

CBUG05 DEBUG MONITOR PROGRAM FOR MC146805E2 MICROPROCESSOR UNIT

1. INTRODUCTION

CBUG05 is a debug monitor program written for the MC146805E2 Microprocessor Unit and contained in the MCM65516 2K × 8 CMOS ROM. CBUG05 allows for rapid development and evaluation of hardware and M6805 Family type software, using memory and register examine/change commands as well as breakpoint and single instruction trace commands. CBUG05 also includes software to set and display time, using an optional MC146818 Real-Time Clock (RTC), and routines to punch and load an optional cassette interface. Figure 1 shows a minimum system which only requires the MPU, ROM, keypad inputs and display output interfaces (a schematic diagram of this minimum system is shown in Figure 10). Port A of the MC146805E2 MPU is required for the I/O; however, Port B and all other MC146805E2 MPU features remain available to the user. A possible expanded system is shown in Figure 2. The memory map is shown in Figure 3. Locations \$1700-\$173F are available to the user if the optional MC146818 RTC is not used.

FEATURES:

- MC146805E2 Eight-Bit CMOS MPU
 - Expandable Multiplied Address/Data Bus
 - Eight-Bit I/O Port
 - Eight-Bit Timer with Prescaler
 - Maskable External Interrupt
 - 16 Levels of Subroutine Nesting
 - Minimum of 38 Bytes of Unused Internal RAM
- MCM65516 2K × 8 CMOS with CBUG05
 - Memory and Register Examine/Change
 - Breakpoints and Single Instruction Trace
 - Branch Offset Calculation
 - Set/Display Current Time (w/optional MC146818 Real-Time Clock)
 - Punch/Load/Verify Cassette Tape (w/optional cassette tape interface)
 - Stop Command for Low-power Software Standby
 - Software Alterable Interrupt Vectors

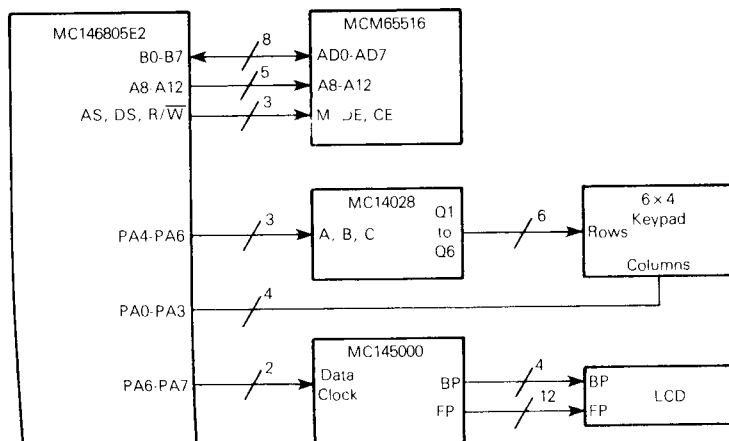


Figure 1. Minimum CBUG05 System

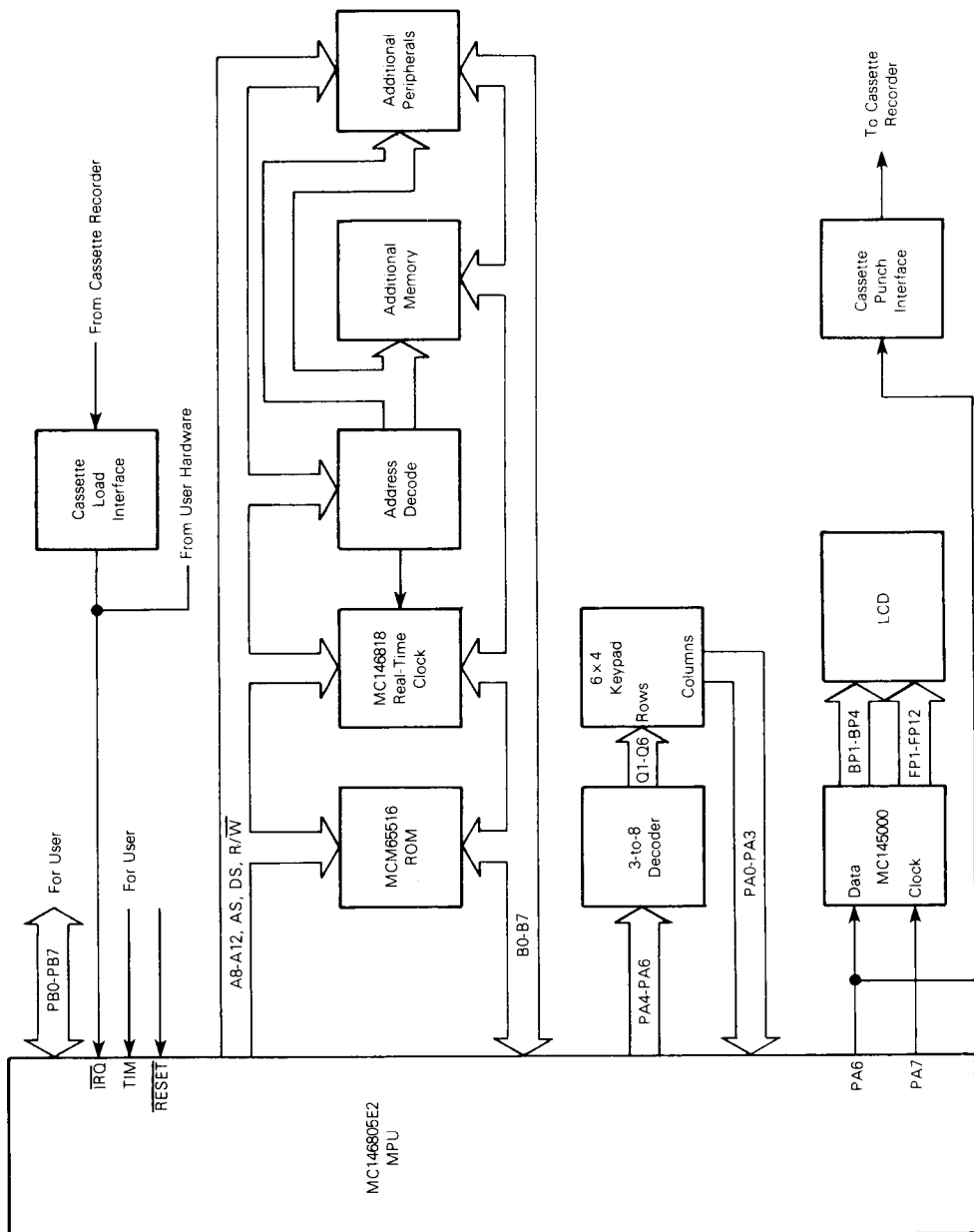


Figure 2. Expanded CBUG05 System

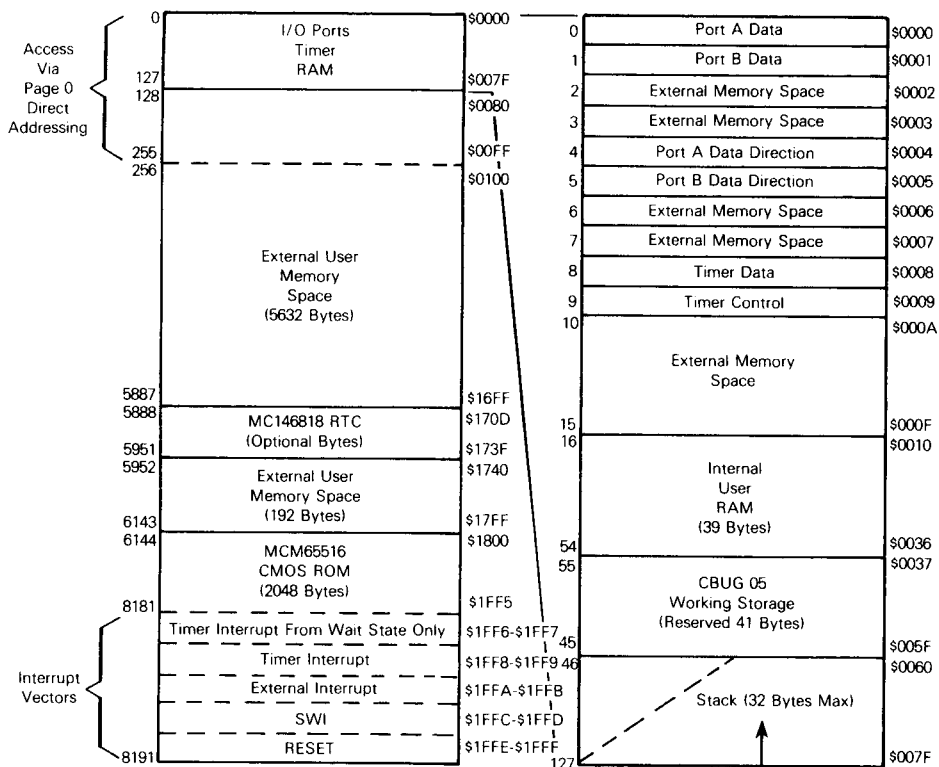


Figure 3. Address Map

2. CBUG05 COMMAND DESCRIPTION

Commands are entered in one of two ways:

(1) If the command requires no additional user input, then only the command key need be depressed; e.g., TR (CBUG05 will execute one instruction), and (2) If the command allows additional user input then the ENT key is used to enter the users input.

ESC will allow exit from all commands except STOP, V, L, & P once the ending address is entered.

- 1) RS — Reset MC146805E2
- 2) P — Punch cassette tape
- 3) L — Load cassette tape
- 4) V — Verify cassette tape against memory
- 5) ST — Set current time
- 6) DT — Display current time
- 7) OFF — Calculate branch offset
- 8) BP — Set/display breakpoints
- 9) BCL — Disable one or all breakpoints
- 10) TR — Execute one instruction
- 11) GO — Begin program execution

- 12) PC — Display user program counter
- 13) AR — Examine/change user accumulator
- 14) XR — Examine/change user index register
- 15) CC — Examine/change user condition code register
- 16) SP — Display user stack pointer
- 17) M — Examine/change memory contents
- 18) STOP — Put the system into a low power standby mode

- RS — 1) Automatic on power-up
- 2) Press RS to:
 - a) Return from STOP
 - b) Return to monitor when program control is lost

- STOP — 1) MC146805E2 oscillator is halted reducing current requirements
- 2) Command sequence:
 - a) Press STOP
 - b) Display will be cleared

- P — 1) Place recorder into the record mode
- 2) Press P
- 3) 'bA' will be displayed

- 4) Enter beginning address
 - 5) Press ENT
 - 6) 'EA' will be displayed
 - 7) Enter ending address
 - 8) Press ENT
 - 9) Prompt '=' will be returned when punch is finished
- L** — 1) Command sequence
- a) Press L
 - b) Display will be cleared
 - c) Depress PLAY on recorder
- 2) Valid loads will be followed by the prompt '='
- 3) Checksum errors are indicated by 'Err'
 - 4) Bad memory stores are indicated by displaying the address of the bad memory.
- V** — 1) Command sequence
- a) Press V
 - b) Display will be cleared
 - c) Depress PLAY on recorder
- 2) If the compare is successful, the prompt '=' is returned
- 3) Checksum errors are indicated by 'Err'
 - 4) If the compare is unsuccessful, the address of the memory location is displayed
- ST** — 1) MC146818 is used
- 2) Command sequence
- a) Press ST
 - b) '0000 A' will be displayed
 - c) Enter time in a 12 hour format
 - d) Press P for PM (AM is the default)
 - e) Press ENT
 - f) Prompt '=' will be returned
- DT** — 1) Press DT
- 2) current time will be displayed if MC146818 has been initialized
- OFF** — 1) Beginning and ending addresses point to the instruction opcode addresses
- 2) The opcode for the branch instruction must exist at the beginning address so the monitor can determine whether to do a bit branch or a conditional branch
- 3) Command sequence
- a) Press OFF
 - b) 'bA' will be displayed
 - c) Enter beginning address
 - d) Press ENT
 - e) 'EA' will be displayed
 - f) Enter ending address
 - g) Press ENT
- 4) If valid:
- a) 'USE xx' will be displayed.
 - b) xx will be loaded into beginning address + 2 for bit branches and address + 1 for conditional branches.
- 5) If not valid:
- a) Offset calculation result is displayed in 2's complement and 'Or' (out of range) is displayed
 - b) No change is made to instruction at the beginning address.
- BP** — 1) Three breakpoints are available: 0, 1, 2
- 2) Command sequence
- a) Press BP
 - b) Breakpoint 0 will be displayed
 - c) 'bOFF 0' will be displayed if breakpoint 0 is disabled
 - d) Enter new breakpoint address if desired
 - e) Press ENT
 - f) Next breakpoint will be displayed and open for entry
- BCL** — 1) Disable all breakpoints
- a) Press BCL
 - b) 'bCl' will be displayed
 - c) Press ENT
 - d) Prompt '=' will be returned
- 2) Disable only one breakpoint
- a) Press BCL
 - b) 'bCl' will be displayed
 - c) Enter the number of the breakpoint to be disabled
 - d) Press ENT
 - e) Prompt '=' will be returned
- TR** — 1) Press TR
- 2) The instruction located at the user PC will be executed
- 3) New user PC will be displayed
- GO** — 1) If breakpoints are enabled, the instruction at the breakpoint address will be executed and the PC of the next instruction to be executed will be displayed
- 2) Continue execution with the instruction at the user PC
- a) Press GO
 - b) Current user PC is displayed
 - c) Press ENT
- 3) Begin execution at new address
- a) Press GO
 - b) Current user PC is displayed
 - c) Enter the new PC address
 - d) Press ENT
- M** — 1) Press M
- 2) Last address will be displayed
- 3) Enter new address if desired
- 4) Press ENT
- 5) Address and contents of the address will be displayed in format 'aaaa xx'
- 6) Enter new contents if desired
- 7) Save (use one)
- a) Press ENT (next address and contents will be displayed)
 - b) Press M (previous address and contents will be displayed)
- PC** — 1) Not alterable
- 2) Command sequence
- a) Press PC
 - b) Current user PC displayed in format 'aaaa PC'

segments 0, 2, 4, 5, and 7. The decimal point is displayed by setting bit 3 of the display byte to a logical "1" (effectively adding eight to the display byte). Data in BCD or binary format is translated by CBUG05, into the display format, using a lookup table. CBUG05 then left-shifts the character to the MC145000 via port A of the MC146805E2.

Table 2. Display Format Conversions

Displayed Digit	Display Format Hex Code
0	D7
1	06
2	E3
3	A7
4	36
5	B5
6	F5
7	07
8	F7
9	B7
A	77
b	F4
C	D1
d	E6
E	F1
F	71
P	73
Y	B6
H	76
U	D6
L	D0
blank	00
- (dash)	20
= (equal)	A0
n	64
r	60
° (degrees)	33

NOTE: A Decimal Point can be added to all but the right-most display digit by setting b3 (segment (3)) to a 1.

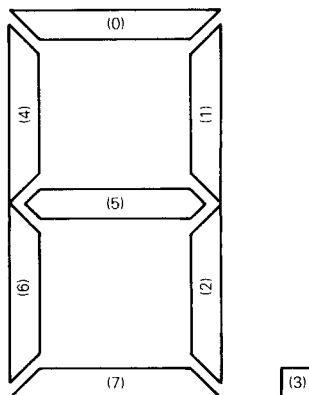


Figure 5. Display Digit Format

Several display routines are available for the user. Figure 6 describes the address, function, and use of these routines. All routines are called using a jump-to-subroutine (JSR) instruction. Most display outputs are initiated by filling a display table with all six characters in the display format to be displayed, then calling a routine (DISTAB) to display the entire table. In other words, the whole display is rewritten every time any character change is made. The display table is called DTABL (locations \$49-\$4E) and occupies six consecutive bytes where DTABL (location \$49) is the left most digit to be displayed.

5. KEYPAD INPUT

CBUG05 requires a 4×6 keypad such as is shown schematically in Figure 7. The six column lines are derived from a three-bit output from port A bits 4-6 driving a 3-to-8 decoder. By using this method port B is saved for the user. Figure 7 shows the required layout of the 4×6 keypad and 3-to-8 decoder. The keypad is continuously scanned for input. If an input is received, a 3075 MPU cycle debounce insures against spurious input. The required debounce time places a lower limit on the MPU clock frequency. At a 1 MHz bus speed (5 MHz oscillator input), the debounce time is about 3 ms. With a 10 kHz bus speed (50 kHz time base input), the debounce time is about 0.3 seconds. Debounce times of approximately 60 milliseconds or more require the keys be held down a longer time than an operator is normally accustomed.

Five routines are listed in Figure 8 of which two (COLUMN and DEBOUNC) are branch routines and one is a look-up table (STABL). One of the other two routines, KEYSCN, checks for a keyboard input and, if valid, returns it to the accumulator in a column-row format. This format can then be converted to a hexadecimal number which corresponds to the one key that was pressed (see STABL routine and Table 3). Note that hexadecimal numbers 0 through F correspond to the keypad keys 0 through F. The last routine of Figure 8, CHARIN, checks for a character and returns a hexadecimal number to the accumulator.

6. CASSETTE TAPE OPTION

The cassette tape option is included to allow for user program storage. Programs are stored in a modified bi-phase format (see Figure 9). The storage format used defines a zero as more than 300 MPU cycles between transitions and less than 300 MPU cycles between transitions. Data is punched with a start bit of one, eight bits of data and a zero stop bit. Tapes are punched with 16K zeros as a leader followed by a BOT and the ending and beginning addresses. The program is then punched followed by the checksum. Tapes are loaded after 256 consecutive zeros are read. The BOT then synchronizes the loading program. The ending and beginning addresses are loaded and data read and stored accordingly. Finally, the checksum is read and compared to the new computed checksum.

Baud rates are determined by the MPU cycle time. The software is set up to provide a default baud rate of 2400 baud if a one microsecond cycle time is used. Cycle times greater than one microsecond will decrease the baud rate proportionally.

```

*
*****
*
*      CLEAR DISPLAY TABLE
*
*      X REG DESTROYED
*
*****
*
1DF5 AE 05      A CLRTAB LDX      #5
1DF7 6F 49      A CLRLOC CLR      DTABL,X  CLEAR SIX
1DF9 5A                      DECX          LOCATIONS IN
1DFA 2A FB      1DF7      BPL      CLRLOC  DISPLAY TABLE
1DFC 81                      RTS

*
*****
*
*      DISPLAY TABLE CONTENTS
*
*      A,X REGISTERS DESTROYED
*
*****
*
1DFD AE 05      A DISTAB LDX      #5
1DFF E6 49      A DISCHR LDA      DTABL,X  LOAD DISPLAY
1E01 AD 09      1E0C      BSR      DISPLY  TABLE INTO
1E03 5A                      DECX          145000
1E04 2A F9      1DFF      BPL      DISCHR
1E06 81                      RTS

*
*****
*
*      BLANK DISPLAY
*
*      A,X REGISTERS DESTROYED
*
*****
*
1E07 AD EC      1DF5 CLRDIS BSR      CLRTAB  BLANK
1E09 AD F2      1DFD      BSR      DISTAB  DISPLAY
1E0B 81                      RTS

*
*****
*
*      SHIFT ONE CHARACTER INTO
*      DISPLAY
*
*      A REGISTER DESTROYED
*
*****
*
1E0C BF 50      A DISPLY STX      WORK1    SAVE INDEX
1E0E 1D 00      A          BCLR      6,PORTA  CLEAR DATA
1E10 AE 08      A          LDX      #8
1E12 48                      DIS1  LSLA      SET UP
1E13 24 02      1E17      BCC      DIS2     BIT OF
1E15 1C 00      A          BSET      6,PORTA  ACCUMULATOR
1E17 1E 00      A DIS2     BSET      7,PORTA  CLOCK
1E19 1F 00      A          BCLR      7,PORTA  IT
1E1B 1D 00      A          BCLR      6,PORTA  CLEAR DATA
1E1D 5A                      DECX          COMPLETE?
1E1E 26 F2      1E12      BNE      DIS1     NO
1E20 BE 50      A          LDX      WORK1    RESTORE INDEX
1E22 81                      RTS

```

Figure 6. Display Routines

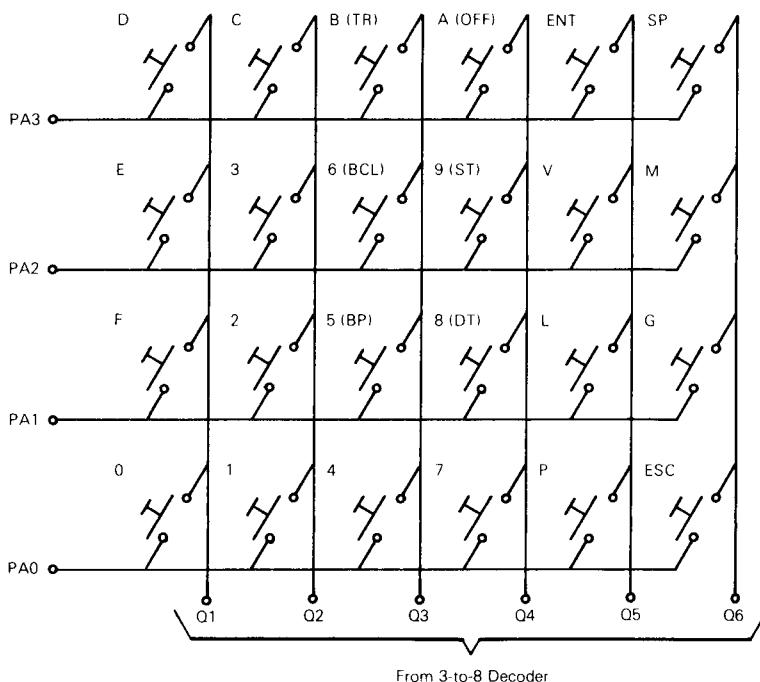


Figure 7. 4x6 Keypad Schematic Diagram

Table 3. Keypad Cross-Reference

KEYPAD CHARACTER	PORT-A DATA	HEXADECIMAL (\$) EQUIVALENT
0 (P.C.)	11	0
F (AR)	12	F
E (XR)	14	E
D (CC)	18	D
1 --	21	1
2 --	22	2
3 --	24	3
C --	28	C
4	31	4
5 (BP)	32	5
6 (B.CL.)	34	6
B (TR)	38	B
7	41	7
8 (DT.)	42	8
9 (ST.)	44	9
A (OFF)	48	A
P	51	17
L	52	16
V	54	15
ENT	58	11
ESC	61	10
G	62	14
M	64	13
SP	68	12

Whatever baud rate is used, the cassette tape and recorder must have an upper frequency response 2-3 times the baud rate and a lower frequency response of 1/2 - 1/3 the baud rate to insure reliability.

7. MC146818 REAL-TIME CLOCK (RTC) OPTION

The RTC can be added to a system to provide time, date, periodic interrupt and many other user functions (see MC146818 ADI-856). The RTC time may be set and displayed using CBUG05 software; however, only the 12-hour mode is available. The displayed time is updated once per second after polling the Update-In-Progress bit (UIP) for a zero. All MC146818 functions are available to the user. The CBUG05 software set and display time routines require that a 4.194304 MHz crystal be used; however, if power consumption is critical then either a 1.04576 MHz or 32.678 KHz oscillator input could be used. The user would be required to set-up the divider chain in the RTC for the particular time base used.

8. INTERNAL AND EXTERNAL MEMORY SPACE

The internal memory space is located in the first 128 bytes of memory and contains the timer registers, I/O port registers, and 112 bytes of RAM. External memory can be mapped at the same addresses as the internal memory space. An MPU write to internal memory space is duplicated externally; however, an MPU read of internal locations will result in only the internal data being recognized. This allows the user to map large memories externally without requiring that accesses to internal memory locations be excluded from the external memory, thus, simplifying external address decoding.

```

*****
*
*      KEYPAD SCAN
*
*      X REGISTER DESTROYED
*
*      A REGISTER CONTAINS VALUE
*
*      CARRY SET IF VALID OUTPUT
*
*****
*
1E23 98      KEYSCN CLC
1E24 4F      CLRA
1E25 AE 06      A      LDX      #6      SETUP
1E27 AB 10      A KEY1  ADD      #$10    ROW
1E29 B7 00      A      STA      PORTA
1E2B AD 06      1E33    BSR      COLUMN  CHECK COLUMNS
1E2D 25 03      1E32    BCS      KEY2    IF VALID GET OUT
1E2F 5A          DECX          ELSE TRY
1E30 26 F5      1E27    BNE      KEY1    NEXT ROW
1E32 81          KEY2  RTS
*
*****
*
*      CHECK FOR KEY CLOSURE
*
*      WITHIN COLUMN AND DEBOUNCE
*
*      A REGISTER CONTAINS VALUE
*
*      CARRY SET IF VALID OUTPUT
*
*****
*
1E33 B6 00      A COLUMN LDA      PORTA  READ KEYPAD
1E35 B7 50      A      STA      WORK1   STORE IT
1E37 A5 0F      A      BIT      #$0F    KEY CLOSED?
1E39 27 19      1E54    BEQ      COLRET  NO GET OUT
1E3B AD 18      1E55    BSR      DBOUNC  ELSE DEBOUNCE
1E3D B6 00      A      LDA      PORTA  RE-READ KEYPAD
1E3F B1 50      A      CMP      WORK1   SAME KEY CLOSED?
1E41 26 11      1E54    BNE      COLRET  NO GET OUT
1E43 99          SEC          SET FLAG FOR VALID
1E44 B6 00      A COL1  LDA      PORTA  KEY
1E46 A5 0F      A      BIT      #$0F    RELEASED?
1E48 26 FA      1E44    BNE      COL1   NO TRY AGAIN
1E4A AD 09      1E55    BSR      DBOUNC  YES DEBOUNCE
1E4C B6 00      A      LDA      PORTA  STILL
1E4E A5 0F      A      BIT      #$0F    RELEASED?
1E50 26 F2      1E44    BNE      COL1   NO TRY AGAIN
1E52 B6 50      A      LDA      WORK1   RETURN CHAR IN A-REG

```

Figure 8. KEYSCN, COLUMN, DEBOUNC, CHRIN, and STABL Routines

```

1E54 81          COLRET RTS          YES GO HOME
*
*****
*
*          PAUSE FOR 3075 CYCLES          *
*
*          A REGISTER DESTROYED          *
*
*****
*
1E55 A6 FF      A DBOUNC LDA    #$FF      PAUSE
1E57 21 FE      1E57 DLOOP BRN    *        256X12
1E59 21 FE      1E59          BRN    *        CYCLES
1E5B 4A          DECA          OR AT
1E5C 26 F9      1E57 BNE      DLOOP LEAST
1E5E 81          RTS          3.0 MS
*

*
*****
*
*          INPUT ONE CHARACTER          *
*
*          A REGISTER CONTAINS HEX VALUE *
*
*          X REGISTER CONTAINS HEX VALUE *
*
*****
*
1E5F          1E5F A CHRIN EQU    *
1E5F CD 1E23      A      JSR    KEYSCN  GET KEY
1E62 24 FB      1E5F          BCC    CHRIN  IF NOT VALID RETRY
1E64 5F          CLRX
1E65 D1 1E6F      A CHRIN1 CMP    STABL,X  CONVERT
1E68 27 03      1E6D BEQ    CHRIN2  TO HEX
1E6A 5C          INCX
1E6B 20 F8      1E65 BRA      CHRIN1
1E6D 9F          CHRIN2 TXA          IF CANCEL
1E6E 81          RTS

*
*****
*
*          CONVERSION TABLE FOR KEYPAD *
*          TO HEX NUMBER          *
*
*****
*
1E6F 11          A STABL FCB    $11      0
1E70 21          A      FCB    $21      1
1E71 22          A      FCB    $22      2
1E72 24          A      FCB    $24      3
1E73 31          A      FCB    $31      4
1E74 32          A      FCB    $32      5
1E75 34          A      FCB    $34      6
1E76 41          A      FCB    $41      7
1E77 42          A      FCB    $42      8
1E78 44          A      FCB    $44      9

```

Figure 8. KEYSCN, COLUMN, DEBOUNC, CHRIN, and STABL Routines (Cont'd)

1E79	48	A	FCB	\$48	A
1E7A	38	A	FCB	\$38	B
1E7B	28	A	FCB	\$28	C
1E7C	18	A	FCB	\$18	D
1E7D	14	A	FCB	\$14	E
1E7E	12	A	FCB	\$12	F
1E7F	61	A	FCB	\$61	CANCEL COMMAND
1E80	58	A	FCB	\$58	ENTER COMMAND
1E81	68	A	FCB	\$68	STACK POINTER
1E82	64	A	FCB	\$64	MEMORY
1E83	62	A	FCB	\$62	GO
1E84	54	A	FCB	\$54	VERIFY TAPE
1E85	52	A	FCB	\$52	LOAD TAPE
1E86	51	A	FCB	\$51	PUNCH TAPE

*

 *
 * HEX TO MUX DISPLAY *
 * CONVERSION TABLE *
 *

Figure 8. KEYSCN, COLUMN, DEBOUNC, CHRIN, and STBL Routines (Cont'd)

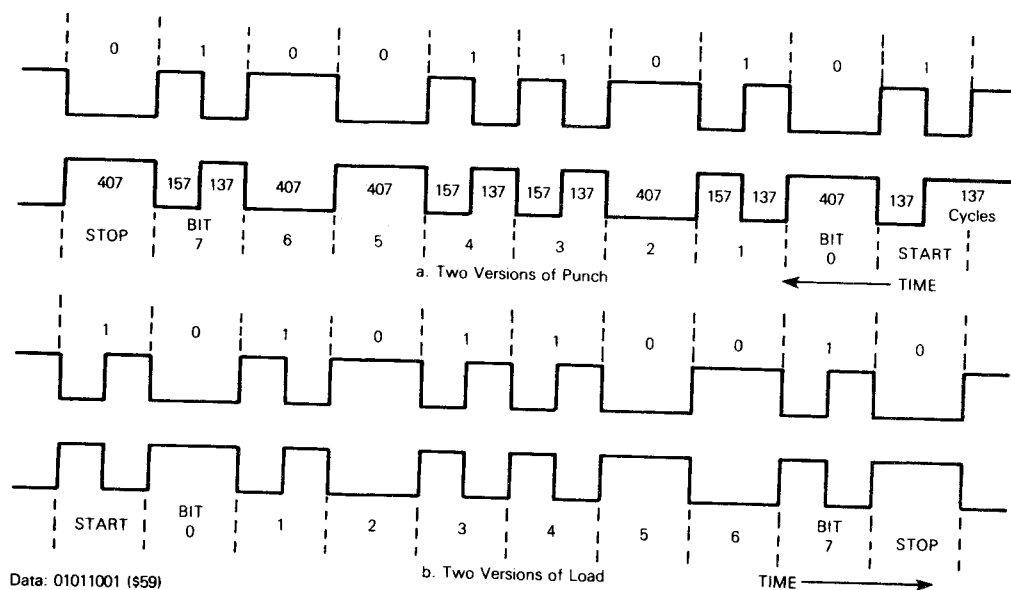


Figure 9. Example of Serial Data Formats for Punch and Load

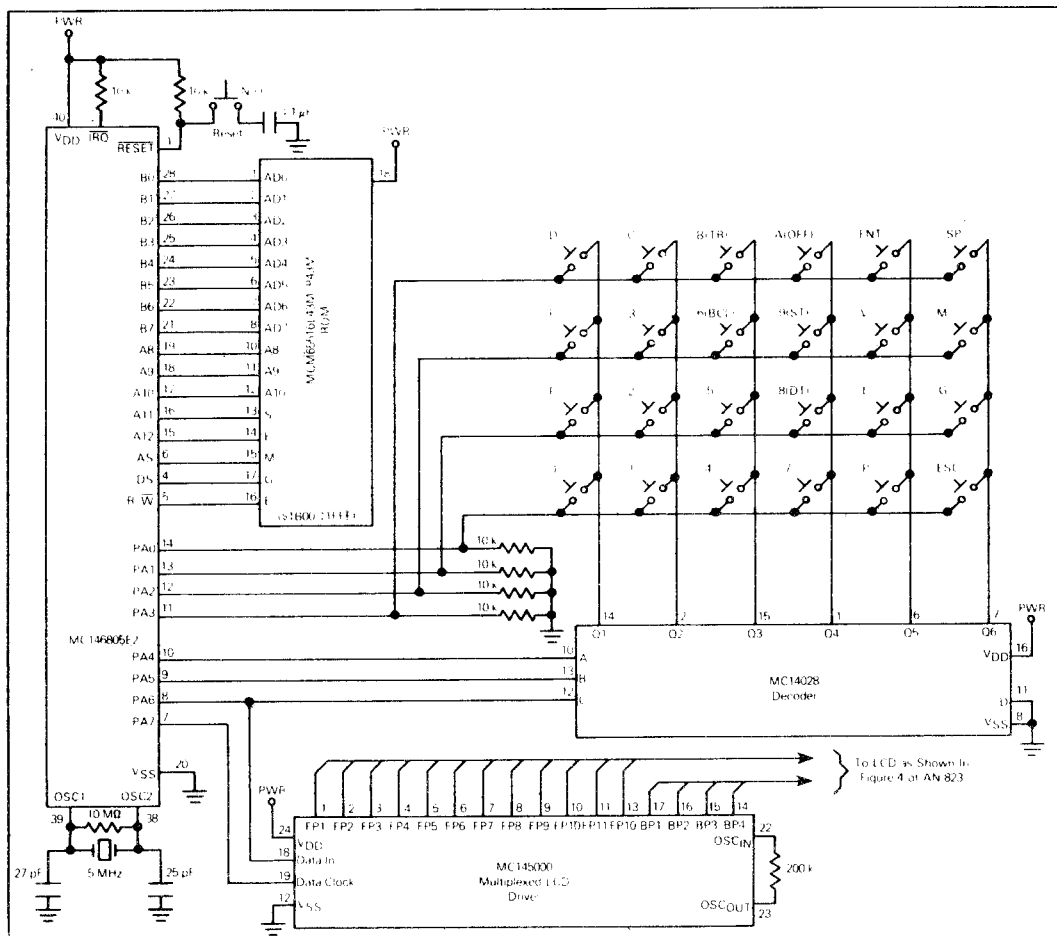


Figure 10. CBUG05 Minimum CMOS Only System Schematic Diagram

PAGE 001 CBUG05 .SA:1

			OPT	CMOS
00001				
00002		*		
00003	0000	A PORTA	EQU	0
00004	0004	A PORTAD	EQU	4
00005	0001	A PORTB	EQU	1
00006	0008	A TIMER	EQU	8
00007	0009	A TIMEC	EQU	9
00008	170A	A CR1	EQU	\$170A
00009	170B	A CR2	EQU	\$170B
00010	1700	A SEC	EQU	\$1700
00011	1702	A MIN	EQU	\$1702
00012	1704	A HOUR	EQU	\$1704
00013	1707	A DAY	EQU	\$1707
00014	1708	A MONTH	EQU	\$1708
00015	1709	A YEAR	EQU	\$1709
00016	1800	A MONSTR	EQU	\$1800
00017	001F	A PCMASK	EQU	\$1F
00018	0003	A NUMBKP	EQU	3
00019	00A0	A PROMPT	EQU	\$A0
00020	00CC	A LJMP	EQU	\$CC
00021	0083	A SWIOP	EQU	\$83
00022		*		
00023A 0040			ORG	\$40
00024		*		
00025	0037	A BKPTBL	EQU	*-3*NUMBKP
00026A 0040	0003	A IRQ	RMB	3
00027A 0043	0003	A TIRO	RMB	3
00028A 0046	0003	A TIROW	RMB	3
00029A 0049	0006	A DTABL	RMB	6
00030A 004F	0001	A SWIFLG	RMB	1
00031A 0050	0001	A WORK1	RMB	1
00032A 0051	0001	A WORK2	RMB	1
00033A 0052	0001	A ADDRH	RMB	1
00034A 0053	0001	A ADDR1	RMB	1
00035A 0054	0001	A WORK3	RMB	1
00036A 0055	0001	A WORK4	RMB	1
00037A 0056	0001	A WORK5	RMB	1
00038A 0057	0001	A WORK6	RMB	1
00039A 0058	0002	A TEMP	RMB	2
00040A 005A	0001	A PNCNT	RMB	1
00041A 005B	0001	A CHKSUM	RMB	1
00042A 005C	0001	A SREF	RMB	1
00043A 005D	0001	A LCNT	RMB	1
00044A 005E	0001	A PCNT1	RMB	1
00045A 005F	0001	A PCNT0	RMB	1
00046		*		

```

00048
00049A 1800          *          ORG      $1800
00050          *
00051A 1800 A6 F0      A RESET  LDA      #$F0      SETUP PORT
00052A 1802 B7 04      A          STA      PORTAD   FOR KEYPAD
00053A 1804 3F 00      A          CLR      PORTA    AND DISPLAY
00054A 1806 3F 5C      A          CLR      SREF     INITIALIZE
00055A 1808 A6 0F      A          LDA      #$F      TAPE SOFTWARE
00056A 180A B7 5D      A          STA      LCNT     FOR 2400 BAUD
00057A 180C A6 12      A          LDA      #$12
00058A 180E B7 5E      A          STA      PCNT1
00059A 1810 A6 26      A          LDA      #$26
00060A 1812 B7 5F      A          STA      PCNT0
00061          *
00062A 1814          1FC5      A VECTOR FDB      IRQV      SET-UP
00063A 1816          1FC7      A          FDB      TIRQV    INTERRUPT
00064A 1818          1FC4      A          FDB      TIRQWV   VECTORS
00065A 181A A6 CC        A          LDA      #LJMP    IN RAM
00066A 181C B7 40      A          STA      IRQ
00067A 181E B7 43      A          STA      TIRQ
00068A 1820 B7 46      A          STA      TIRQW
00069A 1822 C6 1814      A          LDA      VECTOR
00070A 1825 B7 41      A          STA      IRQ+1
00071A 1827 C6 1815      A          LDA      VECTOR+1
00072A 182A B7 42      A          STA      IRQ+2
00073A 182C C6 1816      A          LDA      VECTOR+2
00074A 182F B7 44      A          STA      TIRQ+1
00075A 1831 C6 1817      A          LDA      VECTOR+3
00076A 1834 B7 45      A          STA      TIRQ+2
00077A 1836 C6 1818      A          LDA      VECTOR+4
00078A 1839 B7 47      A          STA      TIRQW+1
00079A 183B C6 1819      A          LDA      VECTOR+5
00080A 183E B7 48      A          STA      TIRQW+2
00081          *
00082A 1840 AE 4F        A          LDX      #SWIFLG
00083A 1842 7F          A          INIT  CLR      0,X      CLEAR
00084A 1843 5C          A          INCX                     WORKING
00085A 1844 A3 56          A          CPX      #WORK5     STORAGE
00086A 1846 23 FA        A          1842  BLS      INIT
00087A 1848 CD 1DD3      A          JSR      SCNBKP    CLEAR
00088A 184B A6 FF        A          LDA      #$FF      ALL
00089A 184D F7          A          REBCLR STA      0,X      BREAKPOINTS
00090A 184E 5C          A          INCX
00091A 184F 5C          A          INCX
00092A 1850 5C          A          INCX
00093A 1851 3A 5A        A          DEC      PNCNT
00094A 1853 26 F8        A          184D  BNE      REBCLR
00095A 1855 83          A          SWI
00096          *
00097          1856      A SWI      EQU      *
00098A 1856 00 4F 04    185D      BRSET   0,SWIFLG,SWCHK FROM RESET?
00099A 1859 10 4F        A          BSET   0,SWIFLG YES
00100A 185B 20 4E        18AB      BRA      GETCMD
00101A 185D CD 1DD3      A SWCHK   JSR      SCNBKP    REMOVE
00102A 1860 F6          A SWIREP  LDA      0,X      BREAKPOINTS
00103A 1861 2B 0B        186E      BMI      SWINOB
00104A 1863 B7 52        A          STA      ADDRH
00105A 1865 E6 01        A          LDA      1,X

```

PAGE 003 CBUG05 .SA:1

00106A	1867	B7 53	A	STA	ADDRL	
00107A	1869	E6 02	A	LDA	2,X	
00108A	186B	CD 1F24	A	JSR	STORE	
00109A	186E	5C		SWINOB	INCX	GET NEXT B.P.
00110A	186F	5C		INCX		
00111A	1870	5C		INCX		
00112A	1871	3A 5A	A	DEC	PNCNT	
00113A	1873	26 EB	1860	BNE	SWIREP	
00114			*			
00115A	1875	CD 1916	A	JSR	LOCSTK	FIND STACK
00116A	1878	E6 08	A	LDA	8,X	
00117A	187A	A0 01	A	SUB	#1	ADJUST
00118A	187C	B7 59	A	STA	TEMP+1	
00119A	187E	E6 07	A	LDA	7,X	
00120A	1880	A2 00	A	SBC	#0	
00121A	1882	B7 58	A	STA	TEMP	
00122A	1884	BF 57	A	STX	WORK6	SAVE STACK LOCATION
00123A	1886	CD 1DD3	A	JSR	SCNBKP	SETUP B.P. SCAN
00124A	1889	F6		SWITRY	LDA	ADJUSTED P.C.
00125A	188A	2B 15	18A1	BMI	SWICMP	IN B.P. TABLE?
00126A	188C	B1 58	A	CMP	TEMP	
00127A	188E	26 11	18A1	BNE	SWICMP	
00128A	1890	E6 01	A	LDA	1,X	
00129A	1892	B1 59	A	CMP	TEMP+1	
00130A	1894	26 0B	18A1	BNE	SWICMP	NO,TRY AGAIN
00131A	1896	BE 57	A	LDX	WORK6	YES,RESTORE S.P.
00132A	1898	E7 08	A	STA	8,X	PUT ADJUSTED P.C.
00133A	189A	B6 58	A	LDA	TEMP	INTO STACK
00134A	189C	E7 07	A	STA	7,X	
00135A	189E	CC 1B31	A	JMP	TRACE	EXECUTE 1 INSTRUCTION
00136A	18A1	5C		SWICMP	INCX	NEXT B.P.
00137A	18A2	5C		INCX		
00138A	18A3	5C		INCX		
00139A	18A4	3A 5A	A	DEC	PNCNT	
00140A	18A6	26 E1	1889	BNE	SWITRY	DONE?
00141A	18A8	CC 1928	A	JMP	PCOUNT	YES PRINT P.C.
00142			*			
00143		18AB	A	GETCMD	EQU	*
00144A	18AB	CD 1DF5	A	JSR	CLRTAB	
00145A	18AE	A6 A0	A	LDA	#PROMPT	PRINT
00146A	18B0	B7 49	A	STA	DTABL	'='
00147A	18B2	CD 1DFD	A	JSR	DISTAB	PROMPT
00148			*			
00149A	18B5	CD 1E23	A	CMDSCN	JSR	
00150A	18B8	24 FB	18B5	BCC	KEYSCN	CHECK KEYPAD
00151A	18BA	5F		CLR X	CMDSCN	
00152A	18BB	B7 50	A	STA	WORK1	
00153A	18BD	D6 18D2	A	RJUMP	PTABL,X	THIS COMMAND?
00154A	18C0	B1 50	A	CMP	WORK1	
00155A	18C2	27 0A	18CE	BEQ	PJUMP	YES
00156A	18C4	A1 68	A	CMP	#S68	
00157A	18C6	27 E3	18AB	BEQ	GETCMD	
00158A	18C8	5C		INCX		NO
00159A	18C9	5C		INCX		GO TO
00160A	18CA	5C		INCX		NEXT
00161A	18CB	5C		INCX		POSSIBLE
00162A	18CC	20 EF	18BD	BRA	RJUMP	TRY AGAIN
00163A	18CE	5C		PJUMP	INCX	GO TO

PAGE 004 CBUG05 .SA:1

00164A 18CF DC 18D2 A JMP PTABL,X COMMAND
00165 *

PAGE 005 CBUG05 .SA:1

00167			*				
00168A 18D2	11	A	PTABL	FCB	\$11		
00169A 18D3	CC	A		FCB	LJMP		
00170A 18D4	1928	A		FDB	PCOUNT	PROGRAM COUNTER	
00171A 18D6	12	A		FCB	\$12		
00172A 18D7	CC	A		FCB	LJMP		
00173A 18D8	1940	A		FDB	AREG	ACCUMULATOR	
00174A 18DA	14	A		FCB	\$14		
00175A 18DB	CC	A		FCB	LJMP		
00176A 18DC	195A	A		FDB	XREG	INDEX REGISTER	
00177A 18DE	18	A		FCB	\$18		
00178A 18DF	CC	A		FCB	LJMP		
00179A 18E0	1977	A		FDB	CCODE	CONDITION CODE	
00180			*				
00181A 18E2	28	A		FCB	\$28		
00182A 18E3	CC	A		FCB	LJMP		
00183A 18E4	1FD7	A		FDB	PWRDWN	UNUSED	
00184			*				
00185A 18E6	32	A		FCB	\$32		
00186A 18E7	CC	A		FCB	LJMP		
00187A 18E8	1A78	A		FDB	BPDIS	DISPLAY/SET BP	
00188A 18EA	34	A		FCB	\$34		
00189A 18EB	CC	A		FCB	LJMP		
00190A 18EC	1AD6	A		FDB	BPCLR	CLEAR BP	
00191A 18EE	38	A		FCB	\$38		
00192A 18EF	CC	A		FCB	LJMP		
00193A 18F0	1B31	A		FDB	TRACE	TRACE ONE INSTRUCTION	
00194			*				
00195A 18F2	42	A		FCB	\$42		
00196A 18F3	CC	A		FCB	LJMP		
00197A 18F4	1C0B	A		FDB	DTIME	DISPLAY TIME	
00198A 18F6	44	A		FCB	\$44		
00199A 18F7	CC	A		FCB	LJMP		
00200A 18F8	1B86	A		FDB	STIME	SET TIME	
00201A 18FA	48	A		FCB	\$48		
00202A 18FB	CC	A		FCB	LJMP		
00203A 18FC	19E5	A		FDB	OFFSET	OFFSET CALCULATION	
00204			*				
00205A 18FE	51	A		FCB	\$51		
00206A 18FF	CC	A		FCB	LJMP		
00207A 1900	1C35	A		FDB	PUNCH	PUNCH TAPE	
00208A 1902	52	A		FCB	\$52		
00209A 1903	CC	A		FCB	LJMP		
00210A 1904	1CDD	A		FDB	TLOAD	LOAD TAPE	
00211A 1906	54	A		FCB	\$54		
00212A 1907	CC	A		FCB	LJMP		
00213A 1908	1D81	A		FDB	VERIFY	VERIFY TAPE	
00214			*				
00215A 190A	62	A		FCB	\$62		
00216A 190B	CC	A		FCB	LJMP		
00217A 190C	1D8F	A		FDB	GO	GO	
00218A 190E	64	A		FCB	\$64		
00219A 190F	CC	A		FCB	LJMP		
00220A 1910	1EAA	A		FDB	MEMEX	MEMORY	
00221A 1912	68	A		FCB	\$68		
00222A 1913	CC	A		FCB	LJMP		
00223A 1914	1DDA	A		FDB	STACK	STACK	
00224			*				

PAGE 007 CBUG05 .SA:1

```

00284A 194F AB 05      A      ADD      #5
00285A 1951 3F 52      A      CLR      ADDRH      SETUP FOR
00286A 1953 B7 53      A      STA      ADDRHL     EXAMINE/CHANGE
00287A 1955 1C 4F      A      BSET     6,SWIFLG
00288A 1957 CC 1EB1    A      JMP      MEMEX3     USING MEMORY ROUTINE
00289
00290
00291
00292
00293
00294
00295
00296
00297A 195A A6 06      A XREG  EQU      *
00298A 195C CD 1DF5    A      LDA      #6      PRINT 'ID'
00299A 195F B7 4A      A      JSR      CLRTAB
00300A 1961 A6 E6      A      STA      DTABL+1
00301A 1963 B7 4B      A      LDA      #SE6
00302A 1965 A6 60      A      STA      DTABL+2
00303A 1967 B7 4C      A      LDA      #S60
00304A 1969 AD AB      A      STA      DTABL+3
00305A 196B 9F      1916 BSR      LOCSTK     FIND INDEX
00306A 196C AB 06      A      TXA      REGISTER VALUE
00307A 196E 3F 52      A      ADD      #6
00308A 1970 B7 53      A      CLR      ADDRH      SETUP FOR
00309A 1972 1C 4F      A      STA      ADDRHL     EXAMINE/CHANGE
00310A 1974 CC 1EB1    A      BSET     6,SWIFLG
00311
00312
00313
00314
00315
00316
00317
00318
00319
00320A 1977 CD 1DF5    A CCODE EQU      *
00321A 197A A6 D1      A      JSR      CLRTAB
00322A 197C B7 49      A      LDA      #SD1
00323A 197E A6 D7      A      STA      DTABL
00324A 1980 B7 4A      A      LDA      #SD7
00325A 1982 A6 E6      A      STA      DTABL+1
00326A 1984 B7 4B      A      LDA      #SE6
00327A 1986 A6 F1      A      STA      DTABL+2
00328A 1988 B7 4C      A      LDA      #SF1
00329A 198A AD 8A      A      STA      DTABL+3
00330A 198C 9F      1916 BSR      LOCSTK     FIND CONDITION
00331A 198D AB 04      A      TXA      CODES
00332A 198F 3F 52      A      ADD      #4
00333A 1991 B7 53      A      CLR      ADDRH      SETUP FOR
00334A 1993 1C 4F      A      STA      ADDRHL     EXAMINE/CHANGE
00335A 1995 CC 1EB1    A      BSET     6,SWIFLG
00336
00337
00338
00339
00340
00341
00337
00338
00339
00340
00341

```

```

00342          *
00343          *          TEMP,TEMP+1 : BEGINNING *
00344          *          ADDRH,ADDRL : ENDING   *
00345          *
00346          *****
00347          *
00348A 1998 19 4F      A BLDRNG BCLR 4,SWIFLG
00349A 199A 17 4F      A          BCLR 3,SWIFLG
00350A 199C CD 1DF5    A          JSR CLRTAB PRINT
00351A 199F A6 F4      A          LDA #F4 'BA'
00352A 19A1 B7 4D      A          STA DTABL+4
00353A 19A3 A6 77      A          LDA #$77
00354A 19A5 B7 4E      A          STA DTABL+5
00355A 19A7 CD 1DFD    A          JSR DISTAB
00356A 19AA CD 1F58    A          JSR BLDADR GET SOURCE ADDR.
00357A 19AD 24 2C      19DB BCC BLDNR1 VALID?
00358A 19AF B6 52      A          LDA ADDRH YES
00359A 19B1 A1 1F      A          CMP #PCMASK TOO BIG?
00360A 19B3 22 2A      19DF BHI BLDNR2 YES
00361A 19B5 B7 58      A          STA TEMP NO SAVE IT
00362A 19B7 B6 53      A          LDA ADDR
00363A 19B9 B7 59      A          STA TEMP+1
00364A 19BB CD 1F15    A          JSR LOAD FETCH OPCODE OF INSTR.
00365A 19BE B7 57      A          STA WORK6 SAVE IT
00366A 19C0 CD 1DF5    A          JSR CLRTAB
00367A 19C3 A6 F1      A          LDA #F1 PRINT 'EA'
00368A 19C5 B7 4D      A          STA DTABL+4
00369A 19C7 A6 77      A          LDA #$77
00370A 19C9 B7 4E      A          STA DTABL+5
00371A 19CB CD 1DFD    A          JSR DISTAB
00372A 19CE CD 1F58    A          JSR BLDADR GET DESTINATION ADDR
00373A 19D1 24 08      19DB BCC BLDNR1 VALID?
00374A 19D3 B6 52      A          LDA ADDRH YES
00375A 19D5 A1 1F      A          CMP #PCMASK TOO BIG?
00376A 19D7 22 06      19DF BHI BLDNR2 YES
00377A 19D9 20 06      19E1 BRA BLDRET
00378A 19DB 18 4F      A BLDNR1 BSET 4,SWIFLG INVALID
00379A 19DD 20 02      19E1 BRA BLDRET
00380A 19DF 16 4F      A BLDNR2 BSET 3,SWIFLG TOO BIG
00381A 19E1 81          BLDRET RTS
00382          *
00383          *****
00384          *
00385          *          CALCULATE BRANCH OFFSET *
00386          *          FOR BIT AND CONDITIONAL *
00387          *          BRANCHES *
00388          *
00389          *          OPCODE MUST BE AT *
00390          *          BEGINNING ADDRESS *
00391          *
00392          *          OFFSET WILL BE INSERTED *
00393          *          INTO BRANCH INSTRUCTION *
00394          *
00395          *****
00396          *
00397A 19E2 CC 1E97    A OFFERR JMP ERROR
00398          *
00399          19E5      A OFFSET EQU *

```

PAGE 009 CBUG05 .SA:1

00400A	19E5	AD	B1	1998		BSR	BLDRNG	
00401A	19E7	08	4F	2B 1A15		BRSET	4,SWIFLG,ORET	
00402A	19EA	06	4F	F5 19E2		BRSET	3,SWIFLG,OFFERR	
00403A	19ED	B6	53		A	LDA	ADDRL	NO FIND APPARRENT
00404A	19EF	B0	59		A	SUB	TEMP+1	OFFSET
00405A	19F1	A0	02		A	SUB	#2	
00406A	19F3	B7	53		A	STA	ADDRL	
00407A	19F5	B6	52		A	LDA	ADDRH	
00408A	19F7	B2	58		A	SBC	TEMP	
00409A	19F9	B7	52		A	STA	ADDRH	
00410A	19FB	B6	57		A	LDA	WORK6	CHECK OPCODE
00411A	19FD	A1	1F		A	CMP	#\$1F	FOR BIT BRANCH
00412A	19FF	23	41	1A42		BLS	OFFST1	
00413A	1A01	B6	52		A	LDA	ADDRH	
00414A	1A03	A1	FF		A	CMP	#\$FF	+ OR - OFFSET?
00415A	1A05	27	03	1A0A		BEQ	OFFST2	
00416A	1A07	4D				TSTA		CHECK OFFSET
00417A	1A08	26	60	1A6A		BNE	OVRERR	FOR +/- 0
00418A	1A0A	B6	53		A OFFST2	LDA	ADDRL	
00419A	1A0C	A1	FF		A	CMP	#\$FF	
00420A	1A0E	27	5A	1A6A		BEQ	OVRERR	
00421A	1A10	AD	06	1A18		BSR	USE	PRINT IT IF VALID
00422A	1A12	CC	18B5		A	JMP	CMDSCN	
00423A	1A15	CC	18AB		A ORET	JMP	GETCMD	
00424					*			
00425A	1A18	CD	1DF5		A USE	JSR	CLRTAB	
00426A	1A1B	A6	D6		A	LDA	#\$D6	PRINT 'USED'
00427A	1A1D	B7	49		A	STA	DTABL	
00428A	1A1F	A6	B5		A	LDA	#\$B5	
00429A	1A21	B7	4A		A	STA	DTABL+1	
00430A	1A23	A6	F1		A	LDA	#\$F1	
00431A	1A25	B7	4B		A	STA	DTABL+2	
00432A	1A27	A6	E6		A	LDA	#\$E6	
00433A	1A29	B7	4C		A	STA	DTABL+3	
00434A	1A2B	B6	53		A	LDA	ADDRL	PRINT OFFSET
00435A	1A2D	CD	1F8C		A	JSR	PRTDAT	
00436A	1A30	97				TAX		
00437A	1A31	B6	59		A	LDA	TEMP+1	
00438A	1A33	AB	01		A	ADD	#1	
00439A	1A35	B7	53		A	STA	ADDRL	
00440A	1A37	B6	58		A	LDA	TEMP	
00441A	1A39	A9	00		A	ADC	#0	PUT INTO
00442A	1A3B	B7	52		A	STA	ADDRH	INSTRUCTION
00443A	1A3D	9F				TXA		
00444A	1A3E	CD	1F24		A	JSR	STORE	
00445A	1A41	81				RTS		
00446					*			
00447A	1A42	B6	53		A OFFST1	LDA	ADDRL	ADJUST FOR
00448A	1A44	A0	01		A	SUB	#1	BIT BRANCH
00449A	1A46	B7	53		A	STA	ADDRL	
00450A	1A48	B6	52		A	LDA	ADDRH	
00451A	1A4A	A2	00		A	SBC	#0	
00452A	1A4C	B7	52		A	STA	ADDRH	
00453A	1A4E	A1	FF		A	CMP	#\$FF	NEG OFFSET?
00454A	1A50	27	03	1A55		BEQ	OFFST3	YES
00455A	1A52	4D				TSTA		CHECK FOR
00456A	1A53	26	15	1A6A		BNE	OVRERR	+/- 0 AND -1
00457A	1A55	B6	53		A OFFST3	LDA	ADDRL	

PAGE 010 CBUG05 .SA:1

```

00458A 1A57 A1 FF      A      CMP      #$FF
00459A 1A59 27 0F      1A6A     BEQ      OVRERR
00460A 1A5B A1 FE      A      CMP      #$FE
00461A 1A5D 27 0B      1A6A     BEQ      OVRERR
00462A 1A5F 3C 59      A      INC      TEMP+1
00463A 1A61 26 02      1A65     BNE      OFFITS
00464A 1A63 3C 58      A      INC      TEMP
00465A 1A65 AD B1      1A18  OFFITS BSR      USE      PRINT IF VALID
00466A 1A67 CC 18B5    A      JMP      CMDSCN
00467
00468A 1A6A A6 D7      A      OVRERR LDA      #$D7      PRINT 'OR'
00469A 1A6C B7 4D      A      STA      DTABL+4
00470A 1A6E A6 60      A      LDA      #$60
00471A 1A70 B7 4E      A      STA      DTABL+5
00472A 1A72 CD 1FB0    A      JSR      PRTADR
00473A 1A75 CC 18B5    A      JMP      CMDSCN
00474
00475
00476
00477
00478
00479
00480
00481
00482A 1A78 3F 57      A      BPDIS EQU      *
00483A 1A7A 3A 57      A      CLR      WORK6
00484A 1A7C CD 1DD3    A      DEC      WORK6
00485A 1A7F BF 51      A      JSR      SCNBKP      FIND B.P. TABLE
00486A 1A81 3F 4D      A      STX      WORK2
00487A 1A83 F6         A      BPDIS1 CLR      DTABL+4
00488A 1A84 2A 10      1A96     LDA      0,X      GET B.P.
00489A 1A86 A6 F4      A      BPL      BPDIS2 VALID?
00490A 1A88 B7 49      A      LDA      #$F4      NO
00491A 1A8A A6 D7      A      STA      DTABL      PRINT 'BOFF'
00492A 1A8C B7 4A      A      LDA      #$D7
00493A 1A8E A6 71      A      STA      DTABL+1
00494A 1A90 B7 4B      A      LDA      #$71
00495A 1A92 B7 4C      A      STA      DTABL+2
00496A 1A94 20 09      A      STA      DTABL+3
00497A 1A96 B7 52      1A9F     BRA      BPDIS4
00498A 1A98 E6 01      A      BPDIS2 STA      ADDR      PRINT B.P.
00499A 1A9A B7 53      A      LDA      1,X
00500A 1A9C CD 1FB0    A      STA      ADDR
00501A 1A9F 3C 57      A      JSR      PRTADR
00502A 1AA1 BE 57      A      JSR      BPDIS4 PRINT B.P. #
00503A 1AA3 D6 1E87    A      INC      WORK6
00504A 1AA6 B7 4E      A      LDX      CTABL,X
00505A 1AA8 CD 1DFD    A      LDA      DTABL+5
00506A 1AAB CD 1F58    A      JSR      DISTAB
00507A 1AAE BE 51      A      JSR      BLDADR      NEW B.P.
00508A 1AB0 25 08      A      LDX      WORK2
00509A 1AB2 A1 10      1ABA     BCS      BPDIS7 YES
00510A 1AB4 27 1A      A      CMP      $10      NO,ESC?
00511A 1AB6 A1 11      1AD0     BEQ      BPRET      GET OUT
00512A 1AB8 27 0B      A      CMP      $11      ENTER?
00513A 1ABA B6 52      1AC5     BEQ      BPDIS5 GET NEXT B.P.
00514A 1ABC A1 1F      A      LDA      ADDR      TOO BIG?
00515A 1ABE 22 13      A      CMP      #PCMASK
00515A 1ABE 22 13      1AD3     BHI      BPERR      YES

```

PAGE 011 CBUG05 .SA:1

```

00516A 1AC0 F7          STA 0,X      NO,STORE NEW B.P.
00517A 1AC1 B6 53      A      LDA  ADDR1
00518A 1AC3 E7 01      A      STA  1,X
00519A 1AC5 5C          BPDIS5 INCX      GET NEXT B.P.
00520A 1AC6 5C          INCX
00521A 1AC7 5C          INCX
00522A 1AC8 BF 51      A      STX  WORK2
00523A 1ACA 3A 5A      A      DEC  PNCNT
00524A 1ACC 26 B3      1A81     BNE  BPDIS1  DONE?
00525A 1ACE 20 A8      1A78     BRA  BPDIS  YES START OVER
00526A 1AD0 CC 18AB    A BPRET JMP  GETCMD
00527          *
00528A 1AD3 CC 1E97    A BPERR JMP  ERROR
00529          *
00530          *****
00531          *
00532          *          BREAKPOINT CLEAR          *
00533          *
00534          *          TYPE # FOR SINGLE          *
00535          *          CLEAR AND ENT FOR ALL          *
00536          *
00537          *****
00538          *
00539          1AD6      A BPCLR EQU  *
00540A 1AD6 CD 1DF5    A      JSR  CLRTAB  PRINT 'BCLR'
00541A 1AD9 A6 F4      A      LDA  #$F4
00542A 1ADB B7 49      A      STA  DTABLE
00543A 1ADD A6 D1      A      LDA  #$D1
00544A 1ADF B7 4A      A      STA  DTABLE+1
00545A 1AE1 A6 D0      A      LDA  #$D0
00546A 1AE3 B7 4B      A      STA  DTABLE+2
00547A 1AE5 A6 60      A      LDA  #$60
00548A 1AE7 B7 4C      A      STA  DTABLE+3
00549A 1AE9 CD 1DFD    A      JSR  DISTAB
00550A 1AEC CD 1DD3    A      JSR  SCNBKP  FIND B.P. TABLE
00551A 1AEF BF 51      A      STX  WORK2
00552A 1AF1 CD 1F49    A      JSR  GETNYB
00553A 1AF4 25 12      1B08     BCS  BPCLR1  ENTER?
00554A 1AF6 A1 11      A      CMP  #$11
00555A 1AF8 26 34      1B2E     BNE  BPCRET  NO
00556A 1AFA A6 FF      A      LDA  $FF      YES,CLEAR ALL
00557A 1AFC BE 51      A      LDX  WORK2
00558A 1AFE F7          BPCLR2 STA 0,X
00559A 1AFF 5C          INCX
00560A 1B00 5C          INCX
00561A 1B01 5C          INCX
00562A 1B02 3A 5A      A      DEC  PNCNT
00563A 1B04 26 F8      1AFE     BNE  BPCLR2
00564A 1B06 20 26      1B2E     BRA  BPCRET
00565A 1B08 A1 03      A BPCLR1 CMP  #NUMBKP  VALID B.P. #?
00566A 1B0A 24 C7      1AD3     BHS  BPERR  NO
00567A 1B0C 97          TAX          YES
00568A 1B0D D6 1E87    A      LDA  CTABL,X  PRINT B.P. #
00569A 1B10 B7 4E      A      STA  DTABLE+5
00570A 1B12 4F          CLRA
00571A 1B13 A0 03      A      SUB  #3      FIND IT
00572A 1B15 AB 03      A BPCLR3 ADD  #3
00573A 1B17 5A          DECX

```

PAGE 012 CBUG05 .SA:1

```

00574A 1B18 2A FB      1B15      BPL      BPCLR3
00575A 1B1A B7 57      A          STA      WORK6
00576A 1B1C CD 1DFD      A          JSR      DISTAB   PRINT B.P.
00577A 1B1F CD 1E5F      A          JSR      CHRIN
00578A 1B22 A1 11      A          CMP      $11      CLEAR IT?
00579A 1B24 26 08      1B2E      BNE      BPCRET   NO
00580A 1B26 A6 37      A          LDA      #BKPTBL YES
00581A 1B28 BB 57      A          ADD      WORK6
00582A 1B2A 97          A          TAX
00583A 1B2B A6 FF      A          LDA      $FF
00584A 1B2D F7          A          STA      0,X
00585A 1B2E CC 18AB      A BPCRET JMP      GETCMD
00586
00587 *****
00588 *
00589 *          TRACE ONE INSTRUCTION *
00590 *
00591 *          TIMER INTERRUPT IS *
00592 *          USED *
00593 *
00594 *****
00595 *
00596      1B31      A TRACE EQU *
00597A 1B31 CD 1916      A          JSR      LOCSTK   FIND S.P.
00598A 1B34 E6 04      A          LDA      4,X
00599A 1B36 A4 08      A          AND      #8
00600A 1B38 B7 57      A          STA      WORK6
00601A 1B3A E6 07      A          LDA      7,X
00602A 1B3C B7 52      A          STA      ADDRH
00603A 1B3E E6 08      A          LDA      8,X
00604A 1B40 B7 53      A          STA      ADDR1
00605A 1B42 CD 1F15      A          JSR      GET OPCODE
00606A 1B45 A1 83      A          CMP      $83      SWI?
00607A 1B47 26 0F      1B58      BNE      TRACE3
00608A 1B49 B6 53      A          LDA      ADDR1   YES
00609A 1B4B AB 01      A          ADD      #1      INC PC
00610A 1B4D E7 08      A          STA      8,X
00611A 1B4F B6 52      A          LDA      ADDRH
00612A 1B51 A9 00      A          ADC      #0
00613A 1B53 E7 07      A          STA      7,X
00614A 1B55 CC 1928      A          JMP      PCOUNT
00615A 1B58 A1 9B      A TRACE3 CMP      $9B      SEI?
00616A 1B5A 26 15      1B71      BNE      TRACE2
00617A 1B5C E6 04      A          LDA      4,X      YES
00618A 1B5E AA 08      A          ORA      #8      SET IT IN
00619A 1B60 E7 04      A          STA      4,X      STACK
00620A 1B62 B6 53      A          LDA      ADDR1
00621A 1B64 AB 01      A          ADD      #1
00622A 1B66 E7 08      A          STA      8,X
00623A 1B68 B6 52      A          LDA      ADDRH
00624A 1B6A A9 00      A          ADC      #0
00625A 1B6C E7 07      A          STA      7,X
00626A 1B6E CC 1928      A          JMP      PCOUNT
00627A 1B71 A1 9A      A TRACE2 CMP      $9A      CLI?
00628A 1B73 26 02      1B77      BNE      TRACE1
00629A 1B75 3F 57      A          CLR      WORK6   YES,CLEAR IT ON STACK
00630A 1B77 E6 04      A TRACE1 LDA      4,X      GET COND. CODE
00631A 1B79 A4 F7      A          AND      $F7      CLEAR IRQ BIT

```

```

00632A 1B7B E7 04      A      STA      4,X      RETURN TO STACK
00633A 1B7D A6 10      A      LDA      #16
00634A 1B7F B7 08      A      STA      TIMER
00635A 1B81 A6 08      A      LDA      #8
00636A 1B83 B7 09      A      STA      TIMEC
00637A 1B85 80          A      RTI              EXECUTE
00638
00639
00640
00641
00642
00643
00644
00645
00646
00647
00648
00649A 1B86 CD 1DF5      A      STIME EQU      *
00650A 1B89 A6 77      A      JSR      CLRTAB
00651A 1B8B B7 4E      A      LDA      #$77      AM BY DEFAULT
00652A 1B8D 3F 53      A      STA      DTABL+5
00653A 1B8F 3F 52      A      CLR      ADDR1
00654A 1B91 CD 1FB0      A      CLR      ADDR2
00655A 1B94 CD 1F49      A      STIME2 JSR      PRTADR
00656A 1B97 25 12      A      JSR      GETNYB      GET INPUT
00657A 1B99 A1 10      A      BCS      STIME1
00658A 1B9B 27 4F      A      CMP      #$10      ESC?
00659A 1B9D A1 11      A      BEQ      STMRET
00660A 1B9F 27 1D      A      CMP      #$11      ENT?
00661A 1BA1 A1 17      A      BEQ      STIME4
00662A 1BA3 26 EC      A      CMP      #$17      P?
00663A 1BA5 A6 73      A      BNE      STIME2
00664A 1BA7 B7 4E      A      LDA      #$73      YES,
00665A 1BA9 20 E6      A      STA      DTABL+5      PRINT P
00666A 1BAB A1 09      A      BRA      STIME2
00667A 1BAD 22 40      A      STIME1 CMP      #9      GT 9?
00668A 1BAF AE 04      A      BHI      STERR
00669A 1BB1 38 53      A      LDX      #4      SHIFT IN NEW
00670A 1BB3 39 52      A      STIME3 LSL      ADDR1      INPUT
00671A 1BB5 5A          A      ROL      ADDR2
00672A 1BB6 26 F9      A      DECX
00673A 1BB8 BA 53      A      BNE      STIME3
00674A 1BBA B7 53      A      ORA      ADDR1
00675A 1BBC 20 D3      A      STA      ADDR1
00676A 1BBE B6 52      A      BRA      STIME2
00677A 1BC0 A1 12      A      STIME4 LDA      ADDR2      HOURS GT 12?
00678A 1BC2 22 2B      A      CMP      #$12
00679A 1BC4 4D          A      BHI      STERR
00680A 1BC5 27 28      A      TSTA
00681A 1BC7 B6 53      A      BEQ      STERR      HOURS EQ 0?
00682A 1BC9 A1 59      A      LDA      ADDR1      MIN? GT 59?
00683A 1BCB 22 22      A      CMP      #$59
00684A 1BCD A6 80      A      BHI      STERR
00685A 1BCF C7 170B      A      LDA      #$80      PUT IN
00686A 1BD2 4F          A      STA      CR2      SET TIME MODE
00687A 1BD3 C7 170A      A      CLRA
00688A 1BD6 04 4E 02 1BDB A      STA      CR1
00689A 1BD9 1E 52      A      BRSET 2,DTABL+5,STIME5 PM?
                                BSET 7,ADDRH YES

```

PAGE 014 CBUG05 .SA:1

```

00690A 1BDB B6 53      A STIME5 LDA      ADDR1      PUT TIME INTO
00691A 1BDD C7 1702    A          STA      MIN        MC146818
00692A 1BE0 B6 52      A          LDA      ADDRHH
00693A 1BE2 C7 1704    A          STA      HOUR
00694A 1BE5 4F         A          CLRA
00695A 1BE6 C7 170B    A          STA      CR2        ALLOW TO RUN
00696A 1BE9 C7 1700    A          STA      SEC        CLR SECONDS
00697A 1BEC CC 18AB    A STMRET JMP      GETCMD
00698                      *
00699A 1BEF CC 1E97    A STERR  JMP      ERROR
00700                      *
00701                      *****
00702                      *
00703                      *          WAIT FOR THE END          *
00704                      *          OF UPDATE CYCLE          *
00705                      *
00706                      *****
00707                      *
00708A 1BF2 CD 1E23    A VALID  JSR      KEYSCN
00709A 1BF5 25 13      1C0A      BCS      VALRET
00710A 1BF7 C6 170A    A          LDA      CR1        IS UIP LOW?
00711A 1BFA A4 80      A          AND      #$80
00712A 1BFC 27 F4      1BF2      BEQ      VALID      YES, WAIT UNTIL HIGH
00713A 1BFE CD 1E23    A VALID2 JSR      KEYSCN
00714A 1C01 25 07      1C0A      BCS      VALRET
00715A 1C03 C6 170A    A          LDA      CR1        UIP MADE NEG TRANSITION
00716A 1C06 A4 80      A          AND      #$80
00717A 1C08 26 F4      1BFE      BNE      VALID2
00718A 1C0A 81         VALRET RTS
00719                      *
00720                      *****
00721                      *
00722                      *          DISPLAY CURRENT TIME          *
00723                      *          FROM MC146818          *
00724                      *
00725                      *          USES 12-HOUR FORMAT          *
00726                      *
00727                      *****
00728                      *
00729                      1C0B      A DTIME  EQU      *
00730A 1C0B CD 1DF5    A          JSR      CLRTAB
00731A 1C0E A6 77      A          LDA      #$77
00732A 1C10 B7 4E      A          STA      DTABL+5
00733A 1C12 AD DE      1BF2      BSR      VALID      UPDATE OVER
00734A 1C14 24 04      1C1A      BCC      DTIME2
00735A 1C16 5F         CLRX
00736A 1C17 CC 18BD    A          JMP      RJUMP
00737A 1C1A C6 1704    A DTIME2 LDA      HOUR
00738A 1C1D B7 52      A          STA      ADDRHH
00739A 1C1F 0F 52 06 1C28 BRCLR   7, ADDRHH, DTIME1 PM?
00740A 1C22 1F 52      A          BCLR   7, ADDRHH
00741A 1C24 A6 73      A          LDA      #$73      PRINT IT
00742A 1C26 B7 4E      A          STA      DTABL+5
00743A 1C28 C6 1702    A DTIME1 LDA      MIN
00744A 1C2B B7 53      A          STA      ADDR1
00745A 1C2D CD 1FB0    A          JSR      PRTADR      PRINT TIME
00746A 1C30 20 D9      1C0B      BRA      DTIME
00747                      *

```

```

00748 *****
00749 *
00750 *
00751 * PUNCH TAPE *
00752 *
00753 * LAST ADDRESS WILL *
00754 * REMAIN UNTIL PUNCH *
00755 * IS COMPLETE *
00756 *
00757 * 2400 BAUD IS DEFAULT *
00758 *****
00759 *
00760A 1C32 CC 1E97 A PUNERR JMP ERROR
00761 *
00762 1C35 A PUNCH EQU *
00763A 1C35 CD 1998 A JSR BLDRNG BUILD RANGE
00764A 1C38 08 4F 49 1C84 A BRSET 4,SWIFLG,PUNRET VALID?
00765A 1C3B 06 4F F4 1C32 A BRSET 3,SWIFLG,PUNERR VAILD?
00766A 1C3E BE 58 A LDX TEMP SWAP ADDRESSES
00767A 1C40 B7 58 A STA TEMP
00768A 1C42 BF 52 A STX ADDRH
00769A 1C44 B6 53 A LDA ADDRRL
00770A 1C46 BE 59 A LDX TEMP+1
00771A 1C48 BF 53 A STX ADDRRL
00772A 1C4A 4C A INCA
00773A 1C4B 26 02 1C4F A BNE PUN3 ADJUST
00774A 1C4D 3C 58 A INC TEMP ENDING
00775A 1C4F B7 59 A PUN3 STA TEMP+1 ADDRESS
00776A 1C51 AD 3F 1C92 A BSR PUNLDR
00777A 1C53 A6 B3 A LDA #B3 PUNCH LEADER
00778A 1C55 AD 50 1CA7 A BSR PUNBYT PUNCH BOT
00779A 1C57 3F 5B A CLR CHKSUM
00780A 1C59 B6 58 A LDA TEMP INITIALIZE CHECKSUM
00781A 1C5B AD 2A 1C87 A BSR PUNIT PUNCH
00782A 1C5D B6 59 A LDA TEMP+1 ENDING ADDRESS
00783A 1C5F AD 26 1C87 A BSR PUNIT
00784A 1C61 B6 52 A LDA ADDRH PUNCH
00785A 1C63 AD 22 1C87 A BSR PUNIT BEGINNING ADDRESS
00786A 1C65 B6 53 A LDA ADDRRL
00787A 1C67 AD 1E 1C87 A BSR PUNIT
00788A 1C69 CD 1F15 A PUN5 JSR LOAD GET BYTE FROM MEMORY
00789A 1C6C AD 19 1C87 A BSR PUNIT PUNCH IT
00790A 1C6E 3C 53 A INC ADDRRL
00791A 1C70 26 02 1C74 A BNE PUN4
00792A 1C72 3C 52 A INC ADDRH
00793A 1C74 B6 58 A PUN4 LDA TEMP FINISHED?
00794A 1C76 B1 52 A CMP ADDRH
00795A 1C78 26 EF 1C69 A BNE PUN5
00796A 1C7A B6 59 A LDA TEMP+1
00797A 1C7C B1 53 A CMP ADDRRL
00798A 1C7E 26 E9 1C69 A BNE PUN5
00799A 1C80 B6 5B A LDA CHKSUM YES, PUNCH
00800A 1C82 AD 23 1CA7 A BSR PUNBYT CHECKSUM
00801A 1C84 CC 18AB A PUNRET JMP GETCMD
00802 *
00803A 1C87 B7 56 A PUNIT STA WORK5
00804A 1C89 AD 1C 1CA7 A BSR PUNBYT PUNCH BYTE
00805A 1C8B B6 56 A LDA WORK5 AND UPDATE

```

PAGE 016 CBUG05 .SA:1

00806A	1C8D	BB	5B	A	ADD	CHKSUM	CHECKSUM
00807A	1C8F	B7	5B	A	STA	CHKSUM	
00808A	1C91	81			RTS		
00809				*			
00810A	1C92	A6	3F	A	PUNLDR	LDA	#S3F PUNCH 16K
00811A	1C94	B7	50	A	STA	WORK1	ZEROS
00812A	1C96	A6	FF	A	LDA	#SFF	
00813A	1C98	B7	51	A	STA	WORK2	
00814A	1C9A	AD	24	1CC0	PUNLD1	BSR	COMO
00815A	1C9C	AD	35	1CD3	BSR	NOCO	
00816A	1C9E	3A	51	A	DEC	WORK2	
00817A	1CA0	26	F8	1C9A	BNE	PUNLD1	
00818A	1CA2	3A	50	A	DEC	WORK1	
00819A	1CA4	26	F4	1C9A	BNE	PUNLD1	
00820A	1CA6	81			RTS		
00821				*			
00822A	1CA7	AE	08	A	PUNBYT	LDX	#8 PUNCH
00823A	1CA9	AD	15	1CC0	BSR	COMO	SYNC
00824A	1CAB	AD	13	1CC0	BSR	COMO	START
00825A	1CAD	AD	11	1CC0	PUNBY1	BSR	COMO SYNC
00826A	1CAF	46			RORA		
00827A	1CB0	24	04	1CB6	BCC	PUNBY2	1 OR 0?
00828A	1CB2	AD	0C	1CC0	BSR	COMO	1
00829A	1CB4	20	02	1CB8	BRA	PUNBY3	
00830A	1CB6	AD	1B	1CD3	PUNBY2	BSR	NOCO
00831A	1CB8	5A			PUNBY3	DECX	0 ALL
00832A	1CB9	26	F2	1CAD	BNE	PUNBY1	DONE?
00833A	1CBB	AD	03	1CC0	BSR	COMO	YES, SYNC
00834A	1CBD	AD	14	1CD3	BSR	NOCO	STOP BIT
00835A	1CBF	81			RTS		
00836				*			
00837A	1CC0	BF	54	A	COMO	STX	WORK3 MAKE A TRANSITION
00838A	1CC2	0D	00 04	1CC9	BRCLR	6, PORTA, COM01	
00839A	1CC5	1D	00	A	BCLR	6, PORTA	
00840A	1CC7	20	02	1CCB	BRA	DELAY	PAUSE
00841A	1CC9	1C	00	A	COM01	BSET	6, PORTA
00842A	1CCB	BE	5E	A	DELAY	LDX	PCNT1
00843A	1CCD	5A			COM02	DECX	
00844A	1CCE	26	FD	1CCD	BNE	COM02	
00845A	1CD0	BE	54	A	LDX	WORK3	
00846A	1CD2	81			RTS		
00847				*			
00848A	1CD3	BF	54	A	NOCO	STX	WORK3 NO TRANSITION
00849A	1CD5	BE	5F	A	LDX	PCNT0	DOUBLE DELAY
00850A	1CD7	5A			NOC01	DECX	
00851A	1CD8	26	FD	1CD7	BNE	NOC01	
00852A	1CDA	BE	54	A	LDX	WORK3	
00853A	1CDC	81			RTS		
00854				*			
00855				*****			
00856				*			
00857				*		LOAD TAPE OR	
00858				*		COMPARE TAPE	
00859				*			
00860				*****			
00861				*			
00862		1CDD		A	TLOAD	EQU	*
00863A	1CDD	1B	4F	A	BCLR	5, SWIFLG	

PAGE 017 CBUG05 .SA:1

00864A	1CDF	CD	1E07	A	JSR	CLRDIS	
00865A	1CE2	A6	FF	A	LOAD0	LDA	#\$FF
00866A	1CE4	AD	78	1D5E	LOAD1	BSR	EDGE
00867A	1CE6	25	FA	1CE2		BCS	LOAD0
00868A	1CE8	4A				DECA	ZEROS
00869A	1CE9	26	F9	1CE4		BNE	LOAD1
00870A	1CEB	AD	50	1D3D	LOAD2	BSR	LOADBY
00871A	1CED	A1	B3	A		CMP	#\$B3
00872A	1CEF	26	FA	1CEB		BNE	LOAD2
00873				*			
00874A	1CF1	3F	5B	A	CLR	CHKSUM	YES, INIT CHECKSUM
00875A	1CF3	CD	1D76	A	JSR	LOADIT	GET ENDING
00876A	1CF6	B7	58	A	STA	TEMP	ADDRESS
00877A	1CF8	AD	7C	1D76	BSR	LOADIT	
00878A	1CFA	B7	59	A	STA	TEMP+1	
00879A	1CFC	AD	78	1D76	BSR	LOADIT	GET BEGINNING
00880A	1CFE	B7	52	A	STA	ADDRH	ADDRESS
00881A	1D00	AD	74	1D76	BSR	LOADIT	
00882A	1D02	B7	53	A	STA	ADDRL	
00883				*			
00884A	1D04	AD	70	1D76	LOAD4	BSR	LOADIT
00885A	1D06	0B	4F	0B 1D14	BRCLR	5,SWIFLG,	GET BYTE
00886A	1D09	B7	57	A	STA	WORK6	LOAD5 COMPARE?
00887A	1D0B	CD	1F15	A	JSR	LOAD	YES, IS IT
00888A	1D0E	B1	57	A	CMP	WORK6	SAME?
00889A	1D10	26	25	1D37	BNE	DISADR	NO
00890A	1D12	20	05	1D19	BRA	LOAD6	YES
00891A	1D14	CD	1F24	A	LOAD5	JSR	STORE
00892A	1D17	25	1E	1D37	BCS	DISADR	NOT COMPARE, SAVE IT
00893A	1D19	3C	53	A	LOAD6	INC	INC ADDRESS
00894A	1D1B	26	02	1D1F	BNE	LOAD3	
00895A	1D1D	3C	52	A	INC	ADDRH	
00896A	1D1F	B6	58	A	LOAD3	LDA	FINSHED?
00897A	1D21	B1	52	A	CMP	ADDRH	
00898A	1D23	26	DF	1D04	BNE	LOAD4	
00899A	1D25	B6	59	A	LDA	TEMP+1	
00900A	1D27	B1	53	A	CMP	ADDRL	
00901A	1D29	26	D9	1D04	BNE	LOAD4	
00902A	1D2B	AD	10	1D3D	BSR	LOADBY	YES, GET
00903A	1D2D	B1	5B	A	CMP	CHKSUM	CHECKSUM
00904A	1D2F	26	03	1D34	BNE	LDERR	NOT SAME -- ERROR
00905A	1D31	CC	18AB	A	JMP	GETCMD	
00906				*			
00907A	1D34	CC	1E97	A	LDERR	JMP	ERROR
00908				*			
00909A	1D37	CD	1FB0	A	DISADR	JSR	PRTADR
00910A	1D3A	CC	18B5	A		JMP	CMDSCN
00911				*			DISPLAY ADDRESS
00912A	1D3D	BF	50	A	LOADBY	STX	FOR ERROR
00913A	1D3F	AE	08	A		LDX	WORK1
00914A	1D41	AD	1B	1D5E		BSR	#8
00915A	1D43	AD	19	1D5E	LODBY1	BSR	EDGE
00916A	1D45	24	FC	1D43		BCC	SET START
00917A	1D47	AD	15	1D5E		BSR	LODBY1
00918A	1D49	5A			LODBY2	DECX	EDGE
00919A	1D4A	2B	0F	1D5B		BMI	LODBYR
00920A	1D4C	44				LSRA	FINISHED?
00921A	1D4D	AD	0F	1D5E		BSR	NO, SHIFT
							GET BIT

PAGE 018 CBUG05 .SA:1

```

00922A 1D4F 24 06      1D57      BCC      LODBY3      1 OR 0?
00923A 1D51 AD 0B      1D5E      BSR      EDGE      IF 1 GET CLEAR NEXT
00924A 1D53 AA 80      A          ORA      #$80      TRANSITION
00925A 1D55 20 F2      1D49      BRA      LODBY2      SHIFT IN 1
00926A 1D57 AA 00      A LODBY3      ORA      #0        SHIFT IN 0
00927A 1D59 20 EE      1D49      BRA      LODBY2
00928A 1D5B BE 50      A LODBYR      LDX      WORK1
00929A 1D5D 81          A          RTS
00930
00931A 1D5E B7 51      A EDGE      STA      WORK2
00932A 1D60 BF 54      A          STX      WORK3
00933A 1D62 5F          A          CLRX
00934A 1D63 5C          EDGE1      INCX
00935A 1D64 4F          A          CLRA
00936A 1D65 2E 01      1D68      BIL      EDGE2
00937A 1D67 4C          A          INCA
00938A 1D68 B1 5C      A EDGE2      CMP      SREF
00939A 1D6A 27 F7      1D63      BEQ      EDGE1
00940A 1D6C B7 5C      A          STA      SREF      UPDATE LEVEL
00941A 1D6E 9F          A          TXA      STATUS
00942A 1D6F B0 5D      A          SUB      LCNT      SET CARRY FOR
00943A 1D71 B6 51      A          LDA      WORK2      1 OR 0
00944A 1D73 BE 54      A          LDX      WORK3
00945A 1D75 81          A          RTS
00946
00947A 1D76 AD C5      1D3D      LOADIT      BSR      LOADBY      GET BYTE
00948A 1D78 B7 55      A          STA      WORK4      AND UPDATE
00949A 1D7A BB 5B      A          ADD      CHKSUM      CHECKSUM
00950A 1D7C B7 5B      A          STA      CHKSUM
00951A 1D7E B6 55      A          LDA      WORK4
00952A 1D80 81          A          RTS
00953
00954      *
00955      *****
00956      *
00957      *          VERIFY TAPE
00958      *
00959      *****
00960      *
00961A 1D81 1A 4F      A VERIFY      EQU      *
00962A 1D83 CD 1E07      A          BSET      5,SWIFLG
00963A 1D86 CC 1CE2      A          JSR      CLRDIS
00964      A          JMP      LOAD0
00965A 1D89 CC 1E97      A GOERR      JMP      ERROR
00966      *
00967A 1D8C CC 18AB      A GOBACK      JMP      GETCMD
00968      *
00969      1D8F      A GO      EQU      *
00970A 1D8F CD 1916      A          JSR      LOCSTK
00971A 1D92 E6 08      A          LDA      8,X
00972A 1D94 B7 53      A          STA      ADDR1
00973A 1D96 E6 07      A          LDA      7,X
00974A 1D98 B7 52      A          STA      ADDRH
00975A 1D9A CD 1F53      A          JSR      GETADR
00976A 1D9D 25 08      1DA7      BCS      GOON      ADDR VALID?
00977A 1D9F A1 10      A          CMP      $10
00978A 1DA1 27 E9      1D8C      BEQ      GOBACK
00979A 1DA3 A1 11      A          CMP      $11

```

PAGE 019 CBUG05 .SA:1

```

00980A 1DA5 26 E2      1D89      BNE      GOERR
00981A 1DA7 CD 1916     A GOON     JSR      LOCSTK  YES PUT IT
00982A 1DAA B6 52      A          LDA      ADDRSH  IN STACK
00983A 1DAC A1 1F      A          CMP      #PCMASK TO BIG?
00984A 1DAE 22 D9      1D89      BHI      GOERR  YES
00985A 1DB0 E7 07      A          STA      7,X
00986A 1DB2 B6 53      A          LDA      ADDRDL
00987A 1DB4 E7 08      A          STA      8,X
00988A 1DB6 AD 1B      1DD3 CONT BSR      SCNBKP  FIND B.P. TABLE
00989A 1DB8 F6          GOINSB LDA      0,X      INSERTPB.P.'S
00990A 1DB9 2B 10      1DCB      BMI      GONOB  VALID?
00991A 1DBB B7 52      A          STA      ADDRSH  YES
00992A 1DBD E6 01      A          LDA      1,X
00993A 1DBF B7 53      A          STA      ADDRDL
00994A 1DC1 CD 1F15     A          JSR      LOAD      SAVE OPCODE
00995A 1DC4 E7 02      A          STA      2,X
00996A 1DC6 A6 83      A          LDA      #SWIOP
00997A 1DC8 CD 1F24     A          JSR      STORE
00998A 1DCB 5C          GONOB     INCX
00999A 1DCC 5C          INCX
01000A 1DCD 5C          INCX
01001A 1DCE 3A 5A      A          DEC      PNCNT
01002A 1DD0 26 E6      1DB8      BNE      GOINSB  DONE?
01003A 1DD2 80          RTI      YES
01004
01005      1DD3      A SCNBKP EQU      *
01006A 1DD3 A6 03      A          LDA      #NUMBKP
01007A 1DD5 B7 5A      A          STA      PNCNT
01008A 1DD7 AE 37      A          LDX      #BKPTBL
01009A 1DD9 81          RTS
01010
01011      *****
01012      *
01013      *      DISPLAY STACK POINTER      *
01014      *
01015      *****
01016      *
01017      1DDA      A STACK EQU      *
01018A 1DDA A6 B5      A          LDA      #$B5  PRINT
01019A 1DDC B7 4D      A          STA      DTABL+4  'SP'
01020A 1DDE A6 73      A          LDA      #$73
01021A 1DE0 B7 4E      A          STA      DTABL+5
01022A 1DE2 4F          CLRA
01023A 1DE3 5F          CLRX
01024A 1DE4 CD 1F8E     A          JSR      PRTBYT
01025A 1DE7 CD 1916     A          JSR      LOCSTK  FIND USER
01026A 1DEA 9F          TXA      STACK POINTER
01027A 1DEB AB 03      A          ADD      #3
01028A 1DED AE 02      A          LDX      #2
01029A 1DEF CD 1F8E     A          JSR      PRTBYT  PRINT IT
01030A 1DF2 CC 18B5     A          JMP      CMDSCN
01031      *

```

```

01033          *
01034          *****
01035          *
01036          *          CLEAR DISPLAY TABLE          *
01037          *
01038          *          X REG DESTROYED          *
01039          *
01040          *****
01041          *
01042A 1DF5 AE 05      A CLRTAB LDX      #5
01043A 1DF7 6F 49      A CLRLOC CLR      DTABL,X    CLEAR SIX
01044A 1DF9 5A          DECX          LOCATIONS IN
01045A 1DFA 2A FB      1DF7      BPL      CLRLOC    DISPLAY TABLE
01046A 1DFC 81          RTS
01047          *
01048          *****
01049          *
01050          *          DISPLAY TABLE CONTENTS          *
01051          *
01052          *          A,X REGISTERS DESTROYED          *
01053          *
01054          *****
01055          *
01056A 1DFD AE 05      A DISTAB LDX      #5
01057A 1DFF E6 49      A DISCHR LDA      DTABL,X    LOAD DISPLAY
01058A 1E01 AD 09      1E0C      BSR      DISPLY    TABLE INTO
01059A 1E03 5A          DECX          145000
01060A 1E04 2A F9      1DFF      BPL      DISCHR
01061A 1E06 81          RTS
01062          *
01063          *****
01064          *
01065          *          BLANK DISPLAY          *
01066          *
01067          *          A,X REGISTERS DESTROYED          *
01068          *
01069          *****
01070          *
01071A 1E07 AD EC      1DF5 CLRDIS BSR      CLRTAB    BLANK
01072A 1E09 AD F2      1DFD      BSR      DISTAB    DISPLAY
01073A 1E0B 81          RTS
01074          *
01075          *****
01076          *
01077          *          SHIFT ONE CHARACTER INTO          *
01078          *          DISPLAY          *
01079          *
01080          *          A REGISTER DESTROYED          *
01081          *
01082          *****
01083          *
01084A 1E0C BF 50      A DISPLY STX      WORK1      SAVE INDEX
01085A 1E0E 1D 00      A          BCLR      6,PORTA  CLEAR DATA
01086A 1E10 AE 08      A          LDX      #8
01087A 1E12 48          DIS1      LSLA          SET UP
01088A 1E13 24 02      1E17      BCC      DIS2      BIT OF
01089A 1E15 1C 00      A          BSET      6,PORTA  ACCUMULATOR
01090A 1E17 1E 00      A DIS2      BSET      7,PORTA  CLOCK

```

PAGE 021 CBUG05 .SA:1

```

01091A 1E19 1F 00      A      BCLR    7,PORTA  IT
01092A 1E1B 1D 00      A      BCLR    6,PORTA  CLEAR DATA
01093A 1E1D 5A          A      DECX          COMPLETE?
01094A 1E1E 26 F2      1E12    BNE      DIS1    NO
01095A 1E20 BE 50      A      LDX      WORK1   RESTORE INDEX
01096A 1E22 81          A      RTS
01097                      *
01098                      *****
01099                      *
01100                      *      KEYPAD SCAN      *
01101                      *      *
01102                      *      X REGISTER DESTROYED      *
01103                      *      *
01104                      *      A REGISTER CONTAINS VALUE      *
01105                      *      *
01106                      *      CARRY SET IF VALID OUTPUT      *
01107                      *
01108                      *****
01109                      *
01110A 1E23 98          A      KEYSCN  CLC
01111A 1E24 4F          A      CLRA
01112A 1E25 AE 06      A      LDX      #6      SETUP
01113A 1E27 AB 10      A KEY1  ADD      #$10    ROW
01114A 1E29 B7 00      A      STA      PORTA
01115A 1E2B AD 06      1E33    BSR      COLUMN  CHECK COLUMNS
01116A 1E2D 25 03      1E32    BCS      KEY2    IF VALID GET OUT
01117A 1E2F 5A          A      DECX          ELSE TRY
01118A 1E30 26 F5      1E27    BNE      KEY1    NEXT ROW
01119A 1E32 81          A      KEY2  RTS
01120                      *
01121                      *****
01122                      *
01123                      *      CHECK FOR KEY CLOSURE      *
01124                      *      WITHIN COLUMN AND DEBOUNCE      *
01125                      *      *
01126                      *      A REGISTER CONTAINS VALUE      *
01127                      *      *
01128                      *      CARRY SET IF VALID OUTPUT      *
01129                      *
01130                      *****
01131                      *
01132A 1E33 B6 00      A COLUMN LDA      PORTA  READ KEYPAD
01133A 1E35 B7 50      A      STA      WORK1  STORE IT
01134A 1E37 A5 0F      A      BIT      #$0F   KEY CLOSED?
01135A 1E39 27 19      1E54    BEQ      COLRET NO GET OUT
01136A 1E3B AD 18      1E55    BSR      DBOUNC  ELSE DEBOUNCE
01137A 1E3D B6 00      A      LDA      PORTA  RE-READ KEYPAD
01138A 1E3F B1 50      A      CMP      WORK1  SAME KEY CLOSED?
01139A 1E41 26 11      1E54    BNE      COLRET NO GET OUT
01140A 1E43 99          A      SEC          SET FLAG FOR VALID
01141A 1E44 B6 00      A COL1  LDA      PORTA  KEY
01142A 1E46 A5 0F      A      BIT      #$0F   RELEASED?
01143A 1E48 26 FA      1E44    BNE      COL1  NO TRY AGAIN
01144A 1E4A AD 09      1E55    BSR      DBOUNC  YES DEBOUNCE
01145A 1E4C B6 00      A      LDA      PORTA  STILL
01146A 1E4E A5 0F      A      BIT      #$0F   RELEASED?
01147A 1E50 26 F2      1E44    BNE      COL1  NO TRY AGAIN
01148A 1E52 B6 50      A      LDA      WORK1  RETURN CHAR IN A-REG

```

PAGE 022 CBUG05 .SA:1

```

01149A 1E54 81      COLRET RTS      YES GO HOME
01150                *
01151                *****
01152                *
01153                *      PAUSE FOR 3075 CYCLES      *
01154                *
01155                *      A REGISTER DESTROYED      *
01156                *
01157                *****
01158                *
01159A 1E55 A6 FF      A DBOUNC LDA    #$FF      PAUSE
01160A 1E57 21 FE      1E57 DLOOP BRN    *      256X12
01161A 1E59 21 FE      1E59          BRN    *      CYCLES
01162A 1E5B 4A          DECA          OR AT
01163A 1E5C 26 F9      1E57          BNE    DLOOP    LEAST
01164A 1E5E 81          RTS          3.7 MS
01165                *

```

```

01167      *
01168      *****
01169      *
01170      *          INPUT ONE CHARACTER          *
01171      *
01172      *          A REGISTER CONTAINS HEX VALUE      *
01173      *
01174      *          X REGISTER CONTAINS HEX VALUE      *
01175      *
01176      *****
01177      *
01178      1E5F      A CHRIN  EQU      *
01179A 1E5F CD 1E23      A      JSR      KEYSCN      GET KEY
01180A 1E62 24 FB      1E5F      BCC      CHRIN      IF NOT VALID RETRY
01181A 1E64 5F      CLRX
01182A 1E65 D1 1E6F      A CHRIN1 CMP      STABL,X      CONVERT
01183A 1E68 27 03      1E6D      BEQ      CHRIN2      TO HEX
01184A 1E6A 5C      INCX
01185A 1E6B 20 F8      1E65      BRA      CHRIN1
01186A 1E6D 9F      CHRIN2 TXA      IF CANCEL
01187A 1E6E 81      RTS
01188      *
01189      *****
01190      *
01191      *          CONVERSION TABLE FOR KEYPAD      *
01192      *          TO HEX NUMBER                      *
01193      *
01194      *****
01195      *
01196A 1E6F      11      A STABL  FCB      $11      0
01197A 1E70      21      A      FCB      $21      1
01198A 1E71      22      A      FCB      $22      2
01199A 1E72      24      A      FCB      $24      3
01200A 1E73      31      A      FCB      $31      4
01201A 1E74      32      A      FCB      $32      5
01202A 1E75      34      A      FCB      $34      6
01203A 1E76      41      A      FCB      $41      7
01204A 1E77      42      A      FCB      $42      8
01205A 1E78      44      A      FCB      $44      9
01206A 1E79      48      A      FCB      $48      A
01207A 1E7A      38      A      FCB      $38      B
01208A 1E7B      28      A      FCB      $28      C
01209A 1E7C      18      A      FCB      $18      D
01210A 1E7D      14      A      FCB      $14      E
01211A 1E7E      12      A      FCB      $12      F
01212A 1E7F      61      A      FCB      $61      CANCEL COMMAND
01213A 1E80      58      A      FCB      $58      ENTER COMMAND
01214A 1E81      68      A      FCB      $68      STACK POINTER
01215A 1E82      64      A      FCB      $64      MEMORY
01216A 1E83      62      A      FCB      $62      GO
01217A 1E84      54      A      FCB      $54      VERIFY TAPE
01218A 1E85      52      A      FCB      $52      LOAD TAPE
01219A 1E86      51      A      FCB      $51      PUNCH TAPE
01220      *
01221      *****
01222      *
01223      *          HEX TO MUX DISPLAY                *
01224      *          CONVERSION TABLE                *

```

PAGE 024 CBUG05 .SA:1

```

01225
01226
01227
01228A 1E87 D7 A CTABL FCB $D7 0
01229A 1E88 06 A FCB 6 1
01230A 1E89 E3 A FCB $E3 2
01231A 1E8A A7 A FCB $A7 3
01232A 1E8B 36 A FCB $36 4
01233A 1E8C B5 A FCB $B5 5
01234A 1E8D F5 A FCB $F5 6
01235A 1E8E 07 A FCB 7 7
01236A 1E8F F7 A FCB $F7 8
01237A 1E90 B7 A FCB $B7 9
01238A 1E91 77 A FCB $77 A
01239A 1E92 F4 A FCB $F4 B
01240A 1E93 D1 A FCB $D1 C
01241A 1E94 E6 A FCB $E6 D
01242A 1E95 F1 A FCB $F1 E
01243A 1E96 71 A FCB $71 F
01244
01245 1E97 A ERROR EQU *
01246A 1E97 CD 1DF5 A JSR CLRTAB
01247A 1E9A A6 F1 A LDA #$F1
01248A 1E9C B7 4A A STA DTABL+1
01249A 1E9E A6 60 A LDA #$60
01250A 1EA0 B7 4B A STA DTABL+2
01251A 1EA2 B7 4C A STA DTABL+3
01252A 1EA4 CD 1DFD A JSR DISTAB
01253A 1EA7 CC 18B5 A JMP CMDSCN

```

```

01255
01256
01257
01258
01259
01260
01261
01262A 1EAA CD 1F53      A MEMEX JSR GETADR BUILD ADDRESS
01263A 1EAD A1 10        A          CMP    #$10
01264A 1EAF 27 5F        1F10        BEQ    MEMEX4
01265A 1EB1 B7 50        A MEMEX3 STA WORK1
01266A 1EB3 B6 52        A          LDA ADDRH
01267A 1EB5 A1 1F        A          CMP    #PCMASK
01268A 1EB7 23 03        1EBC        BLS    MEMOK
01269A 1EB9 CC 1E97      A          JMP    ERROR
01270A 1EBC B6 50        A MEMOK LDA WORK1
01271A 1EBE CD 1F15      A          JSR    LOAD LOAD DATA
01272A 1EC1 CD 1F8C      A          JSR    PRDAT PRINT IT
01273A 1EC4 CD 1F49      A          JSR    GETNYB GET NEW NIBBLE
01274A 1EC7 A1 10        A          CMP    #$10
01275A 1EC9 27 45        1F10        BEQ    MEMEX4
01276A 1ECB A1 11        A          CMP    #$11
01277A 1ECD 27 19        1EE8        BEQ    ADRINC
01278A 1ECF A1 13        A          CMP    #$13
01279A 1ED1 27 2D        1F00        BEQ    ADRDEC
01280A 1ED3 24 08        1EDD        BCC    CMDMDL IF VALID
01281A 1ED5 CD 1F8C      A MEMEX1 JSR PRDAT PRINT IT
01282A 1ED8 CD 1F37      A          JSR    GETBY2 SHIFT IN NEXT
01283A 1EDB 25 F8        1ED5        BCS    MEMEX1 IF VALID TRY AGAIN
01284
01285A 1EDD A1 11        A CMDMDL CMP    #$11 ENTER?
01286A 1EDF 26 15        1EF6        BNE    MEMEX2 NO
01287A 1EE1 B6 51        A          LDA WORK2 RESTORE ACCA
01288A 1EE3 CD 1F24      A          JSR    STORE YES STORE IT
01289A 1EE6 25 C9        1EB1        BCS    MEMEX3 STORE VALID?
01290A 1EE8 0C 4F 25    1F10 ADRINC BRSET 6,SWIFLG,MEMEX4
01291A 1EEB 3C 53        A          INC    ADDR1 YES GOTTO
01292A 1EED 26 02        1EF1        BNE    MEMEX5 NEXT
01293A 1EEF 3C 52        A          INC    ADDRH
01294A 1EF1 CD 1FB0      A MEMEX5 JSR PRDAT PRINT IT
01295A 1EF4 20 BB        1EB1        BRA    MEMEX3 REPEAT
01296A 1EF6 A1 13        A MEMEX2 CMP    #$13 MEMORY?
01297A 1EF8 26 16        1F10        BNE    MEMEX4 NO
01298A 1EFA B6 51        A          LDA WORK2
01299A 1EFC AD 26        1F24        BSR    STORE
01300A 1EFE 25 B1        1EB1        BCS    MEMEX3
01301A 1F00 0C 4F 0D    1F10 ADRDEC BRSET 6,SWIFLG,MEMEX4
01302A 1F03 3D 53        A          ADDRL TST YES THEN
01303A 1F05 26 02        1F09        BNE    CMDMB2 GET PREVIOUS
01304A 1F07 3A 52        A          DEC    ADDRH ADDRESS
01305A 1F09 3A 53        A CMDMB2 DEC ADDR1
01306A 1F0B CD 1FB0      A          JSR    PRDAT PRINT IT
01307A 1F0E 20 A1        1EB1        BRA    MEMEX3 REPEAT
01308A 1F10 1D 4F        A MEMEX4 BCLR 6,SWIFLG INVALID CHAR
01309A 1F12 CC 18AB      A          JMP    GETCMD
01310
01311
01312

```

```

01313          *      LOAD BYTE AT ADDRH,ADDRL          *
01314          *      INTO ACCUMULATOR                  *
01315          *                                           *
01316          *****
01317          *
01318A 1F15 BF 50      A LOAD      STX      WORK1      SETUP
01319A 1F17 AE C6      A          LDX      $C6        ROUTINE
01320A 1F19 BF 51      A LDSTCM  STX      WORK2      TO DO
01321A 1F1B AE 81      A          LDX      $81        TWO BYTE
01322A 1F1D BF 54      A          STX      WORK3      LOAD
01323A 1F1F BD 51      A          JSR      WORK2
01324A 1F21 BE 50      A          LDX      WORK1
01325A 1F23 81          *      RTS
01326          *
01327          *****
01328          *
01329          *      STORE ACCUMULATOR INTO          *
01330          *      BYTE AT ADDRH,ADDRL              *
01331          *                                           *
01332          *****
01333          *
01334A 1F24 BF 50      A STORE    STX      WORK1
01335A 1F26 AE C7      A          LDX      $C7        SETUP
01336A 1F28 AD EF      1F19      BSR      LDSTCM      ROUTINE
01337A 1F2A B7 55      A          STA      WORK4      TO DO
01338A 1F2C CD 1F15    A          JSR      LOAD      TWO BYTE
01339A 1F2F B1 55      A          CMP      WORK4      STORE
01340A 1F31 27 01      1F34      BEQ      STRTS
01341A 1F33 99          *      SEC
01342A 1F34 BE 50      A STRTS   LDX      WORK1
01343A 1F36 81          *      RTS
01344          *

```

```

01346      *
01347      *****
01348      *
01349      *          BUILD A BYTE          *
01350      *
01351      *          A REGISTER CONTAINS BYTE      *
01352      *
01353      *****
01354      *
01355A 1F37 B7 51      A GETBY2 STA      WORK2
01356A 1F39 AD 0E      1F49      BSR      GETNYB
01357A 1F3B 24 0B      1F48      BCC      GETBRT
01358A 1F3D 38 51      A          ASL      WORK2
01359A 1F3F 38 51      A          ASL      WORK2
01360A 1F41 38 51      A          ASL      WORK2
01361A 1F43 38 51      A          ASL      WORK2
01362A 1F45 BA 51      A          ORA      WORK2
01363A 1F47 99          SEC
01364A 1F48 81          GETBRT RTS
01365      *
01366      *****
01367      *
01368      *          GET ONE CHARACTER AND      *
01369      *          CHECK FOR VALID HEX NUMBER  *
01370      *
01371      *          A REGISTER CONTAINS OUTPUT  *
01372      *
01373      *          X REGISTER DESTROYED        *
01374      *
01375      *          CARRY SET IF VALID HEX NUMBER *
01376      *
01377      *****
01378      *
01379A 1F49 CD 1E5F      A GETNYB JSR      CHRIN      GET CHARACTER
01380A 1F4C 98          CLC
01381A 1F4D A1 0F      A          CMP      #$0F      VALID HEX?
01382A 1F4F 22 01      1F52      BHI      GETRET      NO
01383A 1F51 99          SEC                      YES
01384A 1F52 81          GETRET RTS
01385      *
01386      *****
01387      *
01388      *          BUILD ADDRESS                *
01389      *
01390      *          A,X REGISTERS DESTROYED      *
01391      *
01392      *          ADDR,H,ADDRL CONTAIN ADDRESS  *
01393      *
01394      *          CARRY SET IF NEW ADDRESS      *
01395      *
01396      *****
01397      *
01398A 1F53 CD 1DF5      A GETADR JSR      CLRTAB      BLANK DISPLAY
01399A 1F56 AD 58      1FB0      BSR      PRTADR
01400A 1F58 AD EF      1F49 BLDADR BSR      GETNYB      GET CHARACTER
01401A 1F5A 25 0A      1F66      BCS      GETAD1      VALID HEX
01402A 1F5C A1 10      A          CMP      #$10
01403A 1F5E 27 2B      1F8B      BEQ      GETRTS

```

PAGE 028 CBUG05 .SA:1

01404A 1F60 A1 11	A	CMP	#\$11	NO ENTER?
01405A 1F62 27 27	1F8B	BEQ	GETRTS	NO TRY AGAIN
01406A 1F64 20 ED	1F53	BRA	GETADR	
01407A 1F66 3F 52	A	GETAD1 CLR	ADDRH	INIT HIGH ADDRESS
01408A 1F68 B7 53	A	STA	ADDRL	PUT CHAR AWAY
01409A 1F6A AD 44	1FB0	BSR	PRTADR	PRINT NEW ADDRESS
01410A 1F6C AD DB	1F49	GETALP BSR	GETNYB	GET ANOTHER CHAR
01411A 1F6E 24 12	1F82	BCC	GETARG	VALID?
01412A 1F70 48		ASLA		YES
01413A 1F71 48		ASLA		SHIFT IT IN
01414A 1F72 48		ASLA		
01415A 1F73 48		ASLA		
01416A 1F74 AE 04	A	LDX	#4	
01417A 1F76 48		GETASF ASLA		
01418A 1F77 39 53	A	ROL	ADDRL	
01419A 1F79 39 52	A	ROL	ADDRH	
01420A 1F7B 5A		DECX		
01421A 1F7C 26 F8	1F76	BNE	GETASF	
01422A 1F7E AD 30	1FB0	BSR	PRTADR	PRINT NEW ADDR
01423A 1F80 20 EA	1F6C	BRA	GETALP	GET ANOTHER CHAR
01424A 1F82 A1 10	A	GETARG CMP	#\$10	
01425A 1F84 27 05	1F8B	BEQ	GETRTS	
01426A 1F86 A1 11	A	CMP	#\$11	IS ENTER?
01427A 1F88 26 E2	1F6C	BNE	GETALP	NO TRY AGAIN
01428A 1F8A 99		SEC		YES SET FLAG
01429A 1F8B 81		GETRTS RTS		
01430	*			

PAGE 029 CBUG05 .SA:1

```

01432          *
01433          *****
01434          *
01435          *      PRINT ONE BYTE INTO PAIR      *
01436          *      OF DISPLAY DIGITS              *
01437          *
01438          *      A REGISTER CONTAINS BYTE        *
01439          *
01440          *      X REGISTER POINTS TO 1ST        *
01441          *      DIGIT OF PAIR                    *
01442          *
01443          *****
01444          *
01445A 1F8C AE 04      A PRTDAT LDX      #4      PRINT IN LAST TWO DIGIT
01446A 1F8E BF 50      A PRTBYT STX      WORK1
01447A 1F90 B7 55      A          STA      WORK4
01448A 1F92 44          *          LSRA
01449A 1F93 44          *          LSRA
01450A 1F94 44          *          LSRA
01451A 1F95 44          *          LSRA
01452A 1F96 97          *          TAX
01453A 1F97 D6 1E87     A          LDA      CTABL,X
01454A 1F9A BE 50      A          LDX      WORK1
01455A 1F9C E7 49      A          STA      DTABL,X
01456A 1F9E B6 55      A          LDA      WORK4
01457A 1FA0 A4 0F      A          AND      #$0F
01458A 1FA2 97          *          TAX
01459A 1FA3 D6 1E87     A          LDA      CTABL,X
01460A 1FA6 BE 50      A          LDX      WORK1
01461A 1FA8 E7 4A      A          STA      DTABL+1,X
01462A 1FAA CD 1DFD     A          JSR      DISTAB
01463A 1FAD B6 55      A          LDA      WORK4
01464A 1FAF 81          *          RTS
01465          *
01466          *****
01467          *
01468          *      PRINT ADDRESS ADDRH,ADDRL      *
01469          *
01470          *      X REGISTER DESTROYED          *
01471          *
01472          *****
01473          *
01474A 1FB0 B7 56      A PRTADR STA      WORK5
01475A 1FB2 BF 54      A          STX      WORK3
01476A 1FB4 B6 52      A          LDA      ADDRH
01477A 1FB6 5F          *          CLRX
01478A 1FB7 AD D5      1F8E        BSR      PRTBYT
01479A 1FB9 B6 53      A          LDA      ADDR
01480A 1FBB AE 02      A          LDX      #2
01481A 1FBD AD CF      1F8E        BSR      PRTBYT
01482A 1FBF B6 56      A          LDA      WORK5
01483A 1FC1 BE 54      A          LDX      WORK3
01484A 1FC3 81          *          RTS
01485          *

```

PAGE 030 CBUG05 .SA:1

```

01487          *
01488A 1FC4 80      TIRQWV RTI
01489          *
01490A 1FC5 80      IRQV   RTI
01491A 1FC6 80      RTI
01492          *
01493          1FC7    A TIRQV EQU      *
01494A 1FC7 A6 40    A      LDA      #$40
01495A 1FC9 B7 09    A      STA      TIMEC
01496A 1FCB CD 1916  A      JSR      LOCSTK
01497A 1FCE E6 04    A      LDA      4,X
01498A 1FD0 BA 57    A      ORA      WORK6
01499A 1FD2 E7 04    A      STA      4,X
01500A 1FD4 CC 1928  A      JMP      PCOUNT
01501          *
01502A 1FD7 CD 1E07  A PWRDWN JSR      CLRDIS
01503A 1FDA 8E      A      STOP
01504          *
01505A 1FF6          A      ORG      $1FF6
01506          *
01507A 1FF6 0046    A      FDB      TIRQW
01508A 1FF8 0043    A      FDB      TIRQ
01509A 1FFA 0040    A      FDB      IRQ
01510A 1FFC 1856    A      FDB      SWI
01511A 1FFE 1800    A      FDB      RESET
01512          *
01513          END
TOTAL ERRORS 00000--00000

```