

NAM SWTBUG
 * VERSION 1.00

OPT PAG

 *REPLACEMENT FOR MIKBUG ROM
 *FOR SWTPC 6800 COMPUTER SYSTEM
 *COPYRIGHT 1977
 *SOUTHWEST TECHNICAL PROD. CORP.
 *AUGUST, 1977

A000		ORG	\$A000	
A000	IRQ	RMB	2	IRQ POINTER
A002	BEGA	RMB	2	BEGINNING ADDR PNCH
A004	ENDA	RMB	2	ENDING ADDR PNCH
A006	NMI	RMB	2	NMI INTERRUPT VECTOR
A008	SP	RMB	1	S HIGH
A009		RMB	1	S LOW
A00A	PORADD	RMB	2	PORT ADDRESS
A00C	PORECH	RMB	1	ECHO ON/OFF FLAG
A00D	XHI	RMB	1	XREG HIGH
A00E	XLOW	RMB	1	XREG LOW
A00F	CKSM	RMB	1	CHECKSUM
A010	XTEMP	RMB	2	X-REG TEMP STGE
A012	SWIJMP	RMB	2	SWI JUMP VECTOR
A044	TW	EQU	\$A044	TEMPORARY STORAGE
A046	TEMP	EQU	\$A046	TEMPORARY STORAGE
A047	BYTECT	EQU	\$A047	BYTECT AND MCONT TEMP.
8004	CTLPOR	EQU	\$8004	CONTROL PORT ADDRESS
C000	PROM	EQU	\$C000	JUMP TO PROM ADDRESS
A014	BKPT	RMB	2	BREAKPOINT ADDRESS
A016	BKLST	RMB	1	BREAKPOINT DATA
A042		ORG	\$A042	
A042	STACK	RMB	1	SWTBUG STACK
E000		ORG	\$E000	
*I/O INTERRUPT SEQUENCE				
E000	FE A0 00	IRQV	LDX	IRQ
E003	6E 00		JMP	0,X
*JUMP TO USER PROGRAM				
E005	8D 40	JUMP	BSR	BADDR
E007	6E 00		JMP	0,X
E009	10	CURSOR	FCB	\$10,\$16,4 CT-1024 CURSOR CONTROL
E00A	16 04			
*ASCII LOADING ROUTINE				
E00C	BD E3 34	LOAD	JSR	RDON READER ON, DIS ECHO, GET P#
E00F	8D 67	LOAD3	BSR	INCH
E011	81 53		CMP A	#'S

```

E013 26 FA          BNE    LOAD3    1ST CHAR NOT S
E015 8D 61          BSR    INCH      READ CHAR
E017 81 39          CMP A  #'9
E019 27 29          BEQ    LOAD21
E01B 81 31          CMP A  #'1
E01D 26 F0          BNE    LOAD3    2ND CHAR NOT 1
E01F 7F A0 0F      CLR    CKSM      ZERO CHECKSUM
E022 8D 31          BSR    BYTE      READ BYTE
E024 80 02          SUB A  #2
E026 B7 A0 47      STA A  BYTECT    BYTE COUNT

      *BUILD ADDRESS
E029 8D 1C          BSR    BADDR
      *STORE DATA
E02B 8D 28          LOAD11 BSR    BYTE
E02D 7A A0 47      DEC    BYTECT
E030 27 09          BEQ    LOAD15    ZERO BYTE COUNT
E032 A7 00          STA A  0,X      STORE DATA
E034 A1 00          CMP A  0,X      DATA STORED?
E036 26 08          BNE    LOAD19
E038 08            INX
E039 20 F0          BRA    LOAD11
E03B 7C A0 0F      LOAD15 INC    CKSM
E03E 27 CF          BEQ    LOAD3
E040 86 3F          LOAD19 LDA A  #'?
E042 8D 31          BSR    OUTCH
E044 7E E2 D4      LOAD21 JMP    RDOFF1

      *BUILD ADDRESS
E047 8D 0C          BADDR BSR    BYTE    READ 2 FRAMES
E049 B7 A0 0D      STA A  XHI
E04C 8D 07          BSR    BYTE
E04E B7 A0 0E      STA A  XLOW
E051 FE A0 0D      LDX    XHI      LOAD IXR WITH NUMBER
E054 39            RTS

      *INPUT BYTE (TWO FRAMES)
E055 8D 53          BYTE BSR    INHEX    GET HEX CHAR
E057 48            BYTE1 ASL A
E058 48            ASL A
E059 48            ASL A
E05A 48            ASL A
E05B 16            TAB
E05C 8D 4C          BSR    INHEX
E05E 1B            ABA
E05F 16            TAB
E060 FB A0 0F      ADD B  CKSM
E063 F7 A0 0F      STA B  CKSM
E066 39            RTS

E067 44            OUTHL LSR A    OUT HEX LEFT BCD DIGIT
E068 44            LSR A
E069 44            LSR A
E06A 44            LSR A
E06B 84 0F          OUTHR AND A  #$F    OUT HEX RIGHT BCD DIGIT

```

```

E06D 8B 30          ADD A  #$30
E06F 81 39          CMP A  #$39
E071 23 02          BLS   OUTCH
E073 8B 07          ADD A  #$7

                *OUTPUT ONE CHAR
E075 7E E1 D1      OUTCH  JMP    OUTEEE
E078 7E E1 AC      INCH   JMP    INEEE

                *PRINT DATA POINTED TO BY X REG
E07B 8D F8          PDATA2 BSR   OUTCH
E07D 08              INX
E07E A6 00          PDATA1 LDA A  0,X
E080 81 04          CMP A  #4
E082 26 F7          BNE   PDATA2
E084 39              RTS                STOP ON HEX 04

E085 7E E1 4A      C1     JMP    SWTCTL

                *MEMORY EXAMINE AND CHANGE
E088 8D BD          CHANGE BSR   BADDR
E08A CE E1 9D      CHA51  LDX   #MCL
E08D 8D EF          BSR   PDATA1   C/R L/F
E08F CE A0 0D          LDX   #XHI
E092 8D 34          BSR   OUT4HS   PRINT ADDRESS
E094 FE A0 0D          LDX   XHI
E097 8D 31          BSR   OUT2HS   PRINT OLD DATA
E099 8D 31          BSR   OUTS    OUTPUT SPACE
E09B 8D DB          ANOTH  BSR   INCH   INPUT CHAR
E09D 81 20          CMP A  #$20
E09F 27 FA          BEQ   ANOTH
E0A1 81 0D          CMP A  #$D
E0A3 27 E0          BEQ   C1
E0A5 81 5E          CMP A  #'^   UP ARROW?
E0A7 20 2C          BRA   AL3    BRANCH FOR ADJUSTMENT
E0A9 01              NOP

                *INPUT HEX CHARACTER
E0AA 8D CC          INHEX  BSR   INCH
E0AC 80 30          INHEX1 SUB A  #$30
E0AE 2B 4C          BMI   C3
E0B0 81 09          CMP A  #$9
E0B2 2F 0A          BLE   IN1HG
E0B4 81 11          CMP A  #$11
E0B6 2B 44          BMI   C3          NOT HEX
E0B8 81 16          CMP A  #$16
E0BA 2E 40          BGT   C3          NOT HEX
E0BC 80 07          SUB A  #7
E0BE 39              IN1HG  RTS

E0BF A6 00          OUT2H  LDA A  0,X   OUTPUT 2 HEX CHAR
E0C1 8D A4          OUT2HA BSR   OUTHL  OUT LEFT HEX CHAR
E0C3 A6 00          LDA A  0,X
E0C5 08              INX

```

```

E0C6 20 A3          BRA    OUTHR    OUTPUT RIGHT HEX CHAR

E0C8 8D F5          OUT4HS BSR    OUT2H    OUTPUT 4 HEX CHAR + SPACE
E0CA 8D F3          OUT2HS BSR    OUT2H    OUTPUT 2 HEX CHAR + SPACE

E0CC 86 20          OUTS   LDA A  #$20    SPACE
E0CE 20 A5          BRA    OUTCH    (BSR & TRS)

*ENTER POWER ON SEQUENCE
E0D0 8E A0 42      START  LDS    #STACK
E0D3 20 2C          BRA    AL1      BRANCH FOR ADDRESS COMPATIBIL

*****
*PART OF MEMORY EXAMINE AND CHANGE
E0D5 26 07          AL3    BNE    SK1
E0D7 09             DEX
E0D8 09             DEX
E0D9 FF A0 0D          STX    XHI
E0DC 20 AC          BRA    CHA51
E0DE FF A0 0D      SK1    STX    XHI
E0E1 20 02          BRA    AL4

E0E3 20 6D          EOE3   BRA    CONTRL    BRANCH FOR MIKBUG EQUIV. CONT

E0E5 81 30          AL4    CMP A  #$30
E0E7 25 A1          BCS    CHA51
E0E9 81 46          CMP A  #$46
E0EB 22 9D          BHI    CHA51
E0ED 8D BD          BSR    INHEX1
E0EF BD E0 57          JSR    BYTE1
E0F2 09             DEX
E0F3 A7 00          STA A  0,X      CHANGE MEMORY
E0F5 A1 00          CMP A  0,X
E0F7 27 91          BEQ    CHA51    DID CHANGE
E0F9 7E E0 40          JMP    LOAD19   DIDN'T CHANGE
E0FC BE A0 08      C3    LDS    SP
E0FF 20 49          BRA    SWTCTL

*****
*CONTINUE POWER UP SEQUENCE
E101 BF A0 08      AL1    STS    SP      INIT TARGET STACK PTR.
E104 86 FF          LDA A  #$FF
E106 BD E3 08          JSR    SWISET

*CONFIGURE FOR PIA AND SEE IF OK
E109 CE 80 04          LDX    #CTLPOR
E10C BD E2 84          JSR    PIAINI    INIT PIA
E10F A6 00          LDA A  0,X
E111 A1 02          CMP A  2,X
E113 20 02          BRA    AL2

E115 20 19          BRA    PRINT    BRA FOR BILOAD

E117 26 39          AL2    BNE    CONTRL

```

```

*INITIALIZE AS ACIA
E119 86 03          LDA A  #3          ACIA MASTER RESET
E11B A7 00          STA A  0,X
E11D 86 11          LDA A  #$11
E11F A7 00          STA A  0,X
E121 20 2F          BRA    CONTRL

*ENTER FROM SOFTWARE INTERRUPT
E123 01          SF0    NOP
E124 BF A0 08     SFE1   STS    SP          SAVE TARGETS STACK POINTER
*DECREMENT P COUNTER
E127 30          TSX
E128 6D 06          TST    6,X
E12A 26 02          BNE    *+4
E12C 6A 05          DEC    5,X
E12E 6A 06          DEC    6,X
*PRINT CONTENTS OF STACK.
E130 CE E1 9D     PRINT  LDX    #MCL
E133 BD E0 7E          JSR    PDATA1
E136 FE A0 08          LDX    SP
E139 08          INX
E13A 8D 8E          BSR    OUT2HS    COND CODES
E13C 8D 8C          BSR    OUT2HS    ACC B
E13E 8D 8A          BSR    OUT2HS    ACC A
E140 8D 86          BSR    OUT4HS    IXR
E142 8D 84          BSR    OUT4HS    PGM COUNTER
E144 CE A0 08          LDX    #SP
E147 BD E0 C8          JSR    OUT4HS    STACK POINTER
E14A FE A0 12     SWTCTL LDX    SWIJMP
E14D 8C E1 23          CPX    #SF0
E150 27 19          BEQ    CONTR1

E152 8E A0 42     CONTRL LDS    #STACK    SET CONTRL STACK POINTER
E155 CE 80 04          LDX    #CTLPOR   RESET TO CONTROL PORT
E158 FF A0 0A          STX    PORADD
E15B 7F A0 0C          CLR    PORECH    TURN ECHO ON
E15E 8D 73          BSR    SAVGET    GET PORT # AND TYPE
E160 27 03          BEQ    POF1
E162 BD E2 7D          JSR    PIAECH    SET PIA ECHO ON IF MP-C INTER
E165 BD E3 53     POF1   JSR    PNCHOF    TURN PUNCH OFF
E168 BD E3 47          JSR    RDOFF    TURN READER OFF
E16B CE E1 9C     CONTR1 LDX    #MCLOFF
E16E BD E0 7E          JSR    PDATA1    PRINT DATA STRING
E171 8D 39          BSR    INEEE    READ COMMAND CHARACTER

*COMMAND LOOKUP ROUTINE
E173 CE E3 D1     LOOK   LDX    #TABLE
E176 A1 00          OVER   CMP A  0,X
E178 26 07          BNE    SK3
E17A BD E0 CC          JSR    OUTS    SKIP SPACE
E17D EE 01          LDX    1,X
E17F 6E 00          JMP    0,X
E181 08          SK3    INX
E182 08          INX

```

```

E183 08                INX
E184 8C E3 F8         CPX    #TABEND+3
E187 26 ED           BNE    OVER
E189 20 BF          SWTL1  BRA    SWTCTL

                *SOFTWARE INTERRUPT ENTRY POINT
E18B FE A0 12       SFE    LDX    SWIJMP    JUMP TO VECTORED SOFTWARE INT
E18E 6E 00           JMP    0,X

E190 53            S9     FCB    'S','9',4    END OF TAPE
E191 39 04

                *****
E193 0D           MTAPE1  FCB    $D,$A,$15,0,0,0,'S','1',4 PUNCH FORMAT
E194 0A 15
E196 00 00
E198 00 53
E19A 31 04

E19C 13           MCLOFF  FCB    $13          READER OFF
E19D 0D           MCL     FCB    $D,$A,$15,0,0,0,'$',4
E19E 0A 15
E1A0 00 00
E1A2 00 24
E1A4 04

E1A5 20 4C       EIA5    BRA    BILD          BINARY LOADER INPUT
                *****

                *NMI SEQUENCE
E1A7 FE A0 06       NMIV   LDX    NMI          GET NMI VECTOR
E1AA 6E 00           JMP    0,X

E1AC 20 40       INEEE   BRA    INEEE1

                *BYTE SEARCH ROUTINE
E1AE BD E0 47       SEARCH JSR    BADDR          GET TOP ADDRESS
E1B1 FF A0 04           STX    ENDA
E1B4 BD E0 47           JSR    BADDR          GET BOTTOM ADDRESS
E1B7 BD E0 55           JSR    BYTE          GET BYTE TO SEARCH FOR
E1BA 16                TAB
E1BB A6 00           OVE   LDA    A    0,X
E1BD FF A0 0D           STX    XHI
E1C0 11                CBA
E1C1 27 02           BEQ    PNT
E1C3 20 21           BRA    INCR1
E1C5 CE E1 9D       PNT   LDX    #MCL
E1C8 BD E0 7E           JSR    PDATA1
E1CB CE A0 0D           LDX    #XHI
E1CE 20 10           BRA    SKP0
                *****

                *GO TO USER PROGRAM ROUTINE

```

```

E1D0 3B      GOTO    RTI
E1D1 20 3A   OUTEEE  BRA    OUTEE1

```

*SAVE IXR AND LOAD IXR WITH CORRECT

*PORT NUMBER AND TEST FOR TYPE

```

E1D3 FF A0 10 SAVGET  STX    XTEMP    STORE INDEX REGISTER
E1D6 FE A0 0A GETPT1  LDX    PORADD
E1D9 37      ISACIA  PSH  B
E1DA E6 01      LDA  B  1,X
E1DC E1 03      CMP  B  3,X
E1DE 33      PUL  B
E1DF 39      RTS

```

*CONTINUATION OF SEARCH ROUTINE

```

E1E0 BD E0 C8 SKP0    JSR    OUT4HS
E1E3 FE A0 0D      LDX    XHI
E1E6 BC A0 04 INCR1  CPX    ENDA
E1E9 27 9E      BEQ    SWTL1
E1EB 08      INX
E1EC 20 CD      BRA    OVE

E1EE 8D 06 INEEEE1 BSR    INCH8    INPUT 8 BIT CHARACTER
E1F0 84 7F      AND  A  %#01111111 GET RID OF PARITY BIT
E1F2 39      RTS

E1F3 31      BILD   INS          FIX UP STACK WHEN USING
E1F4 31      INS          BINARY LOADER ON SWTPC TAPES
E1F5 31      INS

```

*INPUT ONE CHAR INTO ACC B

```

E1F6 37      INCH8  PSH  B      SAVE ACC B
E1F7 8D DA      BSR    SAVGET    SAVE IXR, GET PORT# AND TYPE
E1F9 26 28      BNE    IN1       INPUT FROM PIA IF NOT
E1FB 86 15      LDA  A  #$15     RECONFIG FOR 8 BIT, 1 SB
E1FD A7 00      STA  A  0,X
E1FF A6 00 ACIAIN LDA  A  0,X
E201 47      ASR  A
E202 24 FB      BCC   ACIAIN    NOT READY
E204 A6 01      LDA  A  1,X     LOAD CHAR
E206 F6 A0 0C      LDA  B  PORECH
E209 27 07      BEQ   ACIOUT    ECHO
E20B 20 11      BRA   RES      DON'T ECHO

```

*OUTPUT ONE CHARACTER

```

E20D 37      OUTEE1  PSH  B      SAVE ACC B
E20E 8D C3      BSR    SAVGET
E210 26 2E      BNE    IOUT

E212 C6 11      ACIOUT LDA  B  #$11
E214 E7 00      STA  B  0,X
E216 E6 00      ACIOU1 LDA  B  0,X

```

```

E218 57          ASR B
E219 57          ASR B
E21A 24 FA      BCC ACIOU1  ACIA NOT READY
E21C A7 01      STA A 1,X    OUTPUT CHARACTER
E21E 33          RES      PUL B  RESTORE ACC B
E21F FE A0 10   LDX      XTEMP
E222 39          RTS

          *PIA INPUT ROUTINE
E223 A6 00      IN1      LDA A 0,X    LOOK FOR START BIT
E225 2B FC          BMI      IN1
E227 8D 3A          BSR      DDL      DELAY HALF BIT TIME
E229 C6 04          LDA B #4        SET DEL FOR FULL BIT TIME
E22B E7 02          STA B 2,X
E22D 58          ASL B      SET UP CNTR WITH 8
E22E 8D 2A      IN3      BSR      DEL  WAIT ONE CHAR TIME
E230 0D          SEC
E231 69 00      ROL      0,X
E233 46          ROR A
E234 5A          DEC B
E235 26 F7          BNE      IN3
E237 8D 21          BSR      DEL  WAIT FOR STOPBIT
E239 F6 A0 0C      LDA B PORECH  IS ECHO DESIRED?
E23C 27 13          BEQ      IOUT2   ECHO
E23E 20 DE          BRA      RES     RESTORE IXR,ACCB

          *PIA OUTPUT ROUTINE
E240 8D 23      IOUT     BSR      DDL1  DELAY ONE HALF BIT TIME
E242 C6 0A          LDA B #$A      SET UP COUNTER
E244 6A 00          DEC      0,X    SET START BIT
E246 8D 16          BSR      DE     START TIMER
E248 8D 10      OUT1     BSR      DEL  DELAY ONE BIT TIME
E24A A7 00          STA A 0,X    PUT OUT ONE DATA BIT
E24C 0D          SEC
E24D 46          ROR A      SHIFT IN NEXT BIT
E24E 5A          DEC B      DECREMENT COUNTER
E24F 26 F7          BNE      OUT1   TEST FOR 0
E251 E6 02      IOUT2    LDA B 2,X  TEST FOR STOP BITS
E253 58          ASL B      SHIFT BIT TO SIGN
E254 2A C8          BPL      RES     BRA FOR 1 STOP BIT
E256 8D 02          BSR      DEL  DELAY FOR STOP BITS
E258 20 C4          BRA      RES
E25A 6D 02      DEL      TST      2,X  IS TIME UP
E25C 2A FC          BPL      DEL
E25E 6C 02      DE       INC      2,X  RESET TIMER
E260 6A 02          DEC      2,X
E262 39          RTS

E263 6F 02      DDL      CLR      2,X  HALF BIT DELAY
E265 8D F7      DDL1     BSR      DE
E267 20 F1          BRA      DEL

          *OPTIONAL PORT ROUTINE
E269 8D 83      OPTL     BSR      INEEE1

```

```

E26B 16          TAB
E26C 7F A0 0B   CLR    PORADD+1  SET I/O ADDRESS FOR $8000
E26F FE A0 0A   LDX    PORADD
E272 8D 10      BSR    PIAINI    INITIALIZE PIA
E274 8D 07      BSR    PIAECH    SET ECHO
E276 CE E3 EF   LDX    #TABLE1  P, L OR E
E279 17         TBA
E27A 7E E1 76   JMP    OVER      LOOK AT TABLE FOR E, L OR P

```

```

E27D 86 34      PIAECH LDA A  #$34    SET DDR
E27F A7 03      STA A  3,X
E281 A7 02      STA A  2,X
E283 39         NOOPT  RTS

```

*PIA INITIALIZATION ROUTINE

```

E284 6C 00      PIAINI INC    0,X      SET DDR
E286 86 07      LDA A  #$7
E288 A7 01      STA A  1,X
E28A 6C 00      INC    0,X
E28C A7 02      STA A  2,X
E28E 39         RTS

```

*MINIFLOPPY DISK BOOT

```

E28F 7F 80 14   DISK   CLR    $8014
E292 8D 2E      BSR    DELAY
E294 C6 0B      LDA B  #$0B
E296 8D 25      BSR    RETT2
E298 E6 04      LOOP1  LDA B  4,X
E29A C5 01      BIT B  #1
E29C 26 FA      BNE    LOOP1
E29E 6F 06      CLR    6,X
E2A0 8D 1D      BSR    RETURN
E2A2 C6 9C      LDA B  #$9C
E2A4 8D 17      BSR    RETT2
E2A6 CE 24 00   LDY    #$2400
E2A9 C5 02      LOOP2  BIT B  #2
E2AB 27 06      BEQ    LOOP3
E2AD B6 80 1B   LDA A  $801B
E2B0 A7 00      STA A  0,X
E2B2 08         INX
E2B3 F6 80 18   LOOP3  LDA B  $8018
E2B6 C5 01      BIT B  #1
E2B8 26 EF      BNE    LOOP2
E2BA 7E 24 00   JMP    $2400
E2BD E7 04      RETT2  STA B  4,X
E2BF 8D 00      RETURN BSR    RETT1
E2C1 39         RETT1  RTS

```

*GENERAL PURPOSE DELAY LOOP

```

E2C2 CE FF FF   DELAY  LDX    #$FFFF
E2C5 09         DELAY1  DEX
E2C6 8C 80 14   CPX    #$8014  STOP AT 8014
E2C9 26 FA      DUM    BNE    DELAY1
E2CB 39         RTS

```

```

*CLRAR SCREEN FOR CT-1024 TYPE TERMINALS
E2CC CE E0 09 CLEAR LDX #CURSOR
E2CF BD E0 7E JSR PDATA1
E2D2 8D F1 BSR DELAY1 DELAY
E2D4 BD E3 47 RDOFF1 JSR RDOFF
E2D7 20 58 BRA C4

*BREAKPOINT ENTERING ROUTINE
E2D9 CE E1 23 BREAK LDX #SF0
E2DC BC A0 12 CPX SWIJMP BREAKPOINTS ALREADY IN USE?
E2DF 27 1A BEQ INUSE
E2E1 08 INX
E2E2 8D 32 BREAK0 BSR STO1
E2E4 BD E0 47 JSR BADDR
E2E7 FF A0 14 STX BKPT
E2EA A6 00 LDA A 0,X
E2EC B7 A0 16 STA A BKLST
E2EF 86 3F LDA A #$3F
E2F1 A7 00 STA A 0,X
E2F3 CE E1 23 LDX #SF0
E2F6 8D 1E BSR STO1
E2F8 7E E1 6B JMP CONTR1
E2FB FE A0 14 INUSE LDX BKPT
E2FE B6 A0 16 LDA A BKLST
E301 A7 00 STA A 0,X
E303 CE E1 24 LDX #SFE1
E306 20 DA BRA BREAK0

E308 B7 A0 43 SWISET STA A STACK+1 FIX POWER UP INTERRUPT
E30B FE A0 12 LDX SWIJMP
E30E 8C E1 23 CPX #SF0
E311 27 06 BEQ STORTN
E313 CE E1 24 STO LDX #SFE1
E316 FF A0 12 STO1 STX SWIJMP
E319 39 STORTN RTS

E31A 8D 5A PUNCH1 BSR PUNCH
E31C 20 0F BRA POF4

*FORMAT END OF TAPE WITH PGM. CTR. AND S9
E31E CE A0 49 PNCHS9 LDX #$A049
E321 FF A0 04 STX ENDA
E324 09 DEX
E325 8D 52 BSR PUNCH2
E327 CE E1 90 LDX #S9
E32A BD E0 7E PDAT JSR PDATA1
E32D 8D 24 POF4 BSR PNCHOF
E32F 8D 91 BSR DELAY
E331 7E E1 52 C4 JMP CONTRL

E334 73 A0 0C RDON COM PORECH DISABLE ECHO FOR ACIA
E337 86 11 LDA A #$11 RON CHAR.

```

```

E339 C6 20          LDA B  #$20      STROBE CHAR
E33B 8D 1A          BSR    STROBE
E33D BD E1 D9      JSR    ISACIA    CHECK TO SEE IF PIA
E340 27 04          BEQ    RTNN
E342 86 3C          LDA A  #$3C      DISABLE PIA ECHO IF PIA
E344 A7 03          STA A  3,X
E346 39            RTNN    RTS

E347 86 13          RDOFF  LDA A  #$13      TURN READER OFF
E349 C6 10          LDA B  #$10
E34B 20 0A          BRA    STROBE

E34D 86 12          PNCHON LDA A  #$12
E34F C6 04          LDA B  #4
E351 20 04          BRA    STROBE

E353 86 14          PNCHOF LDA A  #$14
E355 C6 08          LDA B  #$8

```

*PIA STROBING ROUTINE FOR PUNCH/READ ON/OFF

```

E357 BD E0 75      STROBE JSR    OUTCH
E35A BD E1 D6      JSR    GETPT1
E35D 27 16          BEQ    RTN1
E35F 86 02          LDA A  #2
E361 CA 01          ORA B  #1
E363 8D 0C          BSR    STR2
E365 8D 08          BSR    STR1
E367 86 02          LDA A  #2
E369 C6 01          LDA B  #1
E36B E7 00          STA B  0,X
E36D 8D 02          BSR    STR2
E36F 86 06          STR1  LDA A  #6
E371 A7 01          STR2  STA A  1,X
E373 E7 00          STA B  0,X
E375 39            RTN1  RTS

```

*PUNCH FROM BEGINNING ADDRESS (BEGA) THRU
*ENDING ADDRESS (ENDA)

```

E376 FE A0 02      PUNCH  LDX    BEGA
E379 FF A0 44      PUNCH2 STX    TW
E37C 8D CF          BSR    PNCHON
E37E B6 A0 05      PUN11  LDA A  ENDA+1
E381 B0 A0 45          SUB A  TW+1
E384 F6 A0 04          LDA B  ENDA
E387 F2 A0 44          SBC B  TW
E38A 26 04          BNE    PUN22
E38C 81 10          CMP A  #16
E38E 25 02          BCS    PUN23
E390 86 0F          PUN22  LDA A  #15
E392 8B 04          PUN23  ADD A  #4
E394 B7 A0 47          STA A  BYTECT
E397 80 03          SUB A  #3
E399 B7 A0 46          STA A  TEMP

```

*PUNCH C/R L/F NULLS S1

```

E39C CE E1 93          LDX    #MTAPE1
E39F BD E0 7E          JSR    PDATA1
E3A2 5F                CLR    B
                    *PUNCH FRAME COUNT
E3A3 CE A0 47          LDX    #BYTECT
E3A6 8D 24             BSR    PUNT2    PUNCH 2 HEX CHARACTERS
                    *PUNCH ADDRESS
E3A8 CE A0 44          LDX    #TW
E3AB 8D 1F             BSR    PUNT2
E3AD 8D 1D             BSR    PUNT2
                    *PUNCH DATA
E3AF FE A0 44          LDX    TW
E3B2 8D 18             PUN32  BSR    PUNT2    PUNCH ONE BYTE
E3B4 7A A0 46          DEC    TEMP
E3B7 26 F9             BNE    PUN32
E3B9 FF A0 44          STX    TW
E3BC 53                COM    B
E3BD 37                PSH    B
E3BE 30                TSX
E3BF 8D 0B             BSR    PUNT2    PUNCH CHECKSUM
E3C1 33                PUL    B        RESTORE STACK
E3C2 FE A0 44          LDX    TW
E3C5 09                DEX
E3C6 BC A0 04          CPX    ENDA
E3C9 26 B3             BNE    PUN11
E3CB 39                RTN5   RTS

                    *PUNCH 2 HEX CHAR, UPDATE CHECKSUM
E3CC EB 00             PUNT2  ADD    B    0,X
E3CE 7E E0 BF          JMP    OUT2H    OUTPUT 2 HEX CHAR AND RTS

                    *COMMAND TABLE
E3D1 47                TABLE FCB    'G    GOTO
E3D2 E1 D0             FDB    GOTO
E3D4 5A                FCB    'Z    GOTO PROM
E3D5 C0 00             FDB    PROM
E3D7 4D                FCB    'M    MEMORY EXAM AND CHANGE
E3D8 E0 88             FDB    CHANGE
E3DA 46                FCB    'F    BYTE SEARCH
E3DB E1 AE             FDB    SEARCH
E3DD 52                FCB    'R    REGISTER DUMP
E3DE E1 30             FDB    PRINT
E3E0 4A                FCB    'J    JUMP
E3E1 E0 05             FDB    JUMP
E3E3 43                FCB    'C    CLEAR SCREEN
E3E4 E2 CC             FDB    CLEAR
E3E6 44                FCB    'D    DISK BOOT
E3E7 E2 8F             FDB    DISK
E3E9 42                FCB    'B    BREAKPOINT
E3EA E2 D9             FDB    BREAK
E3EC 4F                FCB    'O    OPTIONAL PORT
E3ED E2 69             FDB    OPTL
E3EF 50                TABLE1 FCB    'P    ASCII PUNCH
E3F0 E3 1A            FDB    PUNCH1

```

E3F2 4C		FCB	'L	ASCII LOAD
E3F3 E0 0C		FDB	LOAD	
E3F5 45	TABEND	FCB	'E	END OF TAPE
E3F6 E3 1E		FDB	PNCHS9	
E3F8		ORG	\$E3F8	
E3F8 E0 00		FDB	IRQV	IRQ VECTOR
E3FA E1 8B		FDB	SFE	SOFTWARE INTERRUPT
E3FC E1 A7		FDB	NMIV	NMI VECTOR
E3FE E0 D0		FDB	START	RESTART VECTOR
A048		ORG	\$A048	
A048 E0 D0		FDB	START	
		END		

ERROR(S) DETECTED

SYMBOL TABLE:

ACIAIN	E1FF	ACIOU1	E216	ACIOUT	E212	AL1	E101	AL2	E117
AL3	E0D5	AL4	E0E5	ANOTH	E09B	BADDR	E047	BEGA	A002
BILD	E1F3	BKLST	A016	BKPT	A014	BREAK	E2D9	BREAK0	E2E2
BYTE	E055	BYTE1	E057	BYTECT	A047	C1	E085	C3	E0FC
C4	E331	CHA51	E08A	CHANGE	E088	CKSM	A00F	CLEAR	E2CC
CONTR1	E16B	CONTRL	E152	CTLPOR	8004	CURSOR	E009	DDL	E263
DDL1	E265	DE	E25E	DEL	E25A	DELAY	E2C2	DELAY1	E2C5
DISK	E28F	DUM	E2C9	EIA5	E1A5	ENDA	A004	EOE3	E0E3
GETPT1	E1D6	GOTO	E1D0	IN1	E223	IN1HG	E0BE	IN3	E22E
INCH	E078	INCH8	E1F6	INCR1	E1E6	INEEE	E1AC	INEEE1	E1EE
INHEX	E0AA	INHEX1	E0AC	INUSE	E2FB	IOUT	E240	IOUT2	E251
IRQ	A000	IRQV	E000	ISACIA	E1D9	JUMP	E005	LOAD	E00C
LOAD11	E02B	LOAD15	E03B	LOAD19	E040	LOAD21	E044	LOAD3	E00F
LOOK	E173	LOOP1	E298	LOOP2	E2A9	LOOP3	E2B3	MCL	E19D
MCLOFF	E19C	MTAPE1	E193	NMI	A006	NMIV	E1A7	NOOPT	E283
OPTL	E269	OUT1	E248	OUT2H	E0BF	OUT2HA	E0C1	OUT2HS	E0CA
OUT4HS	E0C8	OUTCH	E075	OUTEE1	E20D	OUTEEE	E1D1	OUTHL	E067
OUTHR	E06B	OUTS	E0CC	OVE	E1BB	OVER	E176	PDAT	E32A
PDATA1	E07E	PDATA2	E07B	PIAECH	E27D	PIAINI	E284	PNCHOF	E353
PNCHON	E34D	PNCHS9	E31E	PNT	E1C5	POF1	E165	POFC4	E32D
PORADD	A00A	PORECH	A00C	PRINT	E130	PROM	C000	PUN11	E37E
PUN22	E390	PUN23	E392	PUN32	E3B2	PUNCH	E376	PUNCH1	E31A
PUNCH2	E379	PUNT2	E3CC	RDOFF	E347	RDOFF1	E2D4	RDON	E334
RES	E21E	RETT1	E2C1	RETT2	E2BD	RETURN	E2BF	RTN1	E375
RTN5	E3CB	RTNN	E346	S9	E190	SAVGET	E1D3	SEARCH	E1AE
SF0	E123	SFE	E18B	SFE1	E124	SK1	E0DE	SK3	E181
SKP0	E1E0	SP	A008	STACK	A042	START	E0D0	STO	E313
STO1	E316	STORTN	E319	STR1	E36F	STR2	E371	STROBE	E357
SWIJMP	A012	SWISET	E308	SWTCTL	E14A	SWTL1	E189	TABEND	E3F5
TABLE	E3D1	TABLE1	E3EF	TEMP	A046	TW	A044	XHI	A00D
XLOW	A00E	XTEMP	A010						