IMSAI

8048

CONTROL COMPUTER

USER MANUAL

July, 1977

TABLE OF CONTENTS

PREFACE

This manual is designed to guide the user in effective use of the single board micro-controller 8048 CC. It is not intended as a comprehensive guide to the INTEL 8048 Microprocessor chip. For that, a careful reading and study of the INTEL MCS-48™ Microcomputer User's Manual is required. For further information on programming techniques the INTEL MC8-48™ Assembly Language Manual will prove very instructive.

I. FUNCTIONAL DESCRIPTION

The 8048 CC is a single board microcontroller built around the INTEL 8048 Micorprocessor. This microprocessor, optimized for controller applications, is the world's first single chip microprocessor containing all the following features:

1. 8 Bit CPU
2. 1K words of ROM or compatible EPROM program memory
3. 64 words of Internal Registor Memory
4. 27 I/O lines
5. Internal Timer/EVENT counter
6. Oscillator and Clock Driver
7. Reset Circuit
8. Interrupt Circuit
9. Single 5 Volt Supply

NOTE: The 8035-8 provided has a cycle time of 4.1905 us and a timer rate of 134.095 us per count.

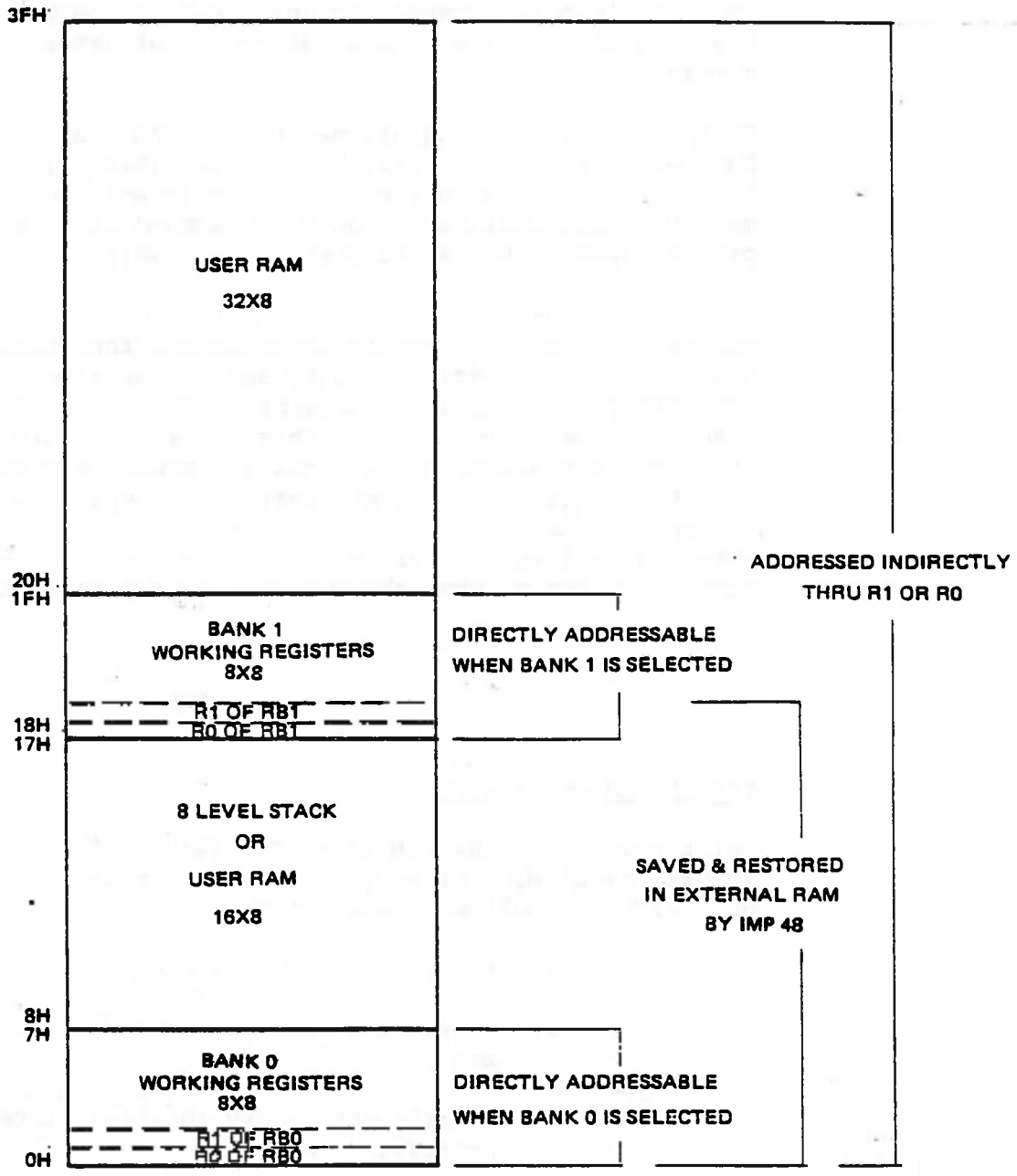
In addition, the following system features have been added to create a 1-board, user programmable controller suitable for use with model railroads, ham radios, household appliances, lights, light shows, and a myriad of other applications:

1. Cassette Interface
2. Serial I/O (RS232, current loop)
3. 5 relays capable of handling 2A at 220 Volts, or 5 Amps at 24 VDC
4. 1K (optional additional 1K) of user programmable program memory
5. Power Supply or battery operation
6. Socket for 2K EPROM

The INTEL 8048/8747 micro-processor chip is designed to accomodate three separate and unique memory spaces: Program memory, Internal register memory, and External RAM (Random access memory).

Program memory, in the simplest case, consists of 1K (1024) bytes of ROM/EPROM on the 8048/8748 chip itself, and is easily expandable to 4K with the addition of additional ROMS/EPROMS. By using memory banking techniques, the program memory space can be expanded to 65K or more. The program memory is the only memory space from which instructions can be fetched and executed.

Internal register memory (IRM) can be divided into 4 separate, though contiguous, areas: Register bank 0, Register bank 1, the stack, user defined RAM. A map of the internal memory is shown in Figure 1.



IN ADDITION R0 OR R1 OF EITHER BANK
MAY BE USED TO ADDRESS 256 WORDS
OF EXTERNAL RAM

FIGURE 1

External RAM memory, as the name implies, is not contained on the 8048 chip. The instruction set allows for direct access of 256 bytes of external RAM memory. By use of memory banking this can also be expanded to any level in 256 byte increments. The 8048 CC contains provisions for 2K of external RAM memory.

Both internal and external memory are RAM memory and can be both written into and read from by the user. There are no instructions for either reading from or writing into program memory. Instructions may only be fetched and executed from program memory by the microprocessor itself.

If program memory is the only memory space in which programs may reside and there are no instructions for writing into this memory space, how does the user enter a program into the 8048 CC? IMSAI has mapped the external RAM memory and program memory into the same memory space. This allows the user to enter data into what appears to be external RAM memory and allows the 8048 to execute these same programs from what now appears to be program memory. Accordingly, external memory will hereafter be used when referring to either program memory or external RAM memory when the two can be used interchangeably.

IMP-48 System Monitor

Built into the 8048/8748 is a 1K System Monitor IMP-48 (Imsai Monitor Program) which provides the user with the following facilities:

1. Enter programs into program memory
2. Enter data into both internal and external memory
3. Examine any memory location, internal or external
4. Execute a user program in standalone mode

5. Execute a user program with software breakpoints

6. Save/retrieve user programs on cassette tape

7. Save and retrieve programs for serial port, i.d., teletype paper tape reader

8. Adjust cassette volume and tone controls to assure error free tape operation

9. Run complete self-diagnostics

These facilities are more completely described in the 8048 CC Users Guide section of this manual and Appendix D.

It should be apparent that both the IMP-48 and any users program access the same 8048 resources, e.e., stack, registers, internal memory, external memory, timer. The IMP-48 must manipulate these resources in such a way as to force the fewest restrictions while using these same resources. Unfortunately some restrictions were necessary in order to implement several of the monitor features, notably the breakpoint function.

Restrictions on User's Program in Debug Mode

There is continuous interaction between the user's program and the system monitor while operating in the breakpoint mode. Therefore, the following restrictions were necessary, to allow the monitor to perform the required housekeeping. These are:

1. R7 of Register Bank 2 should not be used at all in breakpoint mode. Whenever a breakpoint occurs, the monitor saves the A reg in R7 of RB-2, destroying whatever was previously there.

2. Location FE1 through FFD of external memory should not be used for any reason (program storage or data) when operating in breakpoint mode. When a breakpoint is reached and the IMP-48 reentered, internal memory locations 0 - 19_{16} (Register Bank 0, Registers 0 and 1 of Register Bank 1 and the stack) are saved in external RAM locations FE4 - FED. This internal memory is restored prior to returning to the users program. (See Figure 3)

Locations FE1 and FE2 are used to save the 2 bytes of user program that are replaced by EN I instruction to implement the breakpoint.

Locations FFE and FFF are memory mapped ports for the display and keypad. Accessing these locations (read and write) will cause mostly predictable but usually undesirable results.

Location FE3 holds the stack ptr to the stored stack needed for return to the user program.

3. The 8048 stack has room for eight levels of nesting. Since breakpointing requires one level, the user is limited to 7 levels within his own program when operating in breakpoint mode. Exceeding this limit will result in the lowest levels being lost and will cause unpredictable results.

4. Since all user programs reside at location 800H or higher (the second 2K memory block) all jumps and calls within a users program must have MB1 selected. It is therefore recommended that the first instruction of a user program be a SEL MB1 (F5).

Figure 3

Location	Description
FFF	8279 Control port
FFE	8279 data port
FFD	R1 of RB1
FFC	R0 of RB1
FFB } FEC	stack
FEB	R7 of RB0
FEA	R6 of RB0
FE9	R5 of RB0
FE8	R4 of RB0
FE7	R3 of RB0
FE6	R2 of RB0
FE5	R1 of RB0
FE4	R0 of RB0
FE3	Stack ptr for return
FE2	Break byte
FE1	Byte following break byte

If the user wishes to call any of the monitor routines, MBO must first be selected with the SEL MBO instruction.

5. The monitor uses location FE1 - FFD. Therefore, if only 1K of memory is installed on the 8048 CC it must reside at locations C00 - FFFH (U18-U25).

6. NOTE: Breakpoints should not be placed at RET (83), RETR (93), JMPP @A (B3), or any single byte instruction preceeding the destination of a CALL or BRANCH. There are no restrictions on placing breakpoints at 2 byte instructions except that the breakpoint must be the first of the 2 bytes.

II. THEORY OF OPERATION

The IMSAI 8048 CC (Control Computer) is an INTEL 8048/8748/8035 based single card computer and development system. The CPU chip has three versions: the 8035 with no internal program memory, the 8048 with 1K bytes of internal ROM, and the 8748 with 1K bytes of internal EPROM.

For internal operation of the CPU and other chips manufactured by Intel, see the "MCS-48 Microcomputer User's Manual" provided. NOTE: The 8035-8 provided has a cycle time of 4.1905 µs and a timer rate of 134.095 µs per count.

The 8048 CC uses a multiplexed address and data bus and a latched address bus. The multiplexed bus has valid data on the trailing (rising) edge of the read and write strobes and is therefore called the "data bus". This bi-directional bus is connected to the 8279 keyboard and display controller, the RAM and the 2716 2K PROM. The data bus also goes to two 74LS174 latches that are strobed by $\overline{ALE}$ to latch the low order (A0 thru A7) addresses from the data bus. The high order addresses are output on the lower four bits of port 2 of the 8048. These lines are buffered by the four remaining latches in the 74LS174's. The latched address bus is connected to the RAM, the 2716 PROM and to a gate array to enable the 8279 Keyboard and Display Controller.

The 8048 CC has a direct address range of 4,096 bytes and extension bits (A12, A13) to enable address extension to 16K bytes. The address range is divided as follows: 000(hex) to 3FF, 8048 ROM or 8748 PROM or the first half of the 2716 PROM if the external enable line of the CPU is held high; 400 to 7FF, the second half of the 2716 PROM; 800 to FFD, RAM; FFE, data port of 8279 Keyboard and Display Controller; FFF, control port of 8279.

Data is strobed out of the CPU by $\overline{WR}$ which is buffered by a 74LS08. Data is strobed into the CPU by $\overline{RD}$ or $\overline{PSEN}$. $\overline{RD}$ is used to read in data. $\overline{PSEN}$ is used to read in from program memory. However, these two memory areas are overlaid so the two Read strobes are ORed in a 74LS08. The Read strobes are also inhibited for locations FFE and FFF because those locations are reserved for the 8279 keyboard and display controller.

Keyboard and Display

The 8279 controller uses $\overline{ALE}$ from the 8048 as a clock and is preset to divide by 4, resulting in an internal clock frequency of 100 KHz. This is internally divided

down by a 10 stage binary counter and the last 4 stages appear as signals S0 - S3. These outputs are decoded by a one-of-ten decoder driver (7445) and drive the digits of the display one at a time. The 16 x 8 internal RAM of the 8279 is scanned by the internal counter and the RAM data is presented to 8279 outputs A0 - A3 and B0 - B3. The outputs are buffered and drive the segments and decimal points of the display.

The scanning outputs S0 - S2 also go to a 74LS42 used as a one-of-8 decoder. The first three outputs of the 74LS42 scan the three rows of the keyboard, and if a button in a given row is depressed one of the inputs R0 - R7 will detect the closure when that row is being scanned. After a debounce period, if the button is still depressed the number representing the location of the button will be entered into the FIFO register of the 8279. For a more detailed explanation see the MCS-48 User's Manual section on the 8279.

I/O

The 8048 Chip has 27 IO lines. Of these, 8 are being used by the data bus. Another 8 (port 1) is available for general I/O. T1 and T0, which are software testable flags, are brought out for the user. $\overline{\text{INT}}$ is brought out, but the interrupt is used in the monitor for breakpoints and is tied to ground through a 10K resistor. If the monitor is not used, provisions are made to pull/up the $\overline{\text{INT}}$ input.

The last 8 I/O lines from the CPU (port 2) are used for a large variety of uses. Line P26 and P27 can be used for general output, but they are reserved for expansion of I/O by the addition of 8243 I/O expanders. P24 and P25 may be used for general output but they are intended for handshaking for port 1. As has already been mentioned, P20 - P23 are used for high order address. They are also used along with PROG to control the 8243 I/O expander.

The data address range of the 8048, as opposed to the program address range, is only 256 bytes because the high order 4 address bits do not exist in the I/O instructions. Therefore, if you wish to read or write to RAM you must set the high order 4 bits at port 2-0 to 2-3. Note that the data written to port 2-0 to 2-3 will be destroyed by any I/O command to ports 4 to 7. The easiest way around this is to write the 4 bits every time an I/O instruction

is performed; this is the way it is done in the monitor.

The remaining system I/O is implemented with an 8243 I/O expander. This provides 4 four bit ports (4 thru 7). Instructions are provided for not only input and output but to or and and to the ports. This allows control of individual bits.

Port 4 is available to the user. Port 5 drives 4 of the relays (the fifth has a TTL input uncommitted driver and can be configured as desired). Port 6 is specified as an input port. Bits 0 and 1 are used for handshaking for port 1. Bit 2 is the teletype and RS232 input. Either RS232 or teletype may be used by simply leaving the other input open. Bit 3 is the cassette input. Port 7 is specified as an output port. Bits 0 and 1 may be used for output or by jumpering may become address extension bits A12 and A13. Bit 2 is the teletype and RS232 output and bit 3 is the cassette output.

III. ASSEMBLY INSTRUCTIONS----------

The proper location and orientation of parts mounted on the printed circuit board is shown in the Board Silk Screen Drawing. Before soldering, ensure that each component is mounted in the correct position and oriented properly.

BEFORE PROCEEDING

Unpack all parts and check against the parts list. Do not discard any packing material until all parts are accounted for.

ASSEMBLY OF PRINTED CIRCUIT BOARD

Note that all components are mounted on the side of the board with the white silkscreened markings.

1.( ) Install and solder eight 47 ohm, ¼ watt (yellow-violet-black) resistors at the indicated locations:

( ) R1	( ) R6
( ) R2	( ) R11
( ) R4	( ) R12
( ) R5	( ) R27

2.( ) Install and solder one 2.2K ohm, ¼ watt (red-red-red) resistor at location R3.

3.( ) Install and solder thirteen 1K ohm, ¼ watt (brown-black-red) resistors at the indicated locations:

( ) R7	( ) R33	( ) R37
( ) R21	( ) R34	( ) R38
( ) R29	( ) R35	( ) R39
( ) R30	( ) R36	( ) R40
		( ) R41

4.( ) Install and solder one 220 ohm, ¼ watt (red-red-brown) resistor at location R19.

5.( ) Install and solder one 3.3K ohm, ¼ watt (orange-orange-red) resistor at location R8.

6.( ) Install and solder three 4.7K ohm, ¼ watt (yellow-violet-red) resistors at the indicated locations:

( ) R9	( ) R13	( ) R16

7.( ) Install and solder six 470 ohm 1/4 watt (yellow-violet-brown) resistors at the indicated locations:

( ) R10 ( ) R24
( ) R22 ( ) R25
( ) R23 ( ) R28

8.( ) Install and solder three 22K ohm 1/4 watt (red-red-orange) resistors at the indicated locations:

( ) R14 ( ) R26
( ) R15

9.( ) Install and solder one 33K ohm 1/4 watt (orange-orange-orange) resistor at location R17.

10.( ) Install and solder one 620 ohm 1/4 watt (blue-red-brown) resistor at location R18.

11.( ) Install and solder one 1 Meg ohm 1/4 watt (brown-black-green) resistor at location R20.

12.( ) Install and solder one 10K ohm 1/4 watt (brown-black-orange) resistor at location R31.

13.( ) Install and solder one 47K ohm 1/4 watt (yellow-violet-orange) resistor at location R32.

14.( ) Install and solder two 1N4002 diodes at the indicated locations, observing polarity (the stripe on the diode should match the stripe on the silkscreen):

( ) CR1 ( ) CR2

15.( ) Install and solder eight 1N4148 diodes at the indicated locations, observing polarity:

( ) CR3 ( ) CR7
( ) CR4 ( ) CR8
( ) CR5 ( ) CR9
( ) CR6 ( ) CR10

16.( ) Install the two 24-pin sockets at locations U4 and U11. Pin One of the sockets is indicated by a filled in inside corner. This should be the upper right corner when the socket is mounted. Also, note that Pin One's on the printed circuit board are marked by a square solder pad.

17.( ) Install the two 40 pin sockets at locations U6 and U9. Note that Pin One is to the upper left.

18.( ) If you are installing the socket kit, do it at this time. Install and solder nine 14-pin sockets and twenty-one 16-pin sockets. Note that Pin One of U5 is to the upper right and Pin One of the rest are to the upper left. The augat pins are used to socket U1, U2, and U3. The remaining augat pins may be used in jumper locations if desired. (For the majority of applications the board does not need to be jumpered.)

19.( ) Install and solder, if sockets are not used, eight (sixteen if the Expander Module Kit is used) 2102 IC's in the indicated locations. Note that the Pin One end of the IC's is marked with a notch or deep pit.

( ) U18	( ) U26	
( ) U19	( ) U27	
( ) U20	( ) U28	EXPANDER
( ) U21	( ) U29	MODULE
( ) U22	( ) U30	KIT
( ) U23	( ) U31	
( ) U24	( ) U32	
( ) U25	( ) U33	

20.( ) Install and solder one 3130 IC in location U1. If the 3130 comes with leads formed in two rows just insert in the indicated location with the package tab oriented as shown. If the leads need to be formed into two rows for insertion note that Pin 1 is the first lead clockwise from the package tab as observed looking at the lead end of the package.

21.( ) Install and solder one 4N25 IC at location U2.

22.( ) Install and solder one 4N33 IC at location U3.

23.( ) Install one 8243 IC at location U4.

24.( ) Install and solder two 74LS04 IC's at locations ( ) U5 and ( ) U17.

25.( ) Install and solder one 74LS08 IC at location U7.

26.( ) Install and solder one 74LS32 IC at location U8.

27.( ) Install one 8279 IC at location U9.

28.( ) Install and solder one 3082 IC at location U10.

29.( ) Install and solder two 74LS174 IC's at locations ( ) U12 and ( ) U13.

30.( ) Install and solder one 74LS30 IC at location U14.

31.( ) Install and solder one 74LS10 IC at location U15.

32.( ) Install and solder one 74LS27 IC at location U16.

33.( ) Install and solder one 7445 IC at location U34.

34.( ) Install and solder two 74LS126 IC's at locations ( ) U35 and ( ) U36.

35.( ) Install and solder one 74LS42 IC at location U37.

36.( ) Install and solder two 2N3904 transistors at locations ( ) Q1 and ( ) Q7. Orient the flat on the cases to match the silkscreen.

37.( ) Install and solder five 2N3906 transistors at the indicated locations. Orient the flat on the cases to match the silkscreen and note that the center lead is pulled toward the flat on the case to mount the transistor.

( ) Q2 ( ) Q5
( ) Q3 ( ) Q6
( ) Q4

38.( ) Install and solder five relays at the locations indicated. Do <u>not</u> bend any of the pins of the relays as it may change their contact gaps.

( ) K1 ( ) K4
( ) K2 ( ) K5
( ) K3

39.( ) Install and solder one 33uF tantalum capacitor at location C1. Make sure that the polarity is correct, that the component + lead goes in the appropriately marked hole.

40.( ) Install and solder fifteen (twenty-three if the RAM Expander Module kit is used) .1uF disk ceramic capacitors at the indicated locations. Make sure that the capacitors (or any other

components) do not extend more than .5 inch above the surface of the board to avoid interference with the acrylic cover.

( ) C2	( ) C13	( ) C18
( ) C7	( ) C14	( ) C19
( ) C9	( ) C15	( ) C20
( ) C10	( ) C16	( ) C21
( ) C12	( ) C17	( ) C22

( ) C23	( ) C27	EXPANDER
( ) C24	( ) C28	MODULE
( ) C25	( ) C29	KITS
( ) C26	( ) C30	

41. ( ) Install and solder the 39pF capacitor at location C3.

42. ( ) Install and solder the 470pF capacitor at location C8.

43. ( ) Install and solder two 20pF capacitors at locations ( ) C5 and ( ) C6.

44. ( ) Install and solder one 2.2uF tantalum capacitor at location C11. Be sure to observe polarity.

45. ( ) Install and solder the crystal at location Y1. The leads are inserted in the holes immediately to the right of the silkscreened square. The crystal may be positioned flat on the board inside the square or the crystal may stand upright if it is mounted snug against the board.

46. ( ) Install and solder one slide switch at location SW.

47. ( ) Insert the keyboard with the red reset button into the printed circuit board at location KEYPAD 1 and fasten with four #2-56 screws.

48. ( ) Insert the remaining keyboard into the printed circuit board at location KEYPAD 2 and fasten with four #2-56 screws.

49. ( ) Solder the pins of the keyboards.

50. ( ) Inspect the right angle mounting strip. This is the flexible strip made of flat metal strips bonded together with thin plastic strips. If the strip you have has more than 18 pins, cut off the excess.

51.( ) Insert the pins of the short side of the mounting strip into the row of holes on the edge of the NSA1298 display. The pins should be inserted from the side that has the clear plastic and the pins on the other side of the strip should point away from the plastic and off of the board. Solder

52.( ) Insert the assembly you have just completed into the printed circuit board at location NSA1298 DISPLAY. The plastic side should face the keypads. Push slightly back on the top of the display and solder the assembly to the printed circuit board.

53.( ) Push back on the upper part of the display with continuous pressure until the display extends only .5 inch above the printed circuit board.

54.( ) Insert the 3 pin header at location J2 and solder. Note that the bent pin side should be inserted into the board and that the straight pin side should point to the edge of the board. Use the row of three holes nearest the edge of the board.

55.( ) Insert the 6 pin header at location J6. The straight pins should face the edge of the board and the molded plastic should rest on the board. Solder.

56.( ) Insert and solder three 26-pin headers at the indicated locations. The straight pin side should face the edge of the board in all cases.

( ) J3 ( ) J4 ( ) J5

57.( ) Mount the red plexiglass (acrylic) panel above the printed circuit board using eight 6-32 x 3/4" allen head screws inserted through the acrylic sheet, eight #6 x ½" nylon spacers and the top of the circuit board. Fasten the screws with the nuts provided and tighten using the allen key provided.

58.( ) Mount the five rubber feet on the back of the circuit board. Two mount just below the outside binder holes near the top of the board. One goes under the left side of the copyright notice in the lower left corner of the board. The remaining two are placed under the keypads: the first midway between the "7" and the "F", and the last midway between the "1" and the "A".

59.( ) Connections may be made to any of the headers with the AMP connectors provided. Use of the crimping tool is the easiest but a longnose pliers may be used. Note that two crimps should be made. The first set of prongs are crimped to the wire and the second set to the insulation to provide strain relief. Also note that there are two kinds of pins. The high insertion force type is intended to be used in the 3 and 6 pin connector. The high insertion force type has a double bend in its contacting surface.

60. ( ) If you have purchased the EROM version, install the 8748 at location U6 (furnished with EROM PAK 1) or, if you have received EROM PAK 2, install the 8035 at location U6 and the 2716 EROM at location U11.

If you have purchased the ROM version, install the 2716 HTP PROM (NOTE: the 2716 HTP is not eraseable) at location U11 and the 8035 at location U6.

This completes the assembly of the 8048 CC Control Computer. Note however, that nothing is installed at location C4 and that there should be an extra .01uF capacitor remaining. Also note that an additional .47uF capacitor is provided to insert at location C9 if a high frequency response tape recorder is used.

POWER SUPPLY ASSEMBLY

Assembly consists of a few simple steps: First connect and solder the power cord to the power supply as shown on the Hookup Instructions. The white wire should go to lug #4 of the transformer and the black to lug #1. The green safety ground should be soldered to the lug that is held in place by the transformer mounting screw. Since this is an open frame supply, if there is any chance of coming in contact with the exposed power terminals (lugs 1 thru 4) they should by covered with insulating tape.

HOOKUP INSTRUCTIONS

AC INPUT CONNECTIONS

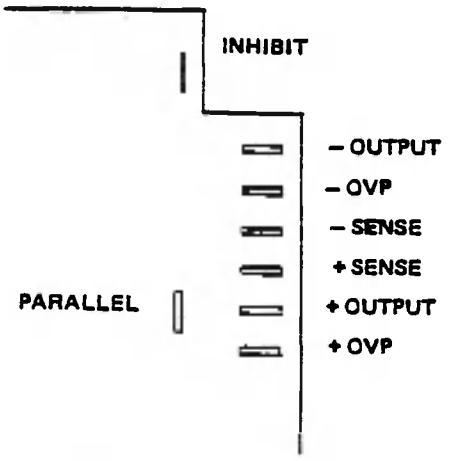


FIGURE 1

Now take the three foot length of gray wire and separate the two wires for an inch at each end. Strip back ¼" of insulation at one end and solder to the output lugs of the power supply. The tinned wire should to to the + OUTPUT lug and the copper wire should go to the -OUTPUT lug. The position of these lugs can be seen clearly in Figure 1.

All that remains is to put the connector on the other end of the wire. Strip the free ends of the wire of 1/8" of insulator, insert each wire into one of the crimp terminals supplied and make two crimps - one on the wire and one on the insulation. If a crimping tool is not available the job may be done with longnose pliers.

Next, insert the terminal that is crimped on the tinned wire into the center position of the black connector shell until the locking prong snaps into the rectangular cutout. To complete the assembly, push the remaining wire and connector into one of the remaining positions of the connector shell.

To use the supply, plug the black connector on to J2 of the 8048 CC. Note that the unused position goes toward the center of the printed cirucuit board. (Plugging the connector on backwards will not cause damage).

Schematics

were

here

1. schematic Diagram
2. Assembly Diagram

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Solder	15-0000001	15'	
Screw	20-1102001	8	2-56x3/16" Phillips Pan Head Machine
Screw	20-3716002	8	6-32x3/4" Allen Head, Button Head Black Oxide Machine
Nut	21-3120001	7	6-32 CAD Hex
Nut	21-3120002	1	6-32 Nylon Hex
Spacer	21-3600004	8	#6x½" Long Nylon
Housing	23-0500005	1	3 Pin, Single Row, Connector Housing AMP 87499-5
Housing	23-0500006	3	26 Pin, Double Row, Connector Housing AMP 2-87456-2
Housing	23-0500007	1	6 Pin, Double Row, Connector Housing AMP 87456-2
Switch	26-1200003	1	Right Angle Mount, Slide SPDT C & K 1101M2A
Relay	26-5100001	5	Minimum 5 Amp SPDT American Zettler AZ-535-16-2
Key	27-0000001	1	5/64" Allen Key (used for kit ass'y only)
Feet	28-0400002	5	Rubber Feet, 3M #5003
Foam Tape	28-0600003	1	1/16x½x½" Double Side Sticky
Resistor	30-2470362	8	47 Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, black)
Resistor	30-3220362	1	220 Ohm, ¼ Watt, 5% Carbon Film (red, red, brown)
Resistor	30-3470362	6	470 Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, brown)
Resistor	30-3620362	1	620 Ohm, ¼ Watt, 5% Carbon Film (blue, red, brown)
Resistor	30-4100362	13	1K Ohm, ¼ Watt, 5% Carbon Film (brown, black, red)

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Resistor	30-4220362	1	2.2K Ohm, ¼ Watt, 5% Carbon Film (red, red, red)
Resistor	30-4330362	1	3.3K Ohm, ¼ Watt, 5% Carbon Film (orange, orange, red)
Resistor	30-4470362	3	4.7K Ohm, ¼ Watt, 5% Carbon Fiom (yellow, violet, red)
Resistor	30-5100362	1	10K Ohm, ¼ Watt, 5% Carbon Film (brown, black, orange)
Resistor	30-5220362	3	22K Ohm, ¼ Watt, 5% Carbon Film (red, red, orange)
Resistor	30-5330362	1	33K Ohm, ¼ Watt, 5% Carbon Film (orange, orange, orange)
Resistor	30-5470362	1	47K Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, orange)
Resistor	30-7100362	1	1 Meg Ohm, ¼ Watt, 5% Carbon Film (brown, black, green)
Capacitor	32-0220020	2	20pF, Min. 50V, Disk Ceramic
Capacitor	32-0239010	1	39pF, Min. 10V, Disk Ceramic
Capacitor	32-0347010	1	470pF Disk Ceramic
Capacitor	32-2001010	1	.01uF Disk Ceramic
Capacitor	32-2010010	15	0.1uF Disk Ceramic
Capacitor	32-2047010	1	.47uF, 30V Ceramic
Capacitor	32-2122070	1	2.2uF, 16V Tantalum
Capacitor	32-2233070	1	33uF Tantalum
Diode	35-1000007	2	1N4002
Diode	35-1000012	8	1N4148
Transistor	35-2000002	2	2N3904
Transistor	35-2000003	5	2N3906

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Crystal	35-5000006	1	3.579545 MHz
Header	23-0400020	1	6 Pin Right Angle, AP922585-06
Header	23-0400021	1	3 Pin Right Angle, AP92983-01-03
Header	23-0400022	3	26 Pin Right Angle, AP922585-26
IC Socket	23-0800002	2	24 Pin, Solder Tail, Low Profile
IC Socket	23-0800004	2	40 Pin, Solder Tail, Low Profile
Pin	23-0900019	12	Wire Crimp Receptacle, AMP 1-87309-3 (only 9 required)
Pin	23-0900020	90	Wire Crimp Receptacle, AMP 86-016-3 (only 78 required)
4N25	36-0042501	1	Opto Isolator/4N25
4N33	36-0043301	1	Opto Isolator/4N33
3082	36-0308201	1	Transistor Array/CA3082
3130	36-0313001	1	OP AMP/CA3130S
74LS04	36-0740402	2	Hex Inverter (Low Power Schottky)/ DM74LS04N
74LS08	35-0740802	1	Quad 2 Input AND (LPS)/N74LS08A
74LS10	36-0741002	1	Triple 3 Input NAND (LPS)/DM74LS10N
74LS27	36-0742702	1	Triple 3 Input NOR (LPS)/SN74LS27N
74LS30	36-0743002	1	8 Input NAND (LPS)/SN74LS30N
74LS32	36-0743202	1	Quad 2 Input OR (LPS)/SN74LS32N
74LS42	36-0744202	1	4 Line to 10 Line Decoder (LPS)
7445	36-0744501	1	BCD to Decimal Decoder-Driver, Open Collector
74LS126	36-7412602	2	Quad Buffer, Three-State Output (LPS)/ SN74LS126N

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
74LS174	36-7417402	2	Hex D-Type Flip Flop With Clear (LPS)/ 9LS/74LS174
2102	36-0210201	8	2102 FDC/D2102AL-4
8243	36-0824301	1	/D8243
8279	36-0827901	1	/D8279
PROM PAK	88-0000040	1	8048CC EROM PAK 1 (Alt: 91-1700015 8048CC EROM PAK 2)
Strip	23-0900021	1	Multi-Position Right Angle Interconnect Connector, AMP 5244-71-5
Display	60-0000008	1	9 Digit Calculator, NSA 1298
PC Board	92-0000052	1	IMSAI 8048 Rev. 1
Keypad	93-0150001	1	Standard 12 Key, Centralab 2SMD2M00Q75007
Keypad	93-0150002	1	12 Key, Centralab 2SMD2M00Q75008
Panel	93-0150003	1	Plexiglass Panel Rev. B, 8048 Control Computer

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Solder	15-0000001	15'	
Screw	20-1102001	8	2-56x3/16" Phillips Pan Head Machine
Screw	20-3716002	8	6-32x3/4" Allen Head, Button Head Black Oxide Machine
Nut	21-3120001	7	6-32 CAD Hex
Nut	21-3120002	1	6-32 Nylon Hex
Spacer	21-3600004	8	#6x½" Long Nylon
Housing	23-0500005	1	3 Pin, Single Row, Connector Housing AMP 87499-5
Housing	23-0500006	3	26 Pin, Double Row, Connector Housing AMP 2-87456-2
Housing	23-0500007	1	6 Pin, Double Row, Connector Housing AMP 87456-2
Switch	26-1200003	1	Right Angle Mount, Slide SPDT C & K 1101M2A
Relay	26-5100001	5	Minimum 5 Amp SPDT American Zettler AZ-535-16-2
Key	27-0000001	1	5/64" Allen Key (used for kit ass'y only)
Feet	28-0400002	5	Rubber Feet, 3M #5003
Foam Tape	28-0600003	1	1/16x½x½" Double Side Sticky
Resistor	30-2470362	8	47 Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, black)
Resistor	30-3220362	1	220 Ohm, ¼ Watt, 5% Carbon Film (red, red, brown)
Resistor	30-3470362	6	470 Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, brown)
Resistor	30-3620362	1	620 Ohm, ¼ Watt, 5% Carbon Film (blue, red, brown)
Resistor	30-4100362	13	1K Ohm, ¼ Watt, 5% Carbon Film (brown, black, red)

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Resistor	30-4220362	1	2.2K Ohm, ¼ Watt, 5% Carbon Film (red, red, red)
Resistor	30-4330362	1	3.3K Ohm, ¼ Watt, 5% Carbon Film (orange, orange, red)
Resistor	30-4470362	3	4.7K Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, red)
Resistor	30-5100362	1	10K Ohm, ¼ Watt, 5% Carbon Film (brown, black, orange)
Resistor	30-5220362	3	22K Ohm, ¼ Watt, 5% Carbon Film (red, red, orange)
Resistor	30-5330362	1	33K Ohm, ¼ Watt, 5% Carbon Film (orange, orange, orange)
Resistor	30-5470362	1	47K Ohm, ¼ Watt, 5% Carbon Film (yellow, violet, orange)
Resistor	30-7100362	1	1 Meg Ohm, ¼ Watt, 5% Carbon Film (brown, black, green)
Capacitor	32-0220020	2	20pF, Min. 50V, Disk Ceramic
Capacitor	32-0239010	1	39pF, Min. 10V, Disk Ceramic
Capacitor	32-0347010	1	470pF Disk Ceramic
Capacitor	32-2001010	1	.01uF Disk Ceramic
Capacitor	32-2010010	15	0.1uF Disk Ceramic
Capacitor	32-2047010	1	.47uF, 30V Ceramic
Capacitor	32-2122070	1	2.2uF, 16V Tantalum
Capacitor	32-2233070	1	33uF Tantalum
Diode	35-1000007	2	1N4002
Diode	35-1000012	8	1N4148
Transistor	35-2000002	2	2N3904
Transistor	35-2000003	5	2N3906

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
Crystal	35-5000006	1	3.579545 MHz
Header	23-0400020	1	6 Pin Right Angle, AP922585-06
Header	23-0400021	1	3 Pin Right Angle, AP92983-01-03
Header	23-0400022	3	26 Pin Right Angle, AP922585-26
IC Socket	23-0800002	2	24 Pin, Solder Tail, Low Profile
IC Socket	23-0800004	2	40 Pin, Solder Tail, Low Profile
Pin	23-0900019	12	Wire Crimp Receptacle, AMP 1-87309-3 (only 9 required)
Pin	23-0900020	90	Wire Crimp Receptacle, AMP 86-016-3 (only 78 required)
4N25	36-0042501	1	Opto Isolator/4N25
4N33	36-0043301	1	Opto Isolator/4N33
3082	36-0308201	1	Transistor Array/CA3082
3130	36-0313001	1	OP AMP/CA3130S
74LS04	36-0740402	2	Hex Inverter (Low Power Schottky)/ DM74LS04N
74LS08	36-0740802	1	Quad 2 Input AND(LPS)/N74LS08A
74LS10	36-0741002	1	Triple 3 Input NAND (LPS)/DM74LS10N
74LS27	36-0742702	1	Triple 3 Input NOR (LPS)/SN74LS27N
74LS30	36-0743002	1	8 Input NAND (LPS)/SN74LS30N
74LS32	36-0743202	1	Quad 2 Input OR (LPS)/SN74LS32N
74LS42	36-0744202	1	4 Line to 10 Line Decoder (LPS)
7445	36-0744501	1	BCD to Decimal Decoder-Driver, Open Collector
74LS126	36-7412602	2	Quad Buffer, Three-State Output (LPS)/ SN74LS126N

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION/IDENTIFYING MARKS
74LS174	36-7417402	2	Hex D-Type Flip Flop With Clear (LPS)/ 9LS/74LS174
2102	36-0210201	8	2102 FDC/D2102AL-4
8035	36-0803501	1	/D8035
8243	36-0824301	1	/D8243
8279	36-0827901	1	/D9279
EA4748	88-0000022	1	IC Programmed "IMP 48"
Strip	23-0900021	1	Multi-Position Right Angle Interconnect Connector, AMP 5244-71-5
Display	60-0000008	1	9 Digit Calculator, NSA 1298
PC Board	92-0000052	1	IMSAI 8048 Rev. 1
Keypad	93-0150001	1	Standard 12 Key, Centralab 2SMD2M00Q75007
Keypad	93-0150002	1	12 Key, Centralab 2SMD2M00Q75008
Panel	93-0150003	1	Plexiglass Panel Rev. B, 8048 Control Computer

PARTS LIST

ITEM	IMSAI PART #	QUANTITY	DESCRIPTION
Solder	15-0000001	5'	
Wire	22-1020004	36"	20 Gauge, Stranded, Insulated, Gray Insulation, Alpha 1949
Line Cord	22-6000001	1	Belden 17239
Housing	23-0500005	1	3 Pin Connector Housing, Single Row AMP 87499-5
Pin	23-0900019	3	Wire Crimp Receptacle, AMP 1-87390-3 (only 2 required)
Power Supply	60-0000007	1	+5 Volts, 3.0 Amps, Case Size A, ACDC Electronics Model No. EC5N3
Documentation	81-0000042	1	PS-3A Package

IV. USER GUIDE

HARDWARE

Jumpers

The 8048 CC is normally used without making any jumper modifications but jumpers are provided to reconfigure the board.

Jumper JA allows the interrupt of the 8279 Keyboard and Display Controller to drive the $\overline{INT}$ input.

Jumper JB allows R31 to become a pull/up resistor instead of a pull down. This enables wired anding of open collector drivers driving the $\overline{INT}$ bus. However, it should be noted that the monitor requires $\overline{INT}$ to be low to use breakpoints.

Jumper JC disables the Internal ROM/Prom and enables access to the lower 1K bytes of the 2716 PROM.

Jumper JD is used to allow the 8748 to be programmed with external electronics.

Jumper JE is used to enable the use of address extension line A12.

Jumper JF is used to enable the use of address extension line A13.

Jumpers JG (4 jumpers) are used to disconnect a relay from its committed output port to either connect the relay driver to another line or to use the output port line for another purpose.

Jumper JH is used to connect the 8243 I/O expander to port 2-6 to allow the 8243 to be disabled. This allows more 8243's to be connected to P2-0 thru P2-3.

Jumper JI is used to select whether the entire 2K of the 2716 will be accessible or only the first or second K byte. This allows the lower or upper K byte to be accessible when the 8048/8748 is occupying the first kilobyte of the memory address space.

Jumper JJ disconnects the 8243 I/O expander so that the expander will not be damaged if the 8748 is programmed with off card electronics.

Connections

To use the 8048 CC regulated 5V should be connected to pin 2 and ground to pin 3 of J2 (3 pins). Alternatively a six volt battery may be connected with + to pin 1 and - to pin 3 of J2.

J1 is the expansion bus connector and is not provided in the kit but will be provided with expansion boards in the future. It is a standard 50 pin flat cable header available from AP or 3M among others. It should be noted that J3, 4 and 5 will accept 26 pin flat cable connectors.

J3 is configured as an 8 bit bidirectional port with handshaking. P2-4 and 2-5 should be used for outgoing handshakes and P6-0 and 6-1 for incoming handshakes.

J4 has the teletype, RS232 and Cassette I/O and a group of user definable I/O lines.

J5 has the relay contacts for K1 thru K4. Since these contacts may carry high voltages no low level signals go through this connector.

J6 connect to both the control input and the contacts of K5. K5 may be controlled by any on card control line or any other TTL signal connected tp pin 2.

Note that all the relays are activated by a logic low.

SOFTWARE

Exam/Modify Function

In order to examine and/or modify any given memory location, the following sequence of keys must be depressed:

1. EXAM/MODFY

2. PROG MEM if external memory is desired or REG MEM if internal is desired. Remember that since Program memory and external memory are mapped into the same space on the IMSAI 8048 that the PROG Key should be used for both.

3. 0-3 hexadecimal digits corresponding to a valid address for the memory space desired. For internal register memory valid addresses are 0-$3F_{16}$ while for external memory valid addresses are 0-FFD_{16}. Locations FFE and FFF are memory mapped to the displays and keypad. Examining them will result in erroneous operation.

E	P	A	A	A.	b	b	D	b

External

OR

E	r	A	A	A.	b	D	D	b

Internal

Where P = External memory designator
r = Internal register memory designator
AAA = hexadecimal address
DD - data byte at address AAA

At this point the user has three options

1. He can terminate the sequence by hitting ENTER.

2. He can display the next sequential location by hitting NEXT.

3. He can modify the data at the location displayed by entering dexadecimal data via the keypad.

When examining internal registers, 2 different conditions may occur. If the desired register lies between 0 to 19^H the IMP-48 will actually be examining and modifying external RAM locations FE4 to FED as described in Figure 1. This is because when control returns to the user, the internal registers will first be restored from these external locations.

The second condition, examining registers 1A to $3F_{16}$ will affect the actual register since these registers are not used by the IMP-48 and are not restored before entering a program.

Program Execution

Once a program has been entered into program memory via the keypad or external medium, the user can execute that program in either the debug or standalone modes. The standalone mode allows the user's program exclusive use of any and all system resources such as internal and external memory, keypad, display, interrupts, etc. In the debug mode, certain restrictions in the use of system resources are required, since the monitor needs some resources in order to function properly. See the section on restrictions for a discussion of these. The four forms of the EXEC/BREAK function are as follows:

I. Standalone mode

a) [EXEC] [,] [.] - Execute from user location 800_{16}

b) [EXEC] [addr] [,] [.] - Execute from location addr without breakpoints

II. Debug mode using breakpoints

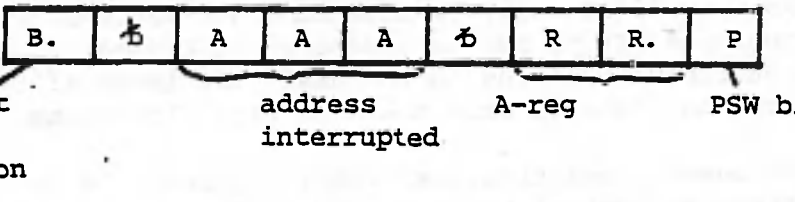
a) [BREAK] [addr1] [,] [addr2] [.]

Begin execution at addr1 and continue until reaching addr2 .

b) [BREAK] [,] [addr2] [.]

Continue from last breakpoint (or location 800H if there was no last breakpoint) until reaching addr2 .

The breakpoint function is implemented by substituting an EN I (05H enable external interrupt) for the existing byte at the specified address addr2 and the following address addr2 +1 and saving the existing bytes in external RAM, locations FE1, FE2. When the breakpoint is reached, an interrupt occurs and the monitor interrupt handler restores the original bytes and performs all necessary housekeeping to preserve and restore internal registers. After each breakpoint occurs the display will be set as appears below:



NOTE: Breakpoints should not be placed at RET (83), RETR (93), JMPP @A (B3), or any single byte instruction preceeding the destination of a CALL or BRANCH. There are no restrictions on placing breakpoints at 2 byte instructions except that the breakpoint must be the first of the 2 bytes.

Reset Function

Hitting the Reset Key forces a "power on" sequence of events to be activated and results in the display being cleared. Register Bank 0 is selected, interrupts are disabled, outstanding breakpoints are not restored. The user should manually restore them before proceeding.

Clear Entry Function

Any incorrectly entered function can be cleared before the final ENTER key by hitting the CLEAR ENTRY KEY. This will also blank the display and display the prompt character ("=").

Using the Relays

The 8048CC contains 5 relays, 4 of which are under immediate control of the CPU via a user program. The fifth relay is not connected to any I/O lines but can easily be jumpered to any of the unused I/O lines available on edge connector J3 or J4.

Relays RELY0 through RELY3 are connected to Port 5 of the 8243, Bits 0, 1, 2, 3 respectively. In order to close a particular relay, a zero must be sent out to the appropriate bit of Port 5. To open a relay, a one must be output to the appropriate bit of Port 5.

SAVING AND RETRIEVING PROGRAMS

The 8048 CC provides all the necessary hardware and software for saving and retrieving programs on cassette tape or any serial device operating with current loop or RS232 at 110 baud 1 start bit, 2 stop bits, no parity check (i.e., teletype paper tape). The data format for both media is described in Appendix A.

Cassette

The recording technique used for the cassette interface is bi-phase encoding. With this technique there is either 1 or 2 transitions per bit depending on whether 2 consecutive bits are the same or different. There is always a transition during the midpoint of each bit period. This allows the input routines to be self clocking on this center transition providing a great deal of immunity from tape speed fluctuations.

One bit is transmitted about every 675 microseconds, resulting in a baud rate of approximately 1500 bits/second.

The IMP-48 commands for reading and writing to the cassette are as follows:

[TOUT] [,]	[<end addr>]	[.]	starting address is always 800H, ending address as specified.
[TIN] [,]	[<end addr>]	[.]	

All storage and retrieval occurs to/from external memory only. There is no provision for saving or retrieving internal registers on any external medium.

Commands for reading a writing to teletype are as follows:

[TIN] [TTY] [<end addr>] [.]

[TOUT] [TTY] [<end addr>] [.]

All data is output and input as binary data and is not recognizable as ascii characters. Therefore no echoin occurs on input.

ADJUSTING THE RECORDING AND PLAYBACK LEVELS

General

Most inexpensive cassette recorders such as J.C. Penny's 6536 have automatic volume control while recording. Therefore volume setting is of no concern during record operations. The tone control often is coupled with a built in condenser microphone. During record operations, this mike should be turned off.

During playback volume is very important and the tone setting usually should be set for greatest treble response.

Specific

Two facilities are built into the 8048 to help the user adjust the volume and tone settings to ensure error free recording. A subroutine called SYNCST: located at 200H in the monitor can be executed using the EXEC command of the monitor:

[EXEC] [200] [,] [.]

This will cause a continuous stream of sync bytes (E6H) to be written to the cassette.

During input operations, all bytes read are displayed in round robin fashion on the display lights. Since an E6H is represented as a "4." on the displays, a solid line of non-changing "4."'s indicates a solid read.

To perform this test use the following command:

[TIN] [,] [BFF] [.]

Put the cassette in the playback mode and adjust volume and tone controls until a solid line of "4." is achieved without any light flicker.

In addition to these facilities all writes include a checksum, and all reads recompute the checksum and test for validity. If a read occurs without error, an equal sign will appear following the read. If an error occurs, ERR will appear.

It is a good idea to record 2 or more blocks of the same program on successive locations on the tape. This simplifies the task of regrouping if a read error occurs. If a wide frequency tape recorder is used it may help to change C8 from 470 pf to .01 uF (included).

```
IMSAI 8048 ASSEMBLER


PASS 1

PASS 2

LINE ADDR INST SOURCE
 0001           ;*******************************************************
 0002           ;       THIS SAMPLE DEMONSTRATES THE USE OF THE TIMER
 0003           ;       AND RELAYS. RELAYS 1 AND 3 ARE CLOSED FOR 250
 0004           ;       MILLISECONDS EACH SECOND WITHOUT AFFECTING
 0005           ;       RELAYS 0 AND 2.
 0006           ;*******************************************************
 0007           ;
 0008 0800              ORG 800H
 0009 0800 F5           SEL MB1          ;USER MEMORY BANK
 0010 0801 55           STRT T           ;TURN ON TIMER
 0011 0802 1415 RELAY:  CALL DLY750      ;DELAY 750 MILLISECONDS
 0012 0804 2305         MOV A,#5         ;MASK
 0013 0806 9D           ANLD P5,A        ;TURN ON RELAYS 1 AND 3
 0014 0807 140E         CALL DLY250      ;WAIT 250 MILLISECONDS
 0015 0809 230A         MOV A,#0AH       ;MASK
 0016 080B 8D           ORLD P5,A        ;TURN OFF RELAYS 1 AND 3
 0017 080C 0402         JMP RELAY        ;START AGAIN
 0018           ;
 0019 080E BA19 DLY250: MOV R2,#25       ;COUNT FOR 250 MILLISECONDS
 0020 0810 1419         CALL DLY10M      ;DELAY 10 MILLISECONDS USING TIMER
 0021 0812 EA10         DJNZ R2,DLY250+2 ;TOTAL 250 MSEC
 0022 0814 83           RET
 0023           ;
 0024 0815 BA4B DLY750: MOV R2,#75       ;COUNT FOR 750 MSEC
 0025 0817 0410         JMP DLY250+2
 0026           ;
 0027 0819 161D DLY10M: JTF $+4
 0028 081B 0419         JMP DLY10M       ;LOOP IF NOT TIMEOUT YET
 0029 081D 23B5         MOV A,#-75       ;TIMER COUNT FOR 10 MSEC
 0030 081F 62           MOV T,A          ;SET TIMER
 0031 0820 55           STRT T           ;START TIMER
 0032 0821 83           RET
 0033

HASH TABLE
20B2 20C7 20D0 2107 20D9 20E2 20EB 20F4 2036 2076 20FD 1D93 1FE3 1FAE 1E6E 1E83
0000 1F85 1EFC 1F9B 2052 1F1C 2064 206D 0000 20BC 208A 1F66 1EAC 1EB6 1EC0 20A8

NEXTSY=2142

SYMBOL TABLE

ADDRESS LINK TYPE FLAGS VALUE  NAME

  210F  0000  60    14    0802  RELAY
  211B  0000  60    14    0815  DLY750
  2128  0000  60    14    080E  DLY250
  2135  0000  60    14    0819  DLY10M
 D ASSEMBLY
```

APPENDIX A

Cassette Tape Data Format

Cassette

Leader	10 seconds worth of 0's
Frame 0	Bit Sync Byte $C3_{16}$. Used to locate center bit transitions
Frame 1	Byte Sync $E6_{16}$. Used to locate beginning bit of byte
Frame 2 - n	Data stream
Frame n + 1	Checksum. 2's complement of the sum, ignoring carry, of all bytes in data stream
Trailer	1 second worth of 0's

The cassette tape data is preceded by about 10 seconds worth of zeros so the data block can be located audibly and to set the automatic level control of many cassette recorders.

Serial (Paper Tape)

The data format is identical to the cassette except the leader consists of approximately 6 inches of leader and 6 inches of trailer. Instaed of a C3H, E6H for sync a record mark FFH is used to begin the record.

APPENDIX B

YMP-48 Program Listing

```
IMSAI 8048 ASSEMBLER


PASS 1

PASS 2

LINE ADDR INST SOURCE
 0001            ;*****************************************************
 0002            ;       IMP-48  IMSAI MONITOR PROGRAM
 0003            ;       TIMING SET FOR 8035-8
 0004            ;       COPYRIGHT IMSAI MANUFACTURING CORP
 0005            ;                 SAN LEANDRO, CALIFORNIA
 0006            ;                 AUGUST 24,1977
 0007            ;       ALL RIGHTS RESERVED
 0008            ;*****************************************************
 0009            ;       SYSTEM EQUATES
 0010            ;
 0011            ;       KEYPAD  NPUTS
 0012    = 0000 K0       =       0
 0013    = 0001 K1       =       1
 0014    = 0002 K2       =       2
 0015    = 0003 K3       =       3
 0016    = 0004 K4       =       4
 0017    = 0005 K5       =       5
 0018    = 0006 K6       =       6
 0019    = 0007 K7       =       7
 0020    = 0008 K8       =       8
 0021    = 0009 K9       =       9
 0022    = 000A KA       =       10
 0023    = 000B KB       =       11
 0024    = 000C KC       =       12
 0025    = 000D KD       =       13
 0026    = 000E KE       =       14
 0027    = 000F KF       =       15
 0028    = 0011 EXEC     =       11H     ;EXECUTE/BREAKPOINT FUNCTION
 0029    = 0012 EXAM     =       12H     ;EXAMINE/MODIFY FUNCTION
 0030    = 0013 PROGM    =       13H     ;PROGRAM MEMORY
 0031    = 0014 REGM     =       14H     ;REGISTER MEMORY
 0032    = 0015 NEXT     =       15H     ;NEXT MEMORY LOCATION
 0033    = 0016 ENTER    =       16H     ;ENTER FUNCTION
 0034    = 0017 CLEER    =       17H     ;CLEAR ENTRY
 0035    = 0000 TAPEO    =       0       ;CASETTE OUT
 0036    = 0001 TAPE     =       1       ;CASETTE IN
 0037    = 000F TTY      =       15      ;TELETYPE COMMANDS
 0038            ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 0039            ;       DISPLAY SEGMENT CODES
 0040            ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 0041            ;               A
 0042            ;            .......
 0043            ;       F    .     .  B
 0044            ;            .  G  .
 0045            ;            .......
 0046            ;            .     .
 0047            ;         E  .     .  C
 0048            ;            .......     .
 0049            ;               D        P
 0050            ;
 0051            ; A '1' IN A SEGMENT POSITION CORRESPONDS TO LIGHT ON
 0052            ;
```

```
0053                    ;                               PGFEDCBA        KEY
0054    = 003F D0       =                               00111111B       ;0
0055    = 0006 D1       =                               00000110B       ;1
0056    = 005B D2       =                               01011011B       ;2
0057    = 004F D3       =                               01001111B       ;3
0058    = 0066 D4       =                               01100110B       ;4
0059    = 006D D5       =                               01101101B       ;5
0060    = 007D D6       =                               01111101B       ;6
0061    = 0007 D7       =                               00000111B       ;7
0062    = 007F D8       =                               01111111B       ;8
0063    = 0067 D9       =                               01100111B       ;9
0064    = 0077 DAA      =                               01110111B       ;A
0065    = 007C DBB      =                               01111100B       ;B
0066    = 0039 DC       =                               00111001B       ;C
0067    = 005E DD       =                               01011110B       ;D
0068    = 0079 DE       =                               01111001B       ;E
0069    = 0071 DF       =                               01110001B       ;F
0070    = 0000 BLNK     =                               00000000B       ;BLANK
0071    = 0080 PERIOD   =                               10000000B       ;PERIOD
0072    = 0050 DR       =                               01010000B       ;R
0073    = 0041 EQSIGN   =                               01000001B       ;=
0074                    ;
0075                    ;*****************************************
0076                    ;       8279 COMMANDS
0077                    ;*****************************************
0078    = 0024 CLOCK    =       24H     ;PRESET VALUE=4 FOR 100KHZ SCAN
0079    = 0040 FIFORD   =       40H     ;READ FIFO NEXT INPUT
0080    = 00D0 CLEARD   =       0D0H    ;CLEAR DISPLAY TO BLANKS
0081    = 0090 DISP0    =       90H     ;DISPLAY AUTO INC FROM POS 0
0082    = 0091 DISP1    =       91H     ;DISPLAY POSITION 1
0083    = 0096 DISP6    =       96H     ;DISPLAY AUTO INC FROM POS 2
0084    = 0092 DISP2    =       92H     ;POSITION 2
0085    = 0095 DISP5    =       95H
0086    = 0007 FIFOCNT  =       7       ;MASK FOR FIFO STATUS COUNT
0087                    ;*****************************************************
0088                    ;       SAVE AREA MEMORY MAP
0089                    ;        FILLED UPON EXECUTION OF BREAKPOINT
0090                    ;*************************************************
0091                    ;LOCATION       CONTENTS
0092                    ;FFF            8279 CONTROL PORT (MEMORY MAPPED)
0093                    ;FFE            8279 DATA PORT (MEMORY MAPPED)
0094                    ;FFD            R1 OF REGISTER BANK 1
0095                    ;FFC            R0 OF REGISTER BANK 1
0096                    ;FFB-FEC        STACK
0097                    ;FEB            R7 OF RB0
0098                    ;FEA            R6 OF RB0
0099                    ;FE9            R5 OF RB0
0100                    ;FE8            R4 OF RB0
0101                    ;FE7            R3 OF RB0
0102                    ;FE6            R2 OF RB0
0103                    ;FE5            R1 IF RB0
0104                    ;FE4            R0 OF RB0
0105                    ;FE3            STACK POSITION PTR
0106                    ;FE2            REMOVED BREAKPOINT BYTE
0107                    ;FE1            BYTE FOLLOWING BREAKPOINT
0108                    ;*************************************************
0109                    ;
0110 0000                       ORG 0           ;RESET ENTRY POINT
0111 0000 15                    DIS I           ;TURN OFF INTERRRUPT SYSTEM
0112 0001 0453                  JMP INIT
```

```
0113                    ;
0114 0003                       ORG 3           ;EXTERNAL INTERRUPT ENTRY POINT
0115 0003 15                    DIS I
0116 0004 D5                    SEL RB1
0117 0005 0409                  JMP SAVE
0118 0007                       ORG 7           ;TIMER INTERRUPT ENTRY POINT
0119 0007 35                    DIS TCNTI
0120 0008 93                    RETR
0121 0009                       ORG 9
0122                    ; SAVE INTERNAL REGISTERS IN EXTERNAL MEMORY
0123 0009 8A0F SAVE:            ORL P2,#15      ;BANKING
0124 000B AF                    MOV R7,A        ;SAVE USER A REG
0125 000C F9                    MOV A,R1
0126 000D B9FD                  MOV R1,#0FDH    ;TOP OF REGISTER SAVE AREA
0127 000F 91                    MOVX @R1,A      ;SAVE R0 AT TOP
0128 0010 2319                  MOV A,#19H      ;# BYTES TO SAE
0129 0012 28                    XCH A,R0
0130 0013 C9                    DEC R1          ;NEXT SAVE LOCATION
0131 0014 91                    MOVX @R1,A
0132 0015 C8    SAVE1:          DEC R0
0133 0016 F0                    MOV A,@R0       ;GET INTERNAL RAM BYTE
0134 0017 91                    MOVX @R1,A      ;SAVE IN EXTERNAL MEMORY
0135 0018 F8                    MOV A,R0        ;GET COUNT
0136 0019 C9                    DEC R1
0137 001A 9615                  JNZ SAVE1       ;NOT DONE YET
0138 001C C7                    MOV A,PSW       ;STACK PTR
0139 001D 91                    MOVX @R1,A      ;SAVE
0140 001E 5442                  CALL INITIO     ;INIT ALL I/O DEVICES AND CLOCK
0141 0020 23FC                  MOV A,#DBB+PERIOD       ;"B."
0142 0022 541F                  CALL OUTDSP     ;DISPLAY
0143                    ; GET USERS RETURN ADDRESS FROM STACK
0144 0024 54D3                  CALL FNDST      ;PT AT USER STACK IN EXT MEM
0145 0026 07                    DEC A           ;BACK UP RETURN ADDRESS
0146 0027 07                    DEC A           ;BACK UP 2 BYTES
0147 0028 A8                    MOV R0,A        ;PC0-7
0148 0029 91                    MOVX @R1,A      ;BACK IN STACK
0149 002A 0302                  ADD A,#2        ;MUST WE BORROW
0150 002C 19                    INC R1          ;PT AT PC8-11
0151 002D 81                    MOVX A,@R1      ;GET IT
0152 002E AB                    MOV R3,A        ;SAVE STATUS
0153 002F 530F                  ANL A,#0FH      ;ONLY PC8-11
0154 0031 AD                    MOV R5,A
0155 0032 07                    DEC A
0156 0033 E637                  JNC BR1
0157 0035 AD                    MOV R5,A
0158 0036 91                    MOVX @R1,A      ;BACK IN STACK
0159 0037 546F BR1:             CALL DISPAD     ;ADDRESS OF BREAKPOINT
0160 0039 8A0F                  ORL P2,#0FH     ;BANK TO TOP
0161 003B B9E2                  MOV R1,#0E2H    ;BREAK BYTE
0162 003D 81                    MOVX A,@R1      ;GET BYTE
0163 003E AE                    MOV R6,A        ;READY FOR OUTPUT
0164 003F C9                    DEC R1          ;BYTE FOLLOWING BREAKPOINT PTR
0165 0040 81                    MOVX A,@R1
0166 0041 AF                    MOV R7,A        ;SAVE IT
0167 0042 5483                  CALL PUTM       ;BREAK BYTE REINSERTED INLINE,BUMP PC
0168 0044 FF                    MOV A,R7
0169 0045 AE                    MOV R6,A
0170 0046 5483                  CALL PUTM       ;FOLLOWING BYTE REINSERTED
0171 0048 D5                    SEL RB1
0172 0049 FF                    MOV A,R7        ;USER A REGISTER
```

```
0173 004A C5            SEL RB0
0174 004B 5468          CALL DISPRG     ;DISPLAY REGISTER
0175 004D FB            MOV A,R3        ;STATUS
0176 004E 47            SWAP A
0177 004F 5462          CALL DSG
0178 0051 0465          JMP KEY
0179                ;
0180 0053 54D3 INIT:    CALL FNDST      ;PT AT USER STACK
0181 0055 27            CLR A
0182 0056 91            MOVX @R1,A      ;DEFAULT RETURN ADDR=800H
0183 0057 2308          MOV A,#8
0184 0059 19            INC R1
0185 005A 91            MOVX @R1,A
0186 005B 3F            MOVD P7,A       ;INIT P7 TO OUTPUT 0
0187 005C 5442          CALL INITIO     ;INIT ALL I/O DEVICES AND CLOCK
0188 005E 0C            MOVD A,P4
0189 005F 0D            MOVD A,P5
0190 0060 0E            MOVD A,P6
0191                ;************************************
0192                ;       COMMAND ENTRY POINT
0193                ;************************************
0194 0061 5413 CMD:     CALL CLEAR
0195 0063 2341          MOV A,#EQSIGN
0196 0065 5427 KEY:     CALL KO         ;DISPLAY A THEN READ KEY
0197 0067 BC00          MOV R4,#0       ;EXT MEM FLAG
0198 0069 AE            MOV R6,A        ;SAVE
0199 006A 5413          CALL CLEAR      ;CLEAR DISPLAY
0200 006C FE            MOV A,R6        ;RESTORE INPUTTED KEY
0201 006D C682          JZ TAPEOUT      ;WRITE CASETTE/TELE
0202 006F 07            DEC A
0203 0070 C684          JZ TAPEIN       ;READ TAPE
0204 0072 03F1          ADD A,#-15      ;SKIP REST OF BOTTOM 2 ROWS
0205 0074 F288          JB7 CMDUD       ;COMMAND UNDEFINED
0206 0076 07            DEC A
0207 0077 C686          JZ EX           ;EXEC/BREAKPOINT FUNCTION
0208 0079 07            DEC A
0209 007A C694          JZ EXAMOD
0210 007C 03FD          ADD A,#-3
0211 007E F288          JB7 CMDUD       ;PROG/REG INVALID COMMANDS
0212 0080 0461          JMP CMD         ;IGNORE , . CLEAR
0213                ;
0214                ;
0215 0082 648A TAPEOUT: JMP TPOUT
0216 0084 246B TAPEIN:  JMP TPIN
0217 0086 04F2 EX:      JMP EXBR
0218    = 0088 ERROR = $
0219 0088 5413 CMDUD:   CALL CLEAR      ;CLEAR DISPLAY
0220 008A 2379          MOV A,#DE
0221 008C 541F          CALL OUTDSP
0222 008E 2350          MOV A,#DR       ;"R"
0223 0090 541F          CALL OUTDSP
0224 0092 0465          JMP KEY
0225                ;
0226                ;
0227                ;
0228    = 0094 EXAMOD = $
0229 0094 AA            MOV R2,A        ;KA0-7=0
0230 0095 2379          MOV A,#DE
0231 0097 5427          CALL KO
0232 0099 03ED          ADD A,#-13H     ;PROG KEY?
```

```
0233 009B C6A8          JZ PMEM          ;YES
0234 009D 07            DEC A            ;REG KEY
0235 009E 9688          JNZ ERROR
0236                ; REGISTER MEMORY KEY
0237 00A0 BC80          MOV R4,#80H      ;REG MEM FLAG
0238 00A2 23D0          MOV A,#DR+PERIOD ;"R."
0239 00A4 BB00          MOV R3,#0        ;DEFAULT =0
0240 00A6 04AA          JMP EX3
0241                ;
0242 00A8 23F3 PMEM:    MOV A,#0F3H      ;"P."
0243 00AA 541F EX3:     CALL OUTDSP      ;POS 0 DISP
0244 00AC BB08          MOV R3,#08H      ;DEFAULT=800H
0245 00AE 54B4          CALL INPADR      ;INPUT ADDRESS
0246 00B0 FC    EX4:    MOV A,R4         ;MEM FLAG
0247 00B1 37            CPL A
0248 00B2 F2BA          JB7 EX1          ;VALID EXTERNAL RANGE
0249 00B4 FA            MOV A,R2         ;GET KA0-7
0250 00B5 53C0          ANL A,#0C0H
0251 00B7 4B            ORL A,R3         ;KA8-11
0252 00B8 9688          JNZ ERROR        ;INVALID RANGE>63
0253 00BA FA    EX1:    MOV A,R2
0254 00BB A8            MOV R0,A         ;PC0-7
0255 00BC FB            MOV A,R3
0256 00BD AD            MOV R5,A         ;PC8-11
0257 00BE 546F          CALL DISPAD
0258 00C0 5499          CALL GETM        ;GET MEMORY BYTE AT PC,USE FLAG
0259 00C2 AE            MOV R6,A
0260 00C3 5468 EX0:     CALL DISPRG
0261 00C5 14D9          CALL INBYTE      ;READ MODIFIER IF ANY
0262 00C7 F7            RLC A            ;GET FLAG FROM INBYTE
0263 00C8 AF            MOV R7,A         ;SAVE IT
0264 00C9 5483          CALL PUTM        ;STORE MODIFIED OR ORIG VALUE BACK
0265 00CB FF            MOV A,R7         ;RETRIEVE FLAG
0266 00CC 1261          JB0 CMD
0267                ; "NEXT " KEY HIT BUMP PC AND CONTINUE
0268 00CE 1A            INC R2
0269 00CF FA            MOV A,R2
0270 00D0 96B0          JNZ EX4          ;NO OVERFLOW IN PC8-11
0271 00D2 1B            INC R3           ;BUMP PC8-11
0272 00D3 FB            MOV A,R3
0273 00D4 530F          ANL A,#0FH       ;ONLY 4 BITS VALID
0274 00D6 AB            MOV R3,A
0275 00D7 04B0          JMP EX4
0276                ;*************************************************
0277                ;       INBYTE- GET DATA BYTE FROM KEYPAD AND DISPLAY
0278                ;*************************************************
0279 00D9 54E6 INBYTE: CALL GTVALD       ;GET VALID HEX DIGIT
0280 00DB F6F0          JC INBOUT
0281 00DD BE00          MOV R6,#0
0282 00DF 2E    INB1:   XCH A,R6
0283 00E0 47            SWAP A
0284 00E1 53F0          ANL A,#0F0H
0285 00E3 4E            ORL A,R6
0286 00E4 AE            MOV R6,A
0287 00E5 2396          MOV A,#DISP6
0288 00E7 5408          CALL CTR8279
0289 00E9 FE            MOV A,R6
0290 00EA 5468          CALL DISPRG
0291 00EC 54E6          CALL GTVALD
0292 00EE E6DF          JNC INB1         ;STILL VALID DIGIT
```

```
0293 00F0 67    INBOUT: RRC A
0294 00F1 83            RET
0295                    ;
0296                    ;*********************************************
0297                    ;       EXECUTE AND BREAK FUNCTIONS
0298                    ;*********************************************
0299 00F2 23D0 EXBR:    MOV A,#DR+PERIOD        ;"R,"
0300 00F4 541F          CALL OUTDSP
0301 00F6 27            CLR A
0302 00F7 AA            MOV R2,A
0303 00F8 AB            MOV R3,A        ;KA=0
0304 00F9 541F          CALL OUTDSP
0305 00FB 54D3          CALL FNDST      ;PT AT USER STACK RETUREN
0306 00FD A8            MOV R0,A        ;PC0-7 FOR RETURN
0307 00FE AA            MOV R2,A
0308 00FF 19            INC R1
0309 0100 81            MOVX A,@R1      ;PC8-11
0310 0101 530F          ANL A,#0FH
0311 0103 AD            MOV R5,A        ;PC8-11
0312 0104 AB            MOV R3,A        ;SET KA
0313 0105 54C9          CALL DISPKA     ;ADDRESS IN POS 2
0314 0107 54B4          CALL INPADR     ;GET RETURN ADDR FROM USER
0315 0109 F7            RLC A
0316 010A AE            MOV R6,A        ;SAVE FLAG
0317 010B FA            MOV A,R2        ;KA0-7
0318 010C A8            MOV R0,A        ;PC0-7
0319 010D FB            MOV A,R3
0320 010E AD            MOV R5,A        ;SET PC FROM KA
0321 010F 546F EXBR1:   CALL DISPAD     ;NEW RETURN ADDR IN POS 2
0322 0111 FE            MOV A,R6        ;GET INPADR FLAG
0323 0112 1251          JB0 RESTORE     ;NO BREAKPOINT
0324 0114 27            CLR A
0325 0115 AA            MOV R2,A
0326 0116 AB            MOV R3,A        ;KA=0 DEFAULT
0327 0117 54C9          CALL DISPKA
0328 0119 54D3          CALL FNDST      ;SET UP STACK FOR RETURN
0329 011B F8            MOV A,R0        ;PC0-7 FOR RETURN
0330 011C 91            MOVX @R1,A
0331 011D 19            INC R1
0332 011E 81            MOVX A,@R1      ;GET PC8-11 RETURN
0333 011F 53F0          ANL A,#0F0H     ;STATUS BITS
0334 0121 4D            ORL A,R5        ;RETURN ADDR 8-11
0335 0122 91            MOVX @R1,A      ;BACK IN STACK
0336 0123 FE            MOV A,R6        ;FLAGS FROM INPADR
0337 0124 1251          JB0 RESTORE     ;ENTER KEY
0338                    ;
0339                    ; MUST GET BREAKPOINT ADDR AND STUFF EN I THERE
0340                    ;
0341 0126 2391          MOV A,#DISP1
0342 0128 5408          CALL CTR8279
0343 012A 23FC          MOV A,#DBB+PERIOD       ;"B."
0344 012C 541F          CALL OUTDSP
0345 012E 27            CLR A
0346 012F AA            MOV R2,A
0347 0130 AB            MOV R3,A
0348 0131 54B4          CALL INPADR     ;GET BREAK ADDR
0349 0133 FA            MOV A,R2
0350 0134 4B            ORL A,R3
0351 0135 C651          JZ RESTORE      ;NO ENTRY
0352 0137 FA            MOV A,R2
```

```
0353 0138 A8            MOV R0,A        ;PC0-7
0354 0139 FB            MOV A,R3
0355 013A AD            MOV R5,A        ;PC8-11 FOR BREAKPOINT
0356 013B 5499          CALL GETM       ;GET IN CODE BYTE
0357 013D AF            MOV R7,A        ;SAVE
0358 013E 2305          MOV A,#05H      ;EN I INSTRUCTION CODE
0359 0140 90            MOVX @R0,A      ;STUFF
0360 0141 548A          CALL BUMPPC     ;PC=PC+1
0361 0143 5499          CALL GETM       ;GET SECOND BYTE FOR STORAGE
0362 0145 8A0F          ORL P2,#0FH
0363 0147 B9E1          MOV R1,#0E1H    ;PT AT SECOND BYTE
0364 0149 91            MOVX @R1,A      ;SAVE UP HIGH
0365 014A FF            MOV A,R7        ;RETRIEVE BREAK BYTE
0366 014B 19            INC R1          ;R1=E2H
0367 014C 91            MOVX @R1,A      ;SAVE IT
0368 014D BE05          MOV R6,#05      ;NOP INST
0369 014F 5483          CALL PUTM       ;STUFF IN LINE
0370 0151 8A0F RESTORE: ORL P2,#15      ;MEM BANK
0371 0153 D5            SEL RB1
0372 0154 B8FB          MOV R0,#0FBH    ;PT AT STACK TOP
0373 0156 B918          MOV R1,#18H     ;COUNTER AND INTERNAL RAM PTR
0374 0158 80    RES1:   MOVX A,@R0      ;GET EXT BYTE
0375 0159 C9            DEC R1
0376 015A A1            MOV @R1,A       ;STORE INTERNAL
0377 015B C8            DEC R0
0378 015C 19            INC R1
0379 015D E958          DJNZ R1,RES1
0380 015F 80            MOVX A,@R0      ;GET STACK PTR
0381 0160 D7            MOV PSW,A       ;RESTORE PSW
0382 0161 D5            SEL RB1
0383 0162 B8FC          MOV R0,#0FCH    ;R0 OF RB1
0384 0164 80            MOVX A,@R0      ;GET R1 OF RB0
0385 0165 A9            MOV R1,A        ;RESTORE IT
0386 0166 19            INC R1          ;PT AT R0 OF RB1
0387 0167 80            MOVX A,@R0
0388 0168 A8            MOV R0,A
0389 0169 FF            MOV A,R7        ;USERS A REG
0390 016A 93            RETR            ;RETURN TO USERS PROG
0391                    ;
0392 016B 2330 TPIN:    MOV A,#30H      ;"I"
0393 016D 5427          CALL KO
0394 016F 03F1          ADD A,#-0FH
0395 0171 C697          JZ TTYIN        ;INPUT TAPE FROM TELETYPE
0396 0173 23B9          MOV A,#DC+PERIOD        ;"C."
0397 0175 34B6          CALL OC
0398 0177 BE00 SYNC:    MOV R6,#0       ;SYNC ON TAPE WITH 0E6H CHR
0399 0179 85            CLR F0
0400 017A 95    SYNC1:  CPL F0          ;SET NON-INVERT FLAG
0401 017B 7406          CALL BITIN      ;GET NEXT BIT
0402 017D D3E6          XRL A,#0E6H     ;EQU SYNC CHR?
0403 017F C685          JZ TPIN1        ;BRIF NON-INVERT SYNC
0404 0181 85            CLR F0          ;SET INVERT FLAG
0405 0182 37            CPL A           ;EQU INVERT SYNC CHR?
0406 0183 967A          JNZ SYNC1       ;BRIF NOT EQU
0407 0185 C5    TPIN1:  SEL RB0
0408 0186 7400          CALL CASIN      ;READ BYTE
0409 0188 34AF          CALL TAPINC     ;COMMON STUFF
0410 018A E985          DJNZ R1,TPIN1
0411 018C D5            SEL RB1
0412 018D E885          DJNZ R0,TPIN1
```

```
0413 018F C5           SEL RB0
0414 0190 7400         CALL CASIN      ;CHECKSUM FROM TAPE
0415 0192 6F   CHECK:  ADD A,R7        ;IS CHECKSUM THE SAME?
0416 0193 96D1         JNZ ERR1                ;CHECKSUM ERROR
0417 0195 0461         JMP CMD
0418 0197 23F3 TTYIN:  MOV A,#0F3H     ;"P."
0419 0199 34B6         CALL OC
0420 019B 7469 TALE1:  CALL TELEIN     ;READ BYTE
0421 019D 17           INC A           ;CHECK FOR RECORD MARK FF
0422 019E 969B         JNZ TALE1       ;NOT YET
0423 01A0 C5   TALE2:  SEL RB0
0424 01A1 7469         CALL TELEIN     ;READ BYTE FROM TAPE
0425 01A3 34AF         CALL TAPINC     ;COMMON
0426 01A5 E9A0         DJNZ R1,TALE2
0427 01A7 D5           SEL RB1
0428 01A8 E8A0         DJNZ R0,TALE2
0429 01AA C5           SEL RB0
0430 01AB 7469         CALL TELEIN     ;READ CHECKSUM
0431 01AD 2492         JMP CHECK
0432                   ;
0433 01AF 2F   TAPINC: XCH A,R7
0434 01B0 6F           ADD A,R7        ;ADD NEW BYTE TO OLD CHECKSUM
0435 01B1 2F           XCH A,R7        ;NEW CHECKSUM
0436 01B2 541F         CALL OUTDSP     ;OUTPUT TO DISPLAY
0437 01B4 4483         JMP PUTM        ;STORE IN MEMORY
0438                   ;
0439 01B6 541F OC:     CALL OUTDSP
0440                   ;
0441 01B8 27   CASCOM: CLR A
0442 01B9 A8           MOV R0,A
0443 01BA AA           MOV R2,A
0444 01BB BD08         MOV R5,#8
0445 01BD 546F         CALL DISPAD
0446 01BF 54B4         CALL INPADR     ;GET END ADDRESS
0447 01C1 FA           MOV A,R2
0448 01C2 17           INC A
0449 01C3 A9           MOV R1,A
0450 01C4 FB           MOV A,R3
0451 01C5 03F8         ADD A,#-8
0452 01C7 F2D1         JB7 ERR1
0453 01C9 D5           SEL RB1
0454 01CA 17           INC A
0455 01CB A8           MOV R0,A
0456 01CC C5           SEL RB0
0457 01CD 27           CLR A
0458 01CE AF           MOV R7,A        ;CLEAR CHECKSUM
0459 01CF 6419         JMP TR2         ;INITIALIZE TIMER
0460                   ;
0461 01D1 0488 ERR1:   JMP ERROR
0462                   ;*************************************************
0463                   ;       PRODUCTION TEST ROUTINES FOR TESTING MEMORY
0464                   ;       KEYPAD AND DISPLAYS FOR PROPER FUNCTIONING
0465                   ;*************************************************
0466 01D3 5413 PRODTST: CALL CLEAR
0467 01D5 2308         MOV A,#8
0468 01D7 BA10 PROD1:  MOV R2,#16
0469 01D9 AB           MOV R3,A        ;SAVE
0470 01DA 5448         CALL DISPDG
0471 01DC FB           MOV A,R3
0472 01DD EAD9         DJNZ R2,$-4     ;FILL DISPLAY WITH SAME CHAR
```

```
0473 01DF 5434 PROD2:  CALL KEYINP      ;READ KEYPAD
0474 01E1 96D7         JNZ PROD1
0475                   ;
0476 01E3 5413 MEMTEST: CALL CLEAR
0477 01E5 B8FD         MOV R0,#0FDH
0478 01E7 BD0F         MOV R5,#0FH      ;TOP OF VALID MEMORY
0479 01E9 5413 NEWAD:  CALL CLEAR
0480 01EB 5473         CALL DISPAD+4
0481 01ED 54AE         CALL BANKER      ;BANK P2 TO ADDRESS IN R0,R5
0482 01EF FA   LOOP:   MOV A,R2
0483 01F0 90           MOVX @R0,A       ;OUTPUT CHAR
0484 01F1 80           MOVX A,@R0       ;INPUT CHAR
0485 01F2 AB           MOV R3,A         ;SAVE
0486 01F3 DA           XRL A,R2         ;IS IT SAME?
0487 01F4 C6F8         JZ $+4
0488 01F6 64E8         JMP MEMERR
0489 01F8 EAEF         DJNZ R2,LOOP     ;TRY NEXT PATTERN
0490 01FA E8E9         DJNZ R0,NEWAD    ;BACK UP 1 AND CONTINUE
0491 01FC CD           DEC R5           ;PREVIOUS PAGE
0492 01FD 24E9         JMP NEWAD
0493                   ;
0494 0200              PAGE
0495                   ;
0496                   ;*****************************************************
0497                   ;      SYNCST:- GENERATES CONTINOUS CASETTE SYNC BYTES
0498                   ;*****************************************************
0499 0200 55   SYNCST: STRT T
0500 0201 A5           CLR F1
0501 0202 BEE6         MOV R6,#0E6H
0502 0204 741F         CALL CASOUT
0503 0206 4402         JMP $-4
0504 0208 8A0F CTR8279: ORL P2,#15      ;BANK FOR 8279
0505 020A D5           SEL RB1
0506 020B B9FF         MOV R1,#0FFH     ;CONTROL PORT
0507 020D 91           MOVX @R1,A       ;OUTPUT COMMAND
0508 020E 81   CTR1:   MOVX A,@R1       ;READ STATUS
0509 020F F20E         JB7 CTR1         ;WAIT FOR DISPLAY AVAILABLE(FOR CLEAR)
0510 0211 C5           SEL RB0
0511 0212 83           RET
0512                   ;
0513 0213 23D0 CLEAR:  MOV A,#CLEARD    ;CLEAR DISPLAY
0514 0215 5408         CALL CTR8279
0515 0217 2308         MOV A,#8
0516 0219 5408         CALL CTR8279     ;16 DIG DISPLAY
0517 021B 2390         MOV A,#DISP0     ;DISPLAY POS 0
0518 021D 4408         JMP CTR8279
0519                   ;
0520 021F D5   OUTDSP: SEL RB1
0521 0220 8A0F         ORL P2,#0FH      ;MEMORY MAP TP DISPLAY
0522 0222 B9FE         MOV R1,#0FEH     ;DATA PORT
0523 0224 91           MOVX @R1,A       ;OUTPUT TO DISPLAY
0524 0225 C5           SEL RB0
0525 0226 83           RET
0526                   ;
0527 0227 541F KO:     CALL OUTDSP
0528                   ;
 529 0229 5434 KEYIN:  CALL KEYINP      ;READ KEY
0530 022B 03E9         ADD A,#-17H      ;CLEAR?
0531 022D 9631         JNZ KEY2         ;NO
0532 022F 0461         JMP CMD          ;YES
```

```
0533 0231 0317 KEY2:    ADD A,#17H
0534 0233 83            RET
0535 0234 2340 KEYINP:  MOV A,#FIFORD   ;READ FROM FIFO COMMAND
0536 0236 5408          CALL CTR8279
0537 0238 D5            SEL RB1
0538 0239 81    KEY1:   MOVX A,@R1      ;STATUS
0539 023A 5307          ANL A,#FIFOCNT  ;ANY KEY PRESSED YET?
0540 023C C639          JZ KEY1         ;NO
0541 023E C9            DEC R1          ;DATA PORT
0542 023F 81            MOVX A,@R1      ;READ KEY
0543 0240 C5            SEL RB0
0544 0241 83            RET
0545                    ;
0546 0242 2324 INITIO:  MOV A,#24H      ;CLOCK PRESET
0547 0244 5408          CALL CTR8279
0548 0246 4413          JMP CLEAR
0549                    ;
0550 0248 5462 DISPDG:  CALL DSG        ;GET DISPLAY CODE
0551 024A 441F DISPDG1: JMP OUTDSP      ;DISPLAY IT
0552                    ;
0553 024C 5462 DISPDGP: CALL DSG        ;GET DISPLAY CODE
0554 024E 4380          ORL A,#PERIOD   ;ADD PERIOD BIT
0555 0250 444A          JMP DISPDG1     ;GO DISPLAY IT
0556                    ;
0557                    ;
0558 0252 3F    DISPTB: D0
0559 0253 06            D1
0560 0254 5B            D2
0561 0255 4F            D3
0562 0256 66            D4
0563 0257 6D            D5
0564 0258 7D            D6
0565 0259 07            D7
0566 025A 7F            D8
0567 025B 67            D9
0568 025C 77            DAA
0569 025D 7C            DBB
0570 025E 39            DC
0571 025F 5E            DD
0572 0260 79            DE
0573 0261 71            DF
0574 0262 530F DSG:     ANL A,#0FH
0575 0264 0352          ADD A,#DISPTB
0576 0266 A3            MOVP A,@A
0577 0267 83            RET
0578                    ;
0579 0268 AF    DISPRG: MOV R7,A        ;DISPLAY A REGISTER
0580 0269 47            SWAP A          ;GET HIGH BYTE FIRST
0581 026A 5448          CALL DISPDG     ;DISPLAY DIGIT
0582 026C FF            MOV A,R7        ;RETRIEVE LOW NIBBLE
0583 026D 444C          JMP DISPDGP     ;DISPLAY WITH PERIOD
0584                    ;
0585    = 026F DISPAD   =       $       ;DISPLAY ADDRESS WITHIN BLANKS
0586 026F 2392          MOV A,#DISP2
0587 0271 5408          CALL CTR8279
0588 0273 FD            MOV A,R5        ;PC8-11
0589 0274 5448          CALL DISPDG     ;DISPLAY DIGIT
0590 0276 F8            MOV A,R0
0591 0277 5468          CALL DISPRG     ;DISPLAY REGISTER
0592 0279 27            CLR A
```

```
0593 027A D5            SEL RB1
0594 027B 91            MOVX @R1,A
0595 027C 91            MOVX @R1,A
0596 027D 91            MOVX @R1,A
0597 027E 91            MOVX @R1,A      ;BLANK TO END OF DISPLAY
0598 027F 2396          MOV A,#DISP6
0599 0281 4408          JMP CTR8279
0600                    ;
0601    = 0283 PUTM     =       $
0602 0283 FC            MOV A,R4        ;MEM FLAG
0603 0284 F290          JB7 PUTM1       ;REGISTER MEMORY
0604 0286 54AE PUTM3:   CALL BANKER     ;EXTERNAL MEM BANKING
0605 0288 FE            MOV A,R6        ;GET BYTE TO STORE
0606 0289 90            MOVX @R0,A      ;STORE IT
0607 028A 18    BUMPPC: INC R0
0608 028B F8            MOV A,R0
0609 028C 968F          JNZ $+3
0610 028E 1D            INC R5          ;IF CARRY NEEDED
0611 028F 83            RET
0612 0290 F8    PUTM1:  MOV A,R0        ;GET ADDR0-7
0613 0291 03E6          ADD A,#-1AH     ;WHERE IS IT
0614 0293 F286          JB7 PUTM3       ;IN EXTERNAL MEMORY
0615 0295 FE            MOV A,R6        ;GET BYTE
0616 0296 A0            MOV @R0,A       ;STORE INTERNAL
0617 0297 448A          JMP BUMPPC      ;BUMP PC AND RETURN
0618                    ;
0619    = 0299 GETM     =       $
0620 0299 FC            MOV A,R4        ;MEM FLAG
0621 029A F2A0          JB7 GETM1       ;REG MEMORY
0622 029C 54AE GETM3:   CALL BANKER     ;BANK FOR EXTERNAL MEM
0623 029E 80            MOVX A,@R0
0624 029F 83            RET
0625 02A0 F8    GETM1:  MOV A,R0        ;ADDR0-7
0626 02A1 03E6          ADD A,#-1AH     ;WHERE IS IT
0627 02A3 F2A7          JB7 GETM2
0628 02A5 F0            MOV A,@R0       ;GET IT FROM INTERNAL MEMORY
0629 02A6 83            RET
0630 02A7 03FE GETM2:   ADD A,#-2       ;PT AT EXTERNAL ADDR
0631 02A9 A8            MOV R0,A
0632 02AA BD0F          MOV R5,#0FH
0633 02AC 449C          JMP GETM3
0634                    ;
0635    = 02AE BANKER   = $             ;BANK TO PC8-11,PC0-7 IN R0
0636 02AE 0A            IN A,P2         ;READ EXISTING P2
0637 02AF 53F0          ANL A,#0F0H     ;REMOVE BANK BITS ONLY
0638 02B1 4D            ORL A,R5        ;MERGE BANK BITS WITH P2 HIGH NIBBLE
0639 02B2 3A            OUTL P2,A       ;PERFORM BANKING
0640 02B3 83            RET
0641    = 02B4 INPADR = $
0642 02B4 54E6          CALL GTVALD     ;GET VALID HEX DATA
0643 02B6 F6C7          JC INPEND       ;TERM . OR ,
0644 02B8 2A            XCH A,R2        ;KA 0-7
0645 02B9 47            SWAP A
0646 02BA AB            MOV R3,A
0647 02BB 53F0          ANL A,#0F0H
0648 02BD 4A            ORL A,R2
0649 02BE AA            MOV R2,A        ;LEFT SHIFT KA0-11 AND MERGE NEW DIGIT
0650 02BF 230F          MOV A,#0FH
0651 02C1 5B            ANL A,R3
0652 02C2 AB            MOV R3,A        ;KA8-11
```

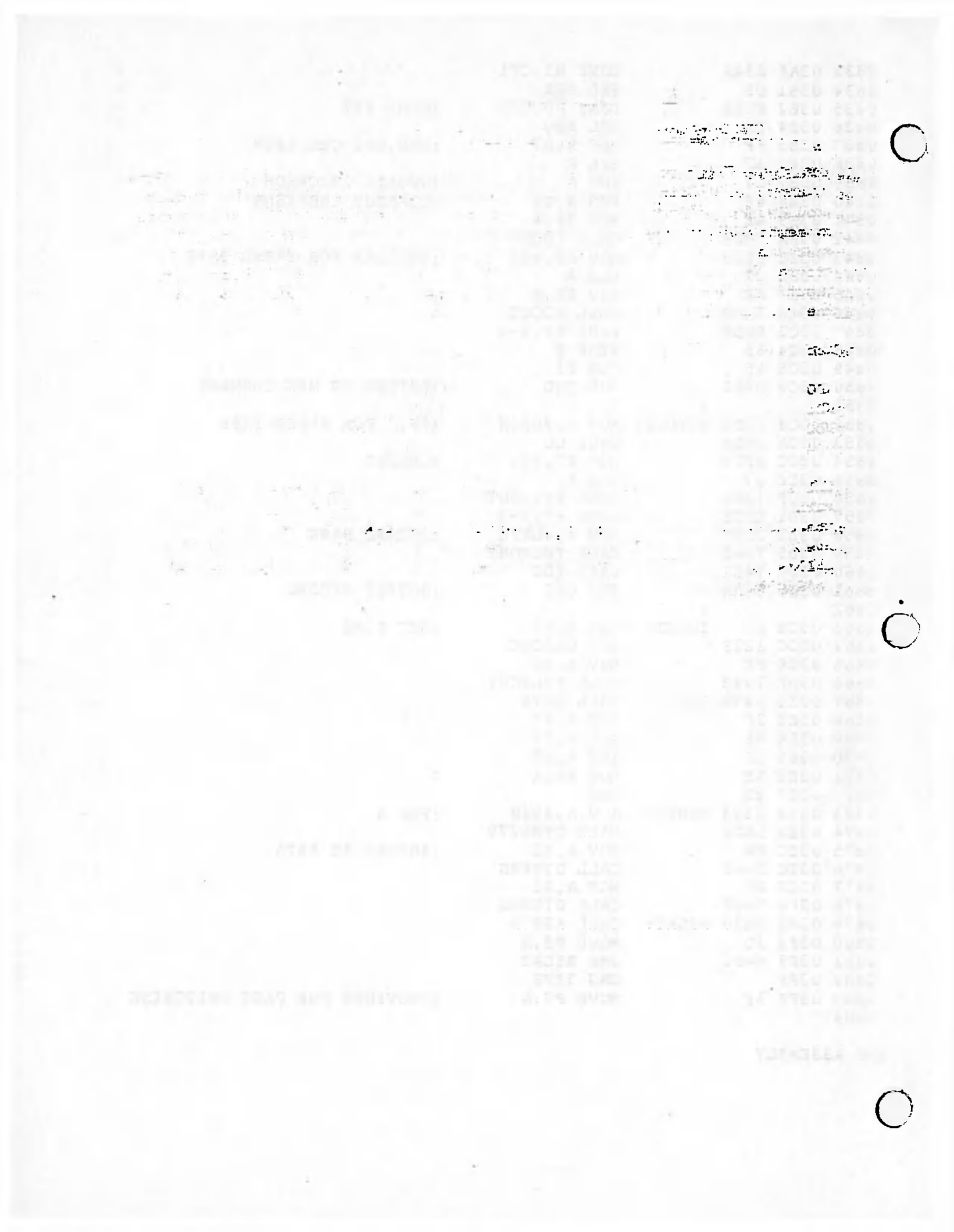
```
0653 02C3 54C9          CALL DISPKA     ;DISPLAY KA IN POS 5-7.
0654 02C5 44B4          JMP INPADR      ;GET ANOTHER
0655 02C7 67    INPEND: RRC A           ;C=0="NEXT",C=1="ENTER"
0656 02C8 83            RET
0657 02C9 2396 DISPKA:  MOV A,#DISP6
0658 02CB 5408          CALL CTR8279    ;POS 5 SETUP
0659 02CD FB            MOV A,R3        ;KA8-11
0660 02CE 5448          CALL DISPDG
0661 02D0 FA            MOV A,R2
0662 02D1 4468          JMP DISPRG
0663 02D3 8A0F FNDST:   ORL P2,#15
0664 02D5 B9E3          MOV R1,#0E3H    ;STACK PTR PTR
0665 02D7 81            MOVX A,@R1      ;STACK PTR
0666 02D8 5307          ANL A,#7
0667 02DA 97            CLR C           ;FOR RLC A
0668 02DB 96DF          JNZ FNDST1      ;NO ROTATE STACK
0669 02DD 2308          MOV A,#8        ;TOP OF STACK
0670 02DF 07    FNDST1: DEC A
0671 02E0 F7            RLC A           ;*2BYTES/ENTRY
0672 02E1 03EC          ADD A,#0ECH     ;BASE PTR OF EXT STORAGE
0673 02E3 A9            MOV R1,A
0674 02E4 81            MOVX A,@R1      ;GET PC0-7
0675 02E5 83            RET
0676                    ;
0677 02E6 5429 GTVALD:  CALL KEYIN      ;READ KEYPAD
0678 02E8 03F0          ADD A,#-16
0679 02EA F2F0          JB7 GT1         ;VALID HEX DIGIT
0680 02EC 03F9          ADD A,#-7       ;SET CARRY AND A=FF IF ENTER,A=FE IF NE
0681 02EE A7            CPL C           ;SET CARRY FOR TERMINATE
0682 02EF 83            RET
0683 02F0 0310 GT1:     ADD A,#16       ;RETSTORE VALUE
0684 02F2 97            CLR C
0685 02F3 83            RET
0686                    ;
0687 02F4 62    TEND1:  MOV T,A
0688 02F5 00            NOP
0689 02F6 00            NOP
0690 02F7 00            NOP
0691 02F8 00            NOP
0692 02F9 55            STRT T
0693 02FA 2308          MOV A,#8
0694 02FC 83            RET
0695                    ;
0696 0300               PAGE
0697                    ;*************************************
0698                    ;       CASETTE INPUT BYTE (TARBEL)
0699                    ;*************************************
0700                    ;
0701 0300 BA07 CASIN:   MOV R2,#7       ;BIT COUNT-1
0702 0302 7406          CALL BITIN
0703 0304 EA02          DJNZ R2,CASIN+2
0704 0306 160A BITIN:   JTF BIT1        ;WAIT FOR TIMEOUT
0705 0308 6406          JMP BITIN
0706 030A 0E    BIT1:   MOVD A,P6       ;GET STATE OF LINE
0707 030B AB            MOV R3,A        ;SAVE IT
0708 030C 47            SWAP A          ;PUT IN BIT 7
0709 030D F7            RLC A           ;INTO CARRY
0710 030E B611          JF0 NOCOMP      ;BRIF NON-INVERT
0711 0310 A7            CPL C           ;INVERT
0712 0311 FE    NOCOMP: MOV A,R6        ;GET ACCUMULATED BYTE
```

```
0713 0312 67          RRC A           ;INCLUDE NEW BIT
0714 0313 AE          MOV R6,A        ;SAVE BYTE
0715 0314 0E   TR1:   MOVD A,P6       ;WAIT FOR TRANSITION
0716 0315 37          CPL A           ;INVERT BITS
0717 0316 DB          XRL A,R3        ;TRANSITION YET?
0718 0317 7214        JB3 TR1         ;NO
0719 0319 23FD TR2:   MOV A,#-3       ;INITIALIZE TIMER
0720 031B 7449        CALL TIMEND
0721 031D FE          MOV A,R6        ;GET BYTE BACK
0722 031E 83          RET
0723                  ;
0724                  ;***********************************************
0725                  ;       CASETTE OUTPUT BYTE (TARBEL)
0726                  ;***********************************************
0727 031F BA08 CASOUT: MOV R2,#8      ;#BITS
0728 0321 FE          MOV A,R6
0729 0322 67          RRC A
0730 0323 AE          MOV R6,A        ;SAVE BYTE
0731 0324 7443        CALL TIMEOUT
0732 0326 F632        JC CASOUT1
0733 0328 07          DEC A
0734 0329 9F          ANLD P7,A
0735 032A 7439        CALL PTIMEOUT
0736 032C 8F          ORLD P7,A       ;FORCE HIGH
0737 032D FE   CASOUT2: MOV A,R6
0738 032E EA22        DJNZ R2,CASOUT+3 ;GET NEXT BIT
0739 0330 67          RRC A           ;RESTORE A
0740 0331 83          RET
0741 0332 8F   CASOUT1: ORLD P7,A     ;FORCE HIGH
0742 0333 7439        CALL PTIMEOUT
0743 0335 07          DEC A           ;A=7
0744 0336 9F          ANLD P7,A       ;FORCE LOW
0745 0337 642D        JMP CASOUT2
0746                  ;
0747 0339 FA   PTIMEOUT: MOV A,R2
0748 033A 07          DEC A
0749 033B 9643        JNZ TIMEOUT
0750 033D 7641        JF1 PTIME1
0751 033F 6443        JMP TIMEOUT
0752 0341 74E1 PTIME1: CALL IOI
0753                  ;
0754 0343 23FE TIMEOUT: MOV A,#-2     ;
0755 0345 1649        JTF TIMEND
0756 0347 6445        JMP TIMEOUT+2
0757 0349 44F4 TIMEND: JMP TEND1
0758                  ;
0759                  ;***********************************************
0760                  ;       TELETYPE OUTPUT DRIVER 110 BAUD
0761                  ;***********************************************
0762 034B AE   TELEOUT: MOV R6,A      ;SAVE
0763 034C BA0B        MOV R2,#11      ;BIT COUNT
0764 034E 97          CLR C           ;FOR START BIT
0765 034F 1653 TELE1:  JTF TELE2
0766 0351 644F        JMP $-2 ;WAIT FOR TIMEOUT
0767 0353 E65A TELE2:  JNC BITOUT1    ;OUTPUT 0 BIT
0768 0355 230B        MOV A,#0BH      ;OUTPUT INVERTED 1 BIT
0769 0357 9F          ANLD P7,A       ;PORT 7 BIT 2
0770 0358 645D        JMP BITOUT3
0771 035A 2304 BITOUT1: MOV A,#4
0772 035C 8F          ORLD P7,A
```

```
0773 035D 97    BITOUT3: CLR C
0774 035E A7             CPL C              ;FOR STOP BITS
0775 035F FE             MOV A,R6
0776 0360 67             RRC A
0777 0361 AE             MOV R6,A
0778 0362 23BD           MOV A,#-67         ;FOR TIMER FOR 9040 USEC DELAY
0779 0364 7449           CALL TIMEND        ;SET TIMER AND START
0780 0366 EA4F           DJNZ R2,TELE1      ;GET NEXT BIT IF ANY
0781 0368 83             RET
0782                     ;*******************************************
0783                     ;       TELETYPE INPUT ROUTINE 110 BAUD
0784                     ;*******************************************
0785 0369 BA08 TELEIN:   MOV R2,#8          ;8 BITS/BYTE
0786 036B 0E    T1:      MOVD A,P6          ;READ LINE
0787 036C 526B           JB2 T1
0788 036E 23DE           MOV A,#-34         ;HALF BIT TIME
0789 0370 7445           CALL TIMEOUT+2     ;START CLOCK
0790 0372 23BD           MOV A,#-67         ;
0791 0374 7445           CALL TIMEOUT+2
0792 0376 0E             MOVD A,P6          ;READ LINE
0793 0377 526B           JB2 T1             ;FALSE START BITE
0794 0379 23BD T3:       MOV A,#-67         ;
0795 037B 7445           CALL TIMEOUT+2     ;1 BIT IMTE
0796 037D 0E             MOVD A,P6
0797 037E 67             RRC A
0798 037F 67             RRC A
0799 0380 67             RRC A    ;BIT IN CARRY
0800 0381 FE             MOV A,R6
0801 0382 67             RRC A              ;MERGE NEW BIT
0802 0383 AE             MOV R6,A           ;SAVE
0803 0384 EA79           DJNZ R2,T3         ;GET NEXT BIT
0804 0386 7443           CALL TIMEOUT       ;WAIT FOR STOP BIT
0805 0388 FE             MOV A,R6           ;RETRIVE BYTE
0806 0389 83             RET
0807                     ;
0808                     ;*****************************************************
0809                     ;       TAPEOUT- OUTPUT BINARY TAPE TO CASETTE OR PTAPE
0810                     ;*****************************************************
0811 038A 233F TPOUT:    MOV A,#03FH        ;"O"
0812 038C 5427           CALL KO
0813 038E 03F1           ADD A,#-0FH        ;IS IT TTY COMMAND?
0814 0390 C6C8           JZ TTYOUT          ;YES
0815 0392 1C             INC R4             ;R4=1=CASETTE,R4=0=TELETYPE
0816                     ; CASETTE OUTPUT HERE
0817 0393 23B9           MOV A,#DC+PERIOD          ;"C."
0818 0395 34B6           CALL OC
0819 0397 BB05           MOV R3,#5
0820 0399 27    CT2:     CLR A
0821 039A AE             MOV R6,A
0822 039B 741F           CALL CASOUT
0823 039D EF99           DJNZ R7,CT2        ;256 ZEROS OUT
0824 039F EB99           DJNZ R3,CT2        ;5 TIMES 256 ZEROS ABOUT 7 SECS
0825 03A1 BEC3           MOV R6,#0C3H
0826 03A3 741F           CALL CASOUT        ;START BYTE FOR BIT SYNC
0827 03A5 BEE6           MOV R6,#0E6H
0828 03A7 B5             CPL F1
0829 03A8 741F           CALL CASOUT        ;SYNC BYTE
0830 03AA C5    CT1:     SEL RB0
0831 03AB 548A           CALL BUMPPC
0832 03AD 74DB           CALL IOOUT         ;OUTPUT TO  DEVICE
```

```
0833 03AF E9AA          DJNZ R1,CT1
0834 03B1 D5            SEL RB1
0835 03B2 E8AA          DJNZ R0,CT1     ;DONE YET
0836 03B4 C5            SEL RB0
0837 03B5 FF            MOV A,R7        ;YES,GET CHECKSUM
0838 03B6 37            CPL A
0839 03B7 17            INC A           ;NEGATE CHECKSUM
0840 03B8 6E            ADD A,R6        ;CORRECT CHECKSUM
0841 03B9 AE            MOV R6,A
0842 03BA 74DB          CALL IOOUT
0843 03BC BB28          MOV R3,#40      ;TRAILER FOR PAPER TAPE
0844 03BE 27            CLR A
0845 03BF AE            MOV R6,A
0846 03C0 74DB          CALL IOOUT
0847 03C2 EBBE          DJNZ R3,$-4
0848 03C4 65            STOP T
0849 03C5 A5            CLR F1
0850 03C6 0461          JMP CMD         ;RETURN TO NEW COMMAND
0851                    ;
0852 03C8 23F3 TTYOUT:  MOV A,#0F3H     ;"P." FOR PAPER TAPE
0853 03CA 34B6          CALL OC
0854 03CC BF28          MOV R7,#40      ;LEADER
0855 03CE 27            CLR A
0856 03CF 744B          CALL TELEOUT
0857 03D1 EFCE          DJNZ R7,$-3
0858 03D3 23FF          MOV A,#0FFH     ;RECORD MARK
0859 03D5 744B          CALL TELEOUT
0860 03D7 74E1          CALL IO1
0861 03D9 64AA          JMP CT1         ;OUTPUT RECORD
0862                    ;
0863 03DB FC    IOOUT:  MOV A,R4        ;GET FLAG
0864 03DC 121F          JB0 CASOUT
0865 03DE FE            MOV A,R6
0866 03DF 744B          CALL TELEOUT
0867 03E1 5499 IO1:     CALL GETM
0868 03E3 2F            XCH A,R7
0869 03E4 6F            ADD A,R7
0870 03E5 2F            XCH A,R7
0871 03E6 AE            MOV R6,A
0872 03E7 83            RET
0873 03E8 2394 MEMERR:  MOV A,#94H      ;POS 4
0874 03EA 5408          CALL CTR8279
0875 03EC FA            MOV A,R2        ;SHOULD BE DATA
0876 03ED 5468          CALL DISPRG
0877 03EF FB            MOV A,R3
0878 03F0 5468          CALL DISPRG
0879 03F2 5429 RELAY:   CALL KEYIN
0880 03F4 3D            MOVD P5,A
0881 03F5 64F2          JMP RELAY
0882 03FF               ORG 3FFH
0883 03FF 3F            MOVD P7,A       ;PROVIDES FOR PAGE SWITCHING
0884

END ASSEMBLY.
```



APPENDIX C

USEFUL MONITOR SUBROUTINES

Several monitor subroutines may prove useful to the user. Some of these are defined and described below. The same procedure should be used when calling any of them, since they all reside in memory bank 0. This memory bank must be selected before executing a call to the subroutine. For example, if a user program residing at Location 800H wanted to call the teletype input-byte routine (TELEIN), which resides at location 366H, the calling sequence would look as follows:

Loc	Machine Inst	Description
800	E5	SEL MB0
801	55	STRT T
802	7466	CALL TELEIN
804	F5	SEL MB1

All subroutines require Rgister Bank 0 to be selected and return with Register Bank 0 selected. Many of the subroutines use and destroy 1 or more registers of RB0 and RB1 during their operations. These are noted in the descriptions which follow. For a complete description, see the monitor listing in Appendix B.

Subroutine	Location	Description
SYNCST:	=200H	Generate continuous sync bytes (E6H) to cassette port. Does not return. Must be reset to return.
CTR8279:	=208H	Output char in A-reg to control port of 8279 chip. REG USED: R1 of RB1, P2, A
CLEAR:	=213H	Clear display, reset for position 0 REG USED: P2, R1 of RB1
OUTDSP:	=21FH	Output A-reg to next display position REG USED: P2, R1 of RB1
KEYINP:	=234H	Read Keypad value 0-17H. Return value in A-reg. REG USED: R1 of RB1
DISPRG:	=268H	Display A-reg in Hex at next 2 chars of display. REG USED: R7, R1 of RB1, A, Carry
DISPAD:	=26FH	Display 12 bit address in R0 and R5 (R0 = bits 0-7, R5 = bits 8-11, R5 bits 4-7 unused) on display in position 2, 3, 4. REG USED: A, R1 of RB1, R7
BANKER:	=2AEH	Use R5 bits 0-3 to bank external memory on P2 bits 0-3
DISPKA:	=2C9H	Display 12 bit address R2, R3 in position 5, 7, 8 of display. R2 = bits 0-7. REG USED: R3 bits 0-3 = bits 8-11 of address A, R1 of RB1, R7
CASIN:	=300H	Read byte from cassette into A-reg. Requires that Timer be started prior to call. REG USED: R2, R3, R6
CASOUT:	=31FH	Output byte to cassette from A-reg. REG USED: R2, R6, Timer, A, F1
TELEOUT:	=34BH	Output A-reg to serial port. REG USED: R2, A, Carry, R6, Timer
TELEIN:	=369H	Input byte from serial port to A-reg REG USED: R2, Carry, R6

APPENDIX D

Change Page

SELF DIAGNOSTIC

Included at location 1D3H is a production test routine to test all present memory, the Keypad and the display. It can be entered by using the command

[EXEC] [1D3] [↵] [.]

There are 2 modes to this routine: Memory test and Display/ Keypad test. The Display/Keypad mode is activated when first entered. In this mode any key pressed on the keypad will be displayed on all the displays. This will indicate bad display segments and/or faulty keypad contacts.

Hitting the Zero key causes the second mode to be entered. Starting at memory location FFDH and working down through memory, all cells are checked for every bit pattern. If a difference between the pattern written and the pattern read is detected, the program stops with the following display;

A	A	A	b	W	W.	I	I.	b

where AAA = address of incorrect byte
WW. = pattern that was put out
II = pattern that is read in

The algorithm for cycling through memory is such that if only the top 1K is present the following will appear, indicating memory good:

B	Ø	Ø	b	Ø	Ø.	F	F	b

If all 2K is present, the following will indicate memory good:

7	Ø	Ø	b	F	F.	Ø	Ø.	b

Any other display will indicate bad memory, or lack of memory in top 1K position.

As soon as the memory display terminates, the relay test begins. This test opens and closes the 4 relays based on what is input from the keypad.

The four lower bits read from the keypad are output to the relays. Audible click indicates a working relay. An output of 0 closes a relay; an output of 1 opens it. To close each relay the

following keys should be input

[E] [C] [8] [Ø]

Four separate and distinct clicks should be audible as each relay closes. Then the following keys allow 1 relay at a time to open also resulting in audible clicks

[1] [3] [7] [F]

CUSTOMER SERVICE

REPLACEMENT PARTS

If you need a replacement part, use only standard parts from commercial sources. Use of surplus or second-run parts will void warranty. If you have trouble locating a part, write IMSAI and include:

- Part number and description as shown in the parts list.
- Serial number of cabinet or board name and revision number.
- Date of purchase.
- Nature of defect.

Note: Parts damaged through carelessness or misuse will not be replaced under warranty.

TECHNICAL CONSULTATION

Need help with your kit or system?

We encourage you to call or write IMSAI for assistance with any technical problems (except program debugging and "customizing" of hardware for special application, which we will not handle).

The effectiveness of our technical assistance depends on the information you furnish. Be sure to include:

- Serial number of cabinet and/or board name and revision number.
- Date of purchase.
- Exact description of problem.
- Everything you have done in attempting to correct the problem.
- All switch positions, connections to other equipment, system configuration, operation procedure, voltage readings and any other information that you think might be useful.

Note: Telephone traffic is lightest at midweek . . . please be sure your manual and all notes are on hand at time of call.

REPAIR SERVICE

Service facilities are available for both warranty and non-warranty repair work. If this service is desired, send IMSAI:

- Name and address.
- Date of purchase.
- Copies of all correspondence and notes relevant to the problem.
- A complete description of the problem.
- Authorization to return your kit C.O.D. for service (IF ANY) and shipping charges.
- The equipment to be repaired should be sent to IMSAI well packed.
- The original packing slip number.

ERRATA INFORMATION

Errata information will be found immediately preceeding the section to which the information applies and should be used for clarification and/or correction of the section indicated.

CAUTION: FAILURE TO OBSERVE PERTINENT INFORMATION WHICH IS INCLUDED WILL VOID WARRANTY.

8048CC ROM, Rev. 1
8048CC EROM, Rev. 1
Errata
EDN 77-0003
6/7/77

ERRATA

Your 8048CC EROM or ROM assembly has sockets provided for 2K of RAM. These sixteen 16 pin sockets should be installed at locations U18 through U33 for use with the 2102 chips.

8048CC EROM Rev. 1
Errata
6/10/77

IMPORTANT

Enclosed are sixteen 16 pin sockets for use with the 2102 chips on your 8048CC EROM Assembly (8 for the eight 2102's provided with your kit and 8 for the Expander Module). If you choose to use them, they should be installed at locations U18 through U33 (refer to Step 19 of the Assembly Instructions).

8048CC
Errata
ECO 77-0108
9/21/77

ERRATA

To ensure the proper functioning of your kit, please note the following changes in your documentation.

1) On page I-7, item 5, U18-U25 should be U26-U33.

2) In the Assembly Instructions Step 19 (page III-3), disregard the present instruction and insert the following:

19a ( ) Install and solder the sixteen 16-pin sockets at the indicated locations.

( ) U18	( ) U22	( ) U26	( ) U30
( ) U19	( ) U23	( ) U27	( ) U31
( ) U20	( ) U24	( ) U28	( ) U32
( ) U21	( ) U25	( ) U29	( ) U33

19b ( ) Install the eight 2102 IC's in the sockets at the indicated locations.

( ) U26	( ) U30
( ) U27	( ) U31
( ) U28	( ) U32
( ) U29	( ) U33

19c ( ) If the Expander Module Kit is used, install the other eight 2102 IC's at the indicated locations.

( ) U18	( ) U22
( ) U19	( ) U23
( ) U20	( ) U24
( ) U21	( ) U25

3) In the Assembly Instructions Step 40 (page III-4 and 5), install and solder ALL 23 .1uF CAPACITORS. All twenty-three should be installed whether or not the Expander Module Kit is to be used.

ERRATA (5/20/77)

Schematic Diagram

On U18 pin 13 was R/W. Change to /CE.

Also on U18 pin 3 was /CE. It is now R/W.

C8 is now 470 pf

C4 is deleted

C9 is now .1 uf

Assembly Diagram

C8 is now 470 pf

C4 is deleted

C9 is now .1 uf

Assembly Instructions

An additional capacitor is provided, .47 uf, which may be used to replace C9 if a high frequency response tape recorder is used.

8048CC
9/21/77

IMPORTANT

The IMP-48 Monitor program may be supplied in either 2716 EPROM or 2716HTP PROM. These are 2K byte devices, and the 1K monitor may be programmed into either the upper or lower half. Your PROM/EPROM chip is labeled "HIGH" or "LOW" as appropriate.

If your PROM/EPROM is labeled "LOW":

a) Install a jumper at location "JC".

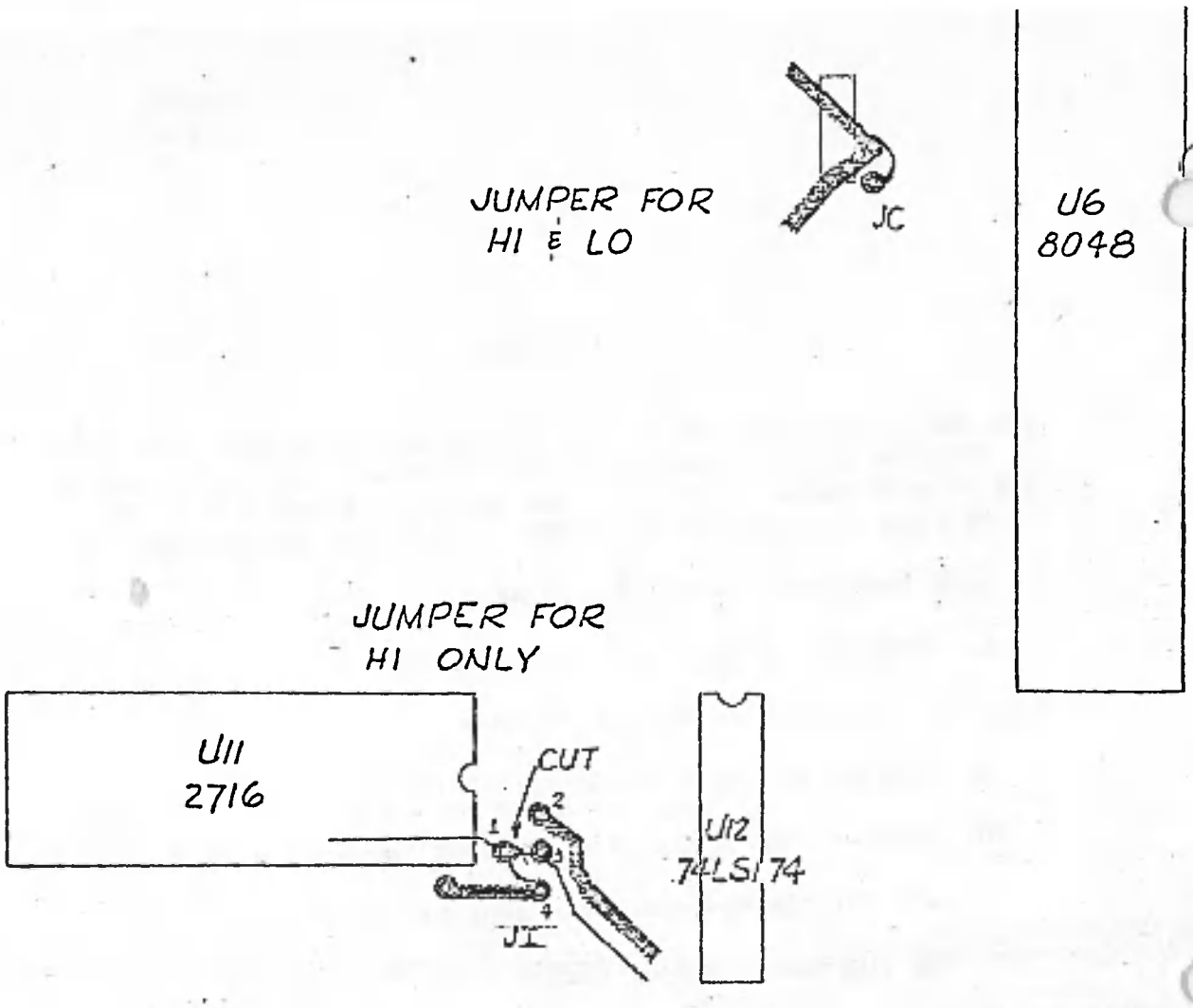
If your PROM/EPROM is labeled "HIGH":

a) Install a jumper at location "JC".

b) Refer to the enclosed drawing of jumper area "JI" and:

1) cut the trace between pads "1" and "3"

2) install a jumper between pads "1" and "4".



3048 CC JUMPERS
COMPONENT SIDE