```
                                .PREL
                                .IDENT  BITBANG
                                .INSERT BBEQU.ASM
                                .DEFINE WASTE[TIME,%LAB]=[
                                MVI     A,TIME
                        %LAB:   DCR     A
                                JRNZ    %LAB]
0099                    BANGER=99H      ; BIT BANGER IO PORT
                        ; *****
                        ; *
                        ; * COMMAND TO READ BLOCK INTO MEMORY
                        ; * GET variable
                        ; *
                        ; *****
0000'  CD 0000:04       GETC::  CALL    TSTVFF#
0003'  CD 0013'                 CALL    TAPGET
0006'  F7                       RST     RSTFIN
                        ; *****
                        ; *
                        ; * LOAD PROGRAM INTO MEMORY
                        ; *
                        ; *****
0007'                   LOADC::
0007'  21 0000:05               LXI     H,TEXT# ; HL=TEXT AREA START ADD
                        R
000A'  CD 0013'                 CALL    TAPGET
000D'  22 0000:06               SHLD    TXTUNF# ; SET NEW TXTUNF
0010'  C3 0000:07               JMP     CLRSCR# ; ENTER CLEAR TO END
                        ; *****
                        ; *
                        ; * READ BLOCK INTO MEMORY
                        ; * HL=READ ADDRESS
                        ; *
                        ; *****
0013'                   TAPGET:
0013'  D5                       PUSH    D
0014'  F3                       DI
0015'                   ..SENW:
0015'  CD 0035'                 CALL    INCHAR          ; AWAIT SENTINEL
                         CHARACTER
0018'  79                       MOV     A,C
0019'  28FA                     JRZ     ..SENW
001B'  FEA5                     CPI     0A5H
001D'  20F6                     JRNZ    ..SENW
001F'  11 4000                  LXI     D,4000H         ; DE=FEEDBACK ST
                        ORE ADDR
0022'                   ..CHRL:
0022'  D5                       PUSH    D
0023'  CD 0035'                 CALL    INCHAR
0026'  D1                       POP     D
0027'  2809                     JRZ     ..DONE
0029'  71                       MOV     M,C
002A'  79                       MOV     A,C
002B'  12                       STAX    D               ; GIVE FEEDBACK
                        ON SCREEN
```

```
002C'  13                      INX     D               ; BUMP FEEDBACK
                       ADDR
002D'  CBA2                    RES     4,D             ; CONSTRAIN TO 4
                       000H-4FFFH
002F'  23                      INX     H
0030'  18F0                    JMPR    ..CHRL
0032'                  ..DONE:
0032'  FB                      EI
0033'  D1                      POP     D
0034'  C9                      RET
                       ; *****
                       ; *
                       ; * SUBROUTINE TO INPUT A CHARACTER
                       ; * RETURNS CHARACTER IN C
                       ; * AND STATUS OF NONZERO UNLESS A TIMEOUT HAPPE
                       NED
                       ; * IN WHICH CASE ZERO STATUS IS RETURNED
                       ; *
                       ; *****
0035'                  INCHAR:
0035'  01 0810                 LXI     B,810H  ; B=BIT CTR, C=TIMEOUT F
                       ACTOR
0038'  CD 0049'        ..SBW:  CALL    INBIT   ; AWAIT START BIT
003B'  2804                    JRZ     ..GETL
003D'  0D                      DCR     C       ; NOT YET - DCR TIMEOUT
003E'  20F8                    JRNZ    ..SBW   ; IF COUNTED DOWN
0040'  C9                      RET             ; RETURN ZERO SET
0041'  CD 0049'        ..GETL: CALL    INBIT
0044'  0F                      RRC             ; BIT GOT TO CY
0045'  CB19                    RARR    C       ; SHIFT INTO C HO
0047'  10F8                    DJNZ    ..GETL
                       ; NOW FALL INTO INBIT TO EAT THE STOP BIT
0049'                  INBIT:
0049'  DB99                    IN      BANGER  ; WAIT TILL WE GET A TRA
                       NSITION
004B'  5F                      MOV     E,A
004C'  DB15            ..INBW: IN      15H     ; CHECK FOR ABORT
004E'  A7                      ANA     A
004F'  C2 0000:08              JNZ     INIT0#  ; HALT SET?
0052'  DB99                    IN      BANGER
0054'  AB                      XRA     E
0055'  0F                      RRC
0056'  30F4                    JRNC    ..INBW
                               WASTE   30      ; WAIT UNTIL SAMPLE POIN
                       TI
0058'  3E1E            +       MVI     A,30
005A'  3D              +..0001: DCR    A
005B'  20FD            +       JRNZ    ..0001
005D'  DB99                    IN      BANGER
005F'  AB                      XRA     E       ; COMPARE TO OLDER STUFF

0060'  E601                    ANI     1
0062'  C8                      RZ              ; 0 IF TRANSITION HAPPEN
                       ED
                               WASTE   29      ; ELSE WAIT UNTIL MIDDLE
```

```
                          OF NEXT CYCLE[
0063'   3E1D            +         MVI     A,29
0065'   3D              +..0002:  DCR     A
0066'   20FD            +         JRNZ    ..0002]
0068'   3C                        INR     A       ; RETURN VAL OF 1
0069'   C9                        RET
                        ; *****
                        ; *
                        ; * COMMAND TO WRITE OUT THE PROGRAM
                        ; *
                        ; *****
006A'                   SAVEC::
006A'   D5                        PUSH    D
006B'   2A 0000:06                LHLD    TXTUNF#
006E'   11 0000:05                LXI     D,TEXT# ; DE=START
0071'   A7                        ANA     A
0072'   ED52                      DSBC    D
0074'   EB                        XCHG            ; HL=ADDR,DE=SIZE
0075'   1815                      JMPR    SAVEE
                        ; *****
                        ; *
                        ; * COMMAND TO WRITE OUT A BLOCK OF WORDS ON TAP
                        E
                        ; * PUT variable,#words
                        ; *
                        ; *****
0077'                   PUTC::
0077'   CD 0000:04               CALL    TSTVFF# ; GRAB A VARIABLE
007A'   E5                        PUSH    H
007B'   CF                        TSTCC   COMMA,BADSAV[   RST     1
007C'   2C              +         .BYTE   COMMA
007D'   20              +         .BYTE   BADSAV-.-1
                        +]
007E'   D7                        RST     RSTEXP  ; GET NUMBA OF BYTES
007F'   7C                        MOV     A,H     ; REJECT BIZARRE VALUES
0080'   B5                        ORA     L       ; LIKE ZERO
0081'   CA 0000:09               JZ      QHOW#
0084'   CB7C                     BIT     7,H     ; OR NEGATIVE VALUES
0086'   C2 0000:09               JNZ     QHOW#
0089'   29                       DAD     H       ; CONVERT WORDS TO BYTES

008A'   EB                       XCHG            ; COUNT=DE
008B'   E3                       XTHL            ; STK=SCAN, HL=START
008C'                   SAVEE:
008C'   F3                       DI
008D'   CD 00D7'                 CALL    LEADER  ; WRITE OUT LEADER
0090'   0EA5                     MVI     C,0A5H  ; WRITE SENTINEL
0092'   CD 00AC'                 CALL    OUTBYT
0095'   CD 00A1'                 CALL    WRBLOC  ; AND DATA BLOCK
0098'   CD 00D7'                 CALL    LEADER  ; THEN TRAILER
009B'   FB                       EI
009C'   D1                       POP     D
009D'   F7                       RST     RSTFIN
009E'   C3 0000:0A      BADSAV:  JMP     QWHAT#
                        ; *****
```

```
                        ; *
                        ; * SUBROUTINE TO WRITE OUT BLOCK OF BYTES
                        ; * HL=LIST, DE=# OF BYTES
                        ; *
                        ; *****
00A1'                   WRBLOC:
00A1'                   ..BYTL:
00A1'   4E                      MOV     C,M
00A2'   CD 00AC'                CALL    OUTBYT
00A5'   23                      INX     H
00A6'   1B                      DCX     D
00A7'   7A                      MOV     A,D
00A8'   B3                      ORA     E
00A9'   20F6                    JRNZ    ..BYTL
00AB'   C9                      RET
                        ; *****
                        ; *
                        ; * WRITE OUT A BYTE ONTO TAPE
                        ; *
                        ; * THIS ROUTINE IS TIME SENSITIVE! CHANGE CAREF
                        ULLY!
                        ; *
                        ; *****
00AC'                   OUTBYT:
00AC'   CD 00F7'                CALL    WRZERO          ; WRITE START BI
                        T
                                WASTE   19              ; FINISH START B
                        IT[
00AF'   3E13            +       MVI     A,19
00B1'   3D              +..0003: DCR    A
00B2'   20FD            +       JRNZ    ..0003]
00B4'   0608                    MVI     B,8             ; WRITE OUT 8 BI
                        TS
00B6'   CB09             ..WRL: RRCR    C       ; SHIFT NEXT BIT TO CY
00B8'   380A                    JRC     ..WR1   ; BRANCH ON BIT VALUE
00BA'   CD 00F7'                CALL    WRZERO  ; THIS GUY ZERO
                                WASTE   19[
00BD'   3E13            +       MVI     A,19
00BF'   3D              +..0004: DCR    A
00C0'   20FD            +       JRNZ    ..0004]
00C2'   1808                    JMPR    ..WRE
00C4'                    ..WR1:
                        ; WRITE ONE BIT
00C4'   CD 00ED'                CALL    WRONE
                                WASTE   42[
00C7'   3E2A            +       MVI     A,42
00C9'   3D              +..0005: DCR    A
00CA'   20FD            +       JRNZ    ..0005]
00CC'                    ..WRE:
00CC'   10E8                    DJNZ    ..WRL   ; LOOP TILL BYTE DONE
00CE'   CD 00ED'                CALL    WRONE   ; WRITE STOP BIT
                                WASTE   43[
00D1'   3E2B            +       MVI     A,43
00D3'   3D              +..0006: DCR    A
00D4'   20FD            +       JRNZ    ..0006]
```

```
00D6'   C9                      RET
                        ; SUBROUTINE TO WRITE OUT 3 SECONDS WORTH OF LEA
                        DER
00D7'                   LEADER:
00D7'   01 0E10                 LXI     B,3600
00DA'                   ..LDR1:
                                WASTE   43[
00DA'   3E2B            +       MVI     A,43
00DC'   3D              +..0007: DCR    A
00DD'   20FD            +       JRNZ    ..0007]
00DF'   CD 00ED'                CALL    WRONE
00E2'   0B                      DCX     B
00E3'   78                      MOV     A,B
00E4'   B1                      ORA     C
00E5'   20F3                    JRNZ    ..LDR1
                                WASTE   42[
00E7'   3E2A            +       MVI     A,42
00E9'   3D              +..0008: DCR    A
00EA'   20FD            +       JRNZ    ..0008]
00EC'   C9                      RET
                        ; SUBROUTINE TO WRITE 1 HALF CYCLE OF A ONE BIT
                        ; 1/1200 SEC
00ED'                   WRONE:
00ED'   D399                    OUT     BANGER
                                WASTE   46[
00EF'   3E2E            +       MVI     A,46
00F1'   3D              +..0009: DCR    A
00F2'   20FD            +       JRNZ    ..0009]
00F4'   D399                    OUT     BANGER
00F6'   C9                      RET
                        ; SUBROUTINE TO WRITE 1 HALF CYCLE OF A ZERO BIT

                        ; 1/2400 SEC
00F7'                   WRZERO:
00F7'   D399                    OUT     BANGER
                                WASTE   23[
00F9'   3E17            +       MVI     A,23
00FB'   3D              +..0010: DCR    A
00FC'   20FD            +       JRNZ    ..0010]
00FE'   D399                    OUT     BANGER
0100'   C9                      RET
                                .END
```

BITBAN -
+++++ SYMBOL TABLE +++++

BADSAV	009E'		BANGER	0099		BOTRAM	8000	
BOTROM	2000		BOTSCR	BC20		CLRSCR	0000:07	X
COMMA	002C		CR	000D		DFTLMT	BC00	
EDKEY	0066		GETC	0000'	I	INBIT	0049'	
INCHAR	0035'		INITO	0000:08	X	KEYBSZ	0028	
LEADER	00D7'		LOADC	0007'	I	NLLN	0067	
OUTBYT	00AC'		PUTC	0077'	I	QHOW	0000:09	X
QWHAT	0000:0A	X	RSTEXP	0002		RSTFIN	0006	
RSTIGN	0004		RSTOCH	0003		RSTPAR	0005	
RUBOUT	001F		SAVEC	006A'	I	SAVEE	008C'	
STPBCT	00FC		TAPEIO	0012		TAPGET	0013'	
TEXT	0000:05	X	TOPSCR	BFFF		TSTVFF	0000:04	X
TXTUNF	0000:06	X	WRBLOC	00A1'		WRONE	00ED'	
WRZERO	00F7'		.BLNK.	0000:03	X	.DATA.	0000*	X
.PROG.	0101'	X						

```
0200                            .LOC    200H    ; TEST SHIT
                                .IDENT  BANGER
                                .PABS
                                .PHEX
1E00                    BANGIN = 1E00H  ; BIT BANGER READ PORT
1E01                    BANG1 = 1E01H   ; BIT BANG CODE TO WRITE A ONE
1E00                    BANG0 = 1E00H   ; BIT BANG CODE TO WRITE A ZERO
                        .DEFINE WASTE[TIME,%LAB]=[
                                MVI A,TIME
                        %LAB:   DCR A
                                JRNZ %LAB]


                        ;INCHAR CLOBBERS A, BC, DE
0200                    INCHAR:         ;NZ IF NO TIMEOUT, Z IF TIMEOUT,
                         CHAR IN C
0200    01 0810                 LXI B,810H      ;8 BITS, 10=TIMEOUT FACT
                        OR
0203    CD 0219         ..SBW:  CALL INBIT      ;WAIT FER START BIT
0206    3004                    JRNC ..GETL     ;GOTIT
0208    0D                      DCR C           ;TIMEOUT?
0209    20F8                    JRNZ ..SBW      ;NOT YET
020B    C9                      RET             ;Z SET
020C    CD 0219         ..GETL: CALL INBIT
020F    DC 0219                 CC INBIT        ;GET ANOTHER 1
0212    CB19                    RARR C
0214    10F6                    DJNZ ..GETL     ;GET 8 BITS
0216    AF                      XRA A
0217    3C                      INR A
0218    C9                      RET             ;RETURN
0219    3A 1E00         INBIT:  LDA BANGIN      ;GET BIT/EAT STOP BIT FR
                        OM ABOVE
021C    5F                      MOV E,A         ;WAIT FOR TRANSITION
021D    3A 1E00         ..INBW: LDA BANGIN      ;
0220    AB                      XRA E           ;CHANGE?
0221    0F                      RRC
0222    30F9                    JRNC ..INBW     ;NO
                                WASTE 30        ;WAIT FOR SAMPLE POINT[
0224    3E1E            +       MVI A,30
0226    3D              +..0001: DCR A
0227    20FD            +       JRNZ ..0001]
0229    3A 1E00                 LDA BANGIN
022C    AB                      XRA E           ;COMPARE TO OLDER
022D    E601                    ANI 1
022F    0F                      RRC
0230    C9                      RET

                        ;OUTBYT CLOBBERS A, BC
0231    4F              OUTBYA: MOV C,A         ;USE CHAR IN A
0232    CD 027B         OUTBYT: CALL WRZERO     ;WRITE START BIT
                                WASTE 19        ;VERY TIME SENSITIVE[
0235    3E13            +       MVI A,19
0237    3D              +..0002: DCR A
0238    20FD            +       JRNZ ..0002]
```

```
023A    0608            MVI B,8         ;WRITE 8 BITS
023C    CB09    ..WRL:  RRCR C          ;NEXT BIT TO CARRY
023E    380A            JRC ..WR1       ;IF ONE, WRITE ONE
0240    CD 027B         CALL WRZERO     ;ELSE WRITE ZERO
                        WASTE 19[
0243    3E13    +       MVI A,19
0245    3D      +..0003: DCR A
0246    20FD    +       JRNZ ..0003]
0248    1808            JMPR ..WRE      ;LOOP COUNTER
024A    CD 0273 ..WR1:  CALL WRONE      ;WRITE ONE-BIT
                        WASTE 42[
024D    3E2A    +       MVI A,42
024F    3D      +..0004: DCR A
0250    20FD    +       JRNZ ..0004]
0252    10E8    ..WRE:  DJNZ ..WRL      ;TILL 8 BITS DONE
0254    CD 0273         CALL WRONE      ;WRITE STOP BIT
                        WASTE 43[
0257    3E2B    +       MVI A,43
0259    3D      +..0005: DCR A
025A    20FD    +       JRNZ ..0005]
025C    C9              RET             ;ALL DONE!


                ;LEADER CLOBBERS BC AND A
025D    01 0E10 LEADER: LXI B,3600      ;3 SECS OF LEADER
0260            ..LDR1: WASTE 43[
0260    3E2B    +       MVI A,43
0262    3D      +..0006: DCR A
0263    20FD    +       JRNZ ..0006]
0265    CD 0273         CALL WRONE      ;LEADER IS ALL ONES
0268    0B              DCX B
0269    78              MOV A,B
026A    B1              ORA C
026B    20F3            JRNZ ..LDR1
                        WASTE 42[
026D    3E2A    +       MVI A,42
026F    3D      +..0007: DCR A
0270    20FD    +       JRNZ ..0007]
0272    C9              RET

                ;WRONE WRITES ONE HALF CYCLE OF ONE-BIT (1/1200
                SEC)

0273    3A 1E01 WRONE:  LDA BANG1       ;CHANGE ITS STATE
                        WASTE   22      ; WAIT SOME, THEN FALL I
                NTO ...[
0276    3E16    +       MVI A,22
0278    3D      +..0008: DCR A
0279    20FD    +       JRNZ ..0008]
                ;       WASTE 46
                ;       LDA BANG0       ;CHANGE STATE AGAIN
                ;       RET

                ;WRZERO WRITES ONE HLAF CYCLE OF ZERO BIT (1/240
                0 SEC)
```

```
027B    3A 1E01       WRZERO: LDA BANG1
                              WASTE 23[
027E    3E17          +       MVI A,23
0280    3D            +..0009: DCR A
0281    20FD          +       JRNZ ..0009]
0283    3A 1E00                LDA BANG0
0286    C9                     RET
                               .END
```

BANGER -
+++++ SYMBOL TABLE +++++

```
BANG0   1E00          BANG1   1E01          BANGIN 1E00
INBIT   0219          INCHAR  0200          LEADER 025D
OUTBYA  0231          OUTBYT  0232          WRONE  0273
WRZERO  027B          .BLNK.  0000#03 X     .DATA. 0000*    X
.PROG.  0000'   X
```