DRIVER ROUTINE, I.E. 00 FOR INITIALLIZATION
* OR 91 FOR READING A BLOCK OR 2D FOR WRITING A BLOCK.  THUS,
* TO INITIALIZE AN ACIA, ACIANO WOULD BE LOADED WITH
* THE ADDRESS OF THE ACIA AND THE USER WOULD THEN JUMP TO THE
* ROUTINE AT 900 BY OPENING 900 AND JUMPING TO IT WITH
* THE 'G' COMMAND.  THE ROUTINE WOULD BE AS FOLLOWS:
*
*CALL 900  BD     JSR
*     901  FB     HI BYTE OF ROUTINE AADDRESS
*     902  00     LOW BYTE OF DRIVER ADDRESS.
*     903  7E     JMP
*     904  FE     ADDRESS OF THE DEBUGGER ON
*     905  4F     THE PDS V3A PROM SET
*     THE ACIA WOULD NOW BE INITIALIZED AND CONTROL WOULD
* HAVE RETURNED TO THE DEBUGGER.   TO READ IN A BLOCK,
* THE NAME WOULD BE PUT IN BLKNAM AND LOCATION 902 WOULD
* BE CHANGED TO 91.   THE USER WOULD THEN  JUMP TO 900.
*
*     TO TEST THE CASSETTE, SET BFRPTR TO E060 AND BFRSZE
* TO E0DF.   THIS WILL ALLOW THE USER TO WRITE OUT DATA
* FROM THE FOURTH, FIFTH, SIXTH AND SEVENTH LINE OF THE
* CRT DISPLAY AND THEN READ IT BACK ONTO THE DISPLAY.   DATA
* CAN BE TYPED ONTO THE SCREEN BY OPENING A LOCATION WITH
* THE DEBUGGER AND THEN MOVING THE CURSOR AROUND THE SCREEN
* TO CHANGE THE CHARACTERS.   THIS IS POSSIBLE BECAUSE THE
* ROUTINE TO INPUT AN ADDRESS CALLS THE EDITOR FOR INPUT.
*
*     THE DRIVER ROUTINES CAN ALSO BE USED TO PERFORM
* I-O WITH A MODEM OR TELETYPE.   THE MAIN HARDWARE
* DIFFERENCE BETWEEN THE CASSETTE AND MODEM/TTY IS THAT THE
* CASSETTE HAS A DIVIDE BY 16 CLOCK AND IS UNIDIRECTIONAL
* WHILE THE MODEM/TTY HAVE A DIVIDE BY 64 CLOCK AND CAN BE
* BIDIRECTIONAL.   BECAUSE OF THE CLOCK CHANGE THE ACIA MUST
* BE TURNED ON WITH A DIFFERENT VALUE BEFORE THE READ BLOCK
* OR WRITE BLOCK ROUTINES ARE ENTERED.   TO TURN ON THE ACIA
* STORE THE VALUE 'B2' INTO LOCATION F050 OR F060.   ONCE
* IT IS TURNED ON, EITHER CASIN OR CASOUT MAY BE CALLED
* REPEATEDLY OR WRITE MODEM BLOCK (WRTMOD) OR READ MODEM
* (RDMOD) CAN BE CALLED ONCE.   WRTMOD & RDMOD ARE THE SAME
* AS WRTBLK & RDBLK EXCEPT THEY DO NOT TURN ON THE ACIA.   A
* PROGRAM TO READ IN A BLOCK OF DATA FROM A MODEM OR TTY
* WOULD THUS BE (PLACED BEFORE THE 'CALL' ROUTINE):
*      8FA  DE 38  LDX     ACIANO   LOADS ACIA POINTTER.
*      8FC  86 B2  LDA A   #$B2     LOADS STARTUP VALUE.
*      8FE  A7 00  STA A   0,X      PUTS START CODE INTO ACIA.
* LOCATION 902 WOULD NOW BE 2F FOR WRITING AND 93 TO READ.
*      TO READ IN A CHARACTER FROM THE TELETYPE TURN ON THE
* ACIA AND GO TO THE FOLLOWING ROUTINE:
*      TTYIN  JSR    CASIN   READS IN A CHAR FROM KEYBOARD.
*             JMP    CASOUT  TYPES OUT CHAR ON PRINTER.
* THE RS232 SHOULD HAVE BEEN STRAPPED TO HALF DUPLEX.
*
*
*      IT IS TO BE STRESSED THAT  THE RELIABILITY OF THE
* CASSETTE CONTROLLER DEPENDS ON THE ADJUSTMENT OF THE
* TRIMMER ON THE SIM BOARD.   IF THE TRIMMER IS OUT OF
* ADJUSTMENT THE DATA WILL NOT READ IN PROPERLY.   BESIDES
* USING THE OSCILLOSCOPE TO ADJUST THE TRIMMER, IT CAN ALSO
* BE ADJUSTED BY READING IN A STRING OF SINGLE CHARACTERS
* FROM  THE CASSETTE AND ADJUSTING IT UNTIL THE CHARACTERS
* SYNC IN PROPERLY.   THE BEST CHARACTER TO USE IS A STRING
* OF 'U'S.   TO READ IN THE STRING FOR TESTING USE, WRITE A
* LOOP TO GET A CHARACTER FROM THE CASSETTE AND THEN
* DISPLAY THAT CHARACTER.   A SAMPLE ROUTINE TO DO THIS IS:
*    A   JSR   $FB7E   LOADS A WITH CASSETTE CHAR.
*        JSR   $FCAD   PDS-V3A PUTCHR ROUTINE.
*        TST   $F001   TESTS KEYBOARD FOR A KEY.
*        BPL   A       SKIPS BACK IF NO INPUT.
* THE ABOVE ROUTINE WOULD INPUT CHARACCTERS UNTIL A KEY
* ON THE KEYBOARD WAS DEPRESSED.
0
```

B2 will produce a signel of eight data bits and 2 stop bits. If the TTY operates on a different code, look up the proper initialization value in the ACIA section of the chip description appendex.

```
                   *       INTLZ INITIALIZES THE ACIA CONTROLLER FOR
                   *       A SPECIFIC TAPE UNIT AT MOUNT TIME.
                   *
FB00 DE 38         INTLZ   LDX     ACIANO    X GETS ACIA ADDRESS.       INTLZ   LDAA  #$13
FB02 86 13                 LDA A   #$13      RESETS THE ACIA.                   BSR   ACIAST
FB04 A7 00                 STA A   0,X                                  ~~TRNOFF  LDAA  #OFF~~
FB06 86 51                 LDA A   #OFF      SETS ACIA TO               ~~ACIAST  LDX   ACIANO~~
FB08 A7 00                 STA A   0,X       /16 2 STOPS. BIT FORMAT.           ~~STAA  0,X~~
FB0A 39                    RTS                                                  ~~RTS~~
                                                                                BRA   TRNOFF

                   *       WRTHDR FORMATS THE HEADER ON THE TAPE.
                   *
FB0B 86 16         WRTHDR  LDA A   #SYN      PUTS SYNC CHARS ONTO TAPE.
FB0D 8D 53                 BSR     CASOUT$   .
FB0F 8D 51                 BSR     CASOUT    .
FB11 8D 4F                 BSR     CASOUT    .
FB13 86 1B                 LDA A   #ESC      . Soh
FB15 8D 4B                 BSR     CASOUT    .
FB17 96 3E                 LDA A   BFRSZE    FOLLOWING OUTPUTS LENGTH.
FB19 D6 3F                 LDA B   BFRSZE+1  LOADS LO BYTE OF END PTR.
FB1B D0 3D                 SUB B   BFRPTR+1  SUBS LO BYTE OF BEGIN PTR.
FB1D 92 3C                 SBC A   BFRPTR    SUBS HI BYTE OF START PTR.
FB1F 8D 41                 BSR     CASOUT    OUTPUTS HI LENGTH BYTE.
FB21 17                    TBA               LOADS LO BYTE OF LENGTH.
FB22 8D 3E                 BSR     CASOUT    OUTPUTS LOW LENGTH BYTE.
FB24 96 33                 LDA A   BLKNAM    PUTS OUT NAME OF BLOCK.
FB26 8D 3A                 BSR     CASOUT    .
FB28 96 34                 LDA A   BLKNAM+1  .
FB2A 8D 36                 BSR     CASOUT    PUTS OUT LAST OF NAME.   } BRA CASOUT
FB2C 39                    RTS               RETURNS BACK TO WRTBLK.  }

                   *       WRTBLK WRITES OUT A BLOCK OF DATA TO THE CASSETTE.
                   *
FB2D 8D 48         WRTBLK  BSR     TURNON    TURNS ON THE CASSETTE.
                   *        FOLLOWING WAITES FOR CASSETTE TO GET UP TO SPEED.
FB2F C6 09         WRTMOD  LDA B   #TIMCNT   LOADS TIME LOOP COUNTER.
FB31 CE 5161       TIME1   LDX     #TIME     MASTER TIME LOOP (1/4 SEC).
FB34 09            TIME2   DEX               COUNTS CYCLES OF LOOP.
FB35 26 FD                 BNE     TIME2     TESTS FOR FIRST TIME OUT.
FB37 5A                    DEC B   -         COUNTS TIMES IN LOOP.          2 1/4 sec
FB38 26 F7                 BNE     TIME1     SKIPS BACK UNTIL DONE.
                   *        THE TIME LOOP IS NOW FINISHED.
FB3A 8D CF                 BSR     WRTHDR    WRITES HEADER ON THE TAPE.
FB3C 8D 10                 BSR     WRTBFR    WRITES OUT BUFFER DATA.
                   *        FOLLOWING WRITES THE TRAILER OUT ONTO THE TAPE.
FB3E 86 17         WRTTLR  LDA A   #ETB      OUTPUTS END-OF-BLOCK CHAR.
FB40 8D 20                 BSR     CASOUT    ETB IS DISPLAYED AS A "W".
FB42 17                    TBA               A GETS CHECKSUM FROM B.
FB43 8D 1D                 BSR     CASOUT    OUTPUTS THE CHECKSUM.
FB45 8D 1B                 BSR     CASOUT    OUTPUTS TRAILER FILLER
FB47 8D 19                 BSR     CASOUT    BYTES.
FB49 8D 17                 BSR     CASOUT4   .
                   *        END OF TRAILER WRITING ROUTINE.
FB4B 8D 63                 BSR     TRNOFF    HALTS CASSETTE DRIVE.  } BRA TRNOFF      (-1)
FB4D 39                    ~~RTS~~           -                      }
```

```
00387                 *       WRTBFR WRITES OUT THE CONTENTS  OF THE
00388                 *       BUFFER ONTO THE CASSETTE TAPE.
00389                 *
00390                 *
00391 FB4E 5F         WRTBFR CLR B   -         INIT CHECKSUM COUNT.
00392 FB4F DE 3C             LDX     BFRPTR    ACIA CONTROL MASK.
00393 FB51 A6 00      WBFR1  LDA A   0,X       LOADS CHAR.  FROM BUFFER.
00394 FB53 DF 00             STX     TMP       SAVES BUFFER PTR.
00395 FB55 8D 0B             BSR     CASOUT    PUTS CHAR ONTO CASSETTE.
00396 FB57 1B                ABA               A GETS A+B.
00397 FB58 16                TAB               B GETS A+B.
00398 FB59 DE 00             LDX     TMP       RESTORES PTR.  INTO BUFFR.
00399 FB5B 9C 3E             CPX     BFRSZE    TESTS IF BUFFER EMPTY.
00400 FB5D 27 17             BEQ     CSOEXT    EXITS WHEN EMPTY.
00401 FB5F 08                INX               INC POINTER.
00402 FB60 26 EF             BNE     WBFR1     SKIPS BACK IF CHARS.  LEFT.



00404                 *       CASOUT TAKES THE CHAR IN A AND PUTS IT OUT
00405                 *       ONTO THE CASSETTE TAPE.
00406                 * ~~CASOUT4~~ BSR    CASOUT
                        CASOU4  BSR    CASOUT2
00407                 * CASOU2  BSR    CASOUT
00408 FB62 36         CASOUT PSH A   .         SAVES CHAR TO READ OUT.
00409 FB63 DE 38             LDX     ACIANO    X GETS PHYSICAL ACIA ADDR.
00410 FB65 86 02             LDA A   #2        LOADS CONTROL TEST BITS.
00411 FB67 A5 00      CASO1  BIT A   0,X       TESTS IF ACIA BUFFER EMPTY.
00412 FB69 27 FC             BEQ     CASO1     LOOPS BACK UNTIL READY.
00413 FB6B 32                PUL A   .         GETS ORIG CHAR.
00414 FB6C A7 01             STA A   1,X       STORS CHAR INTO ACIA BUFFER.
00415 FB6E 7D 003A           TST     NOPRNT    TESTS IF PRINTOUT ALLOWED.
00416 FB71 27 03             BEQ     CSOEXT    SKIPS PRINTING IF A 0.
00417 FB73 B7 E01E           STA A   $E01E     DISPLAYS CHAR ON SCREEN.
00418 FB76 39         CSOEXT RTS



00420                 *        TURNON TURNS ON THE CASSETTE DRIVE.
00421                 *
00422                 *
00423 FB77 DE 38      TURNON LDX     ACIANO    LOADS CASSETTE ACIA ADDRESS.
00424 FB79 86 B1             LDA A   #ON
00425 FB7B A7 00             STA A   0,X       TURNS ACIA ON.
00426 FB7D 39                RTS

                      TURNON  LDAA   # ON
                              BRA    ACIAST     INTL  LDA  #13
                                                      BSR  ACIAST
                      TURNOFF LDAA   # OFF
                              ~~BRA~~    ~~ACIAST~~
                      ACIAST  LDX    ACIANO
                              STAA   0,X
                              RTS
```

```
00429                    *        CASIN READS IN A CHARACTER FROM THE CASSETTE TAPE
00430                    *        INTO THE A ACCUMULATOR.
00431                    *
00432                    *
00433 FB7E DE 38    CASIN  LDX     ACIANO    X GETS THE ACIA ADDRESS.
00434 FB80 86 01           LDA A   #1        LOADS TEST BITS.
00435 FB82 A5 00    CASIN1 BIT A   0,X       TESTS IF ACCIA BUFFR FULL.
00436 FB84 27 FC           BEQ     CASIN1    SKIPS BACK IF NOT IN YET.
00437 FB86 A6 01           LDA A   1,X       LOADS IN CHAR FROM TAPE.
00438 FB88 7D 003A         TST     NOPRNT    TESTS IF PRINT IS OFF.
00439 FB8B 27 03           BEQ     CINEXT    SKIPS DISPLAYING IF 0.
00440 FB8D B7 E01F         STA A   #E01F     DISPLAYS CHR ON TV.
00441 FB90 39       CINEXT RTS               .



00443                    *        RDBLK READS IN A BLOCK FROM THE CASSETTE
00444                    *        TAPE INTO BUFFER MEMORY.
00445                    *              3 has checksum in it
00446                    *
00447 FB91 8D E4    RDBLK  BSR     TURNON    TURNS ON TAPE DRIVE.
00448 FB93 8D 20    RDMOD  BSR     RDHDR     READS IN THE HEADER.
00449 FB95 8D 55           BSR     RDBFR     READS IN DATA INTO BUFFER.
00450                    *         FOLLOWING READS IN TRAILER AND CHECKS CHECKSUM.
00451 FB97 4F       RDTLR  CLR A   -         LOADS A 0 FOR A GOOD READ.
00452 FB98 97 09           STA A   CSTATS    SETS STATUS BYTE TO NO ERR.
00453 FB9A 8D E2           BSR     CASIN     INPUTS END-OF-BLOCK CHAR.
00454 FB9C 81 17           CMP A   #ETB      ETB DISPLAYS AS A "W".
00455 FB9E 27 04           BEQ     RTLR1     SKIPS IF NO ETB ERROR.
00456 FBA0 86 54           LDA A   #ERR4     LOADS TRAILER ERROR CODE.
00457 FBA2 20 07           BRA     RTLR2     SKIPS TO STORE ERROR CODE.
00458 FBA4 8D D8    RTLR1  BSR     CASIN     READS IN CHECKSUM.
00459 FBA6 11              CBA               TESTS CHECKSUM.
00460 FBA7 27 07           BEQ     TRNOFF    SKIPS IF OK.
00461 FBA9 86 43           LDA A   #ERR5     SETS CHECKSUM ERROR CODE.
00462 FBAB 97 09    RTLR2  STA A   CSTATS    SETS ERROR STATUS BYTE.
00463 FBAD B7 E05F         STA A   #E05F     DISP.  ERR CODE ON SCREEN.
00464                    *         END OF TRAILER READ IN.
00465                    *         FOLLOWING TURNS OFF THE CASSETTE DRIVE.
00466 FBB0 86 51    TRNOFF LDA A   #OFF      LOADS COMMAND TO TURN          BRA TRN OFF
00467 FBB2 A7 00           STA A   0,X       OFF ACIA CASSETTE DRIVE.
00468 FBB4 39              RTS               .

                           LDX  ACIANO
```

```
00471                    *       RDHDR FINDS THE START OF THE BLOCK ON THE TAPE,
00472                    *       FIGURES THE BUFFER END AND CHECKS THE NAME.
00473                    *
00474                    *
00475 FBB5 8D C7        RDHDR   BSR     CASIN     FOLLOWING FINDS THE
00476 FBB7 81 16        RDHDR1  CMP A   #SYN      START OF THE HEADER.
00477 FBB9 26 FA                BNE     RDHDR     .
00478 FBBB 8D C1                BSR     CASIN     TESTS FOR HEADR CHAR.
00479 FBBD 81 1B                CMP A   #ESC      START-OF-HEADER.
00480 FBBF 26 F6                BNE     RDHDR1    GOES BACK IF NOT GOOD HDR.
00481 FBC1 8D BB                BSR     CASIN     READS IN HI BYTE OF LEN.
00482 FBC3 16                   TAB               SAVES HI LENGTH BYTE.
00483 FBC4 8D B8                BSR     CASIN     INPUTS LO SIZE (LEN) BYTE.
00484 FBC6 9B 3D                ADD A   BFRPTR+1 FORMS POINTER TO THE
00485 FBC8 D9 3C                ADC B   BFRPTR    TOP BYTE OF THE BUFFER.
00486 FBCA 97 3F                STA A   BFRSZE+1 SAVES THE HI BUFF.  PTR.
00487 FBCC D7 3E                STA B   BFRSZE    TO THE CASSETTE BUFFER.
00488 FBCE 8D AE                BSR     CASIN     READS IN BLOCK NAME.
00489 FBD0 36                   PSH A   -         SAVES FIRST CHAR OF NAME.
00490 FBD1 8D AB                BSR     CASIN     READ IN SECOND CHAR INTO A.
00491 FBD3 D6 3A                LDA B   NOPRNT    TESTS IF PRINT IS OFF.
00492 FBD5 33                   PUL B   -         RESTORES FIRST NAME CHAR.
00493 FBD6 27 06                BEQ     RHDR1     SKIPS IF PRINT FLAG IS 0.
00494 FBD8 F7 E03E              STA B   $E03E     DISPLAYS BLOCK NAME ON
00495 FBDB B7 E03F              STA A   $E03F     THE CRT SCREEN.
00496 FBDE 7D 0033      RHDR1   TST     BLKNAM    TESTS IF NAME IS CHECKED.
00497 FBE1 27 08                BEQ     RHDR2     SKIPS IF NO NAME CHECK.
00498 FBE3 D1 33                CMP B   BLKNAM    TESTS FIRST CHAR OF NAME.
00499 FBE5 26 CE                BNE     RDHDR     SKIPS BACK IF BAD NAME
00500 FBE7 91 34                CMP A   BLKNAM+1 TESTS SECOND NAME CHAR.
00501 FBE9 26 CA                BNE     RDHDR     SKIPS BACK IF BAD NAME.
00502 FBEB 39           RHDR2   RTS               .




00504                    *       RDBFR READS DATA INTO THE MEMORY BUFFER FROM
00505                    *       THE CASSETTE.
00506                    *
00507                    *


00508 FBEC 5F          RDBFR   CLR B   -         INIT B FOR CHECKSUM.
00509 FBED DE 3C                LDX     BFRPTR    LOADS START OF BUFFER.
00510 FBEF DF 00        RBFR1   STX     TMP       .
00511 FBF1 8D 8B                BSR     CASIN     A GETS CHAR READ IN.
00512 FBF3 DE 00                LDX     TMP       X GETS BUFFER PTR.
00513 FBF5 A7 00                STA A   0,X       STORS CHAR INTO BUFFER.
00514 FBF7 1B                   ABA               A GETS A+B.
00515 FBF8 16                   TAB               B GETS A.
00516 FBF9 9C 3E                CPX     BFRSZE    TESTS IF BUFFER FULL.
00517 FBFB 27 EE                BEQ     RHDR2     SKIPS TO EXIT IF ALL IS IN.
00518 FBFD 08                   INX               INC TO NEXT CHAR POSITIN.
00519 FBFE 26 EF                ~~BNE~~BRA RBFR1  GOES BACK IF ANY LEFT.
                                END
```