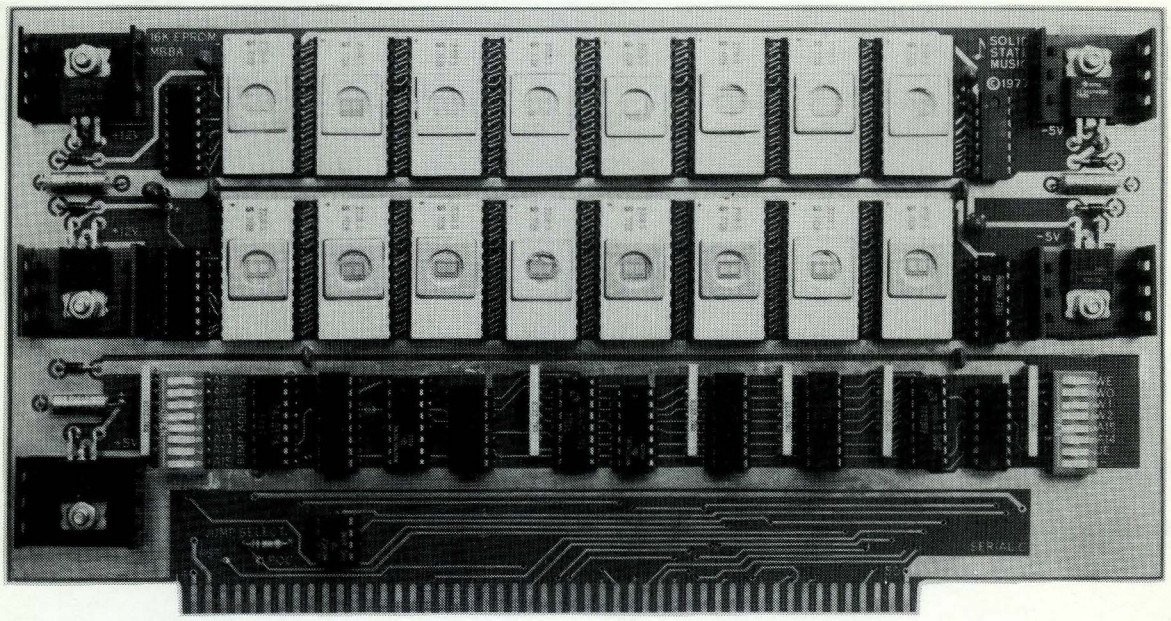
Microcomputer Products, 2116 Walsh Avenue, Santa Clara, CA 95050, (408) 246-2707

MB8A 1K TO 16K EPROM BOARD



FEATURES:

SYSTEM COMPATIBILITY
- S-100 bus computer systems.

MEMORY
- Up to 16K bytes of 2708 EPROMs (not included)
- Any unused EPROM socket will automatically disable the board for that 1K increment. For example, with 8 EPROMs it acts as an 8K board, taking up only 8K of memory address space.

ADDRESSING
- DIP switch selection of memory address assignment in 16K byte increments.
- Magic Mapping ™ allows any byte within ROM to be mixed with any similarly addressed RAM board equipped with Phantom Disable.

VECTOR JUMP
- Power-on/reset vector jump to any 256 byte increment; DIP switch addressable.
- Vector jump can be disabled.
- Vector jump requires other memory boards to be equipped with Phantom Disable.

OTHER FEATURES
- DIP switch selection of 0 to 8 wait state clock cycles, so fast or slow EPROMs can be used.
- All lines buffered, Reverse voltage protection.
- High grade glass epoxy PC board with gold plated edge connector contacts.
- Low profile sockets provided for all ICs.
- Power requirements (less EPROMs) -- +8V @ 160mA, +16V @ 10mA, -16V @ 10mA typical.

We used to be Solid State Music. We still make the blue boards.

C O N T E N T S

SSM MICROCOMPUTER PRODUCTS

2116 Walsh Ave.
Santa Clara, California

MB8-A - 8K/16K EPROM BOARD

1.0 ASSEMBLY INSTRUCTIONS (refer to figure 1)

- [ ] Check kit contents against parts list.
- [ ] Check PC board for possible warpage and straighten if required.
- [ ] Insert 16 24-pin, 9 16-pin, 5 14-pin, and 1 8-pin socket into the component side of the board with the "pin 1" index toward the top of the board. (The component side is the side on which "Solid State Music" is printed.)
- [ ] Place a flat piece of stiff cardboard of appropriate size on top of the sockets to hold them in place.
- [ ] Holding the cardboard in place against the sockets, turn the board over and lay it on a flat surface. (Be sure that all of the socket pins are through the holes.)

  Note: Keep soldering iron tip clean to prevent rosin and sludge from being deposited on traces. Wipe tip frequently on a damp cloth or steel wool.
- [ ] On each socket, solder two of the corner pins, choosing two that are diagonally opposite of each other.
- [ ] Once the sockets are secured, lift the board and check to see if they are flat against the board. If not, seat the sockets by pressing on the top while reheating each soldered pin.
- [ ] Complete soldering the remaining pins of each socket. Touch pin and pad with iron tip, allowing enough solder to flow to form a filet between pin and pad. Keep the tip against the pin and pad just long enough to produce the filet. Too much heat can cause separation of pad and trace from the board. A 600 degree iron is recommended.
- [ ] Insert and solder 6 2.7K SIPs and 1 10K resistor.
- [ ] Insert and solder 5 diodes (observing polarity).
- [ ] Insert and solder 4 0.1uF ceramic capacitors.
- [ ] Observing polarity, insert and solder 4 dipped & 3 tubular tantalum capacitors.

Insert 2 DIP switches with the word "OPEN" toward the left of the board.

Place regulators on the board so the mounting hole in the regulator is in line with the hole in the board. Mark leads for proper bending to match the board holes - allow for a bend radius.

Bend regulator leads to match holes in board.

If available, apply thermal compound to the back side of each regulator case (the side that will contact the heat sink). Use just a little thermal compound. Too much is worse than none at all.

On the front (component side) of the board, position 2 heatsink insulators on the U12 and U24 regulator locations*. Next position heatsink and insert regulator for each of the 5 regulators. Finally, position nut and lockwasher on top of regulator and secure from behind with screw in each case. Be sure regulators and heatsinks fit flat on board and then solder all regulator leads.

2.0 FUNCTIONAL CHECK

WARNING! DO NOT INSTALL OR REMOVE BOARD WITH POWER ON. DAMAGE TO THIS AND OTHER BOARDS COULD OCCUR.

Apply power (+8 volts approx.) to board by plugging into computer or by connection to a suitable power supply. Measure the output of U25. If less than 4.8 volts is measured (allowing for meter accuracy) check for shorts or wiring errors. CAUTION: WHILE IT HAS NEVER HAPPENED TO US, SHORTED REGULATORS HAVE BEEN KNOWN TO EXPLODE WITH POSSIBLE INJURY TO EYES OR HANDS. BETTER SAFE THAN SORRY - KEEP FACE AND HANDS CLEAR OF THE REGULATOR SIDE OF THE BOARD DURING THIS AND SUBSEQUENT TESTS!

Apply power (+16 volts) to the board by plugging into computer or by connection to a suitable power supply. Verify that the outputs of U1 & U13 are between +11.5 & +12.5 Vdc.

Apply power (-16 volts) to the board by plugging into computer or by connection to a suitable power supply. Verify that the outputs of U12 & U24 are between -4.8 & -5.4 Vdc.

Finally, insert the ICs into their sockets, observing polarity.

Now, look the board over carefully. Check for poor solder joints or bridges. Using the component layout drawing, look for improper part location or polarity. A few minutes of careful inspection may save a few hours of troubleshooting.

* See Illustration on page 12.

3.0 SET-UP

3.1 Address Selection

The MB-8 card can be set to one of four possible address locations in 16K increments.

Starting Address	HIGH ORDER BITS A15	A14	SETTING OF S2 A15	A14
ØØØØ (Ø)	Ø	Ø	OPEN	OPEN
4ØØØ (16,384)	Ø	1	OPEN	CLOSED
8ØØØ (32,764)	1	Ø	CLOSED	OPEN
CØØØ (49,148)	1	1	CLOSED	CLOSED

note: open = off
closed = on

Even though the MB-8A card can support 16K of PROM, it can be disabled down to as small as one byte of PROM. No jumpers necessary for disable selection!

3.2 PROM Area Enable/Disable

A. 1K Increments
Where ever PROM is inserted into the MB-8A card, the board can enable at that address location, and the empty ROM sockets will be addresses that the card will disable itself. This will allow the user to have only 1K or 2K of active PROM area. This means you can place a RAM card at an address within the MB-8A's area if there is no PROM in the socket at that address.

B. Less than 1K increments.
If the user wants PROM areas on the MB-8A in fractions of 1K, it is possible.

Example.
Problem: Only a 512 byte program is needed for system operation, the user then wants the rest of his 16K bytes filled with RAM.

Solution: Address the MB-8A card so that it will overlap the wanted 512 byte program's address. Find the socket on the MB-8A card which is the 1K block that overlays the 512 bytes. Program a 2708 PROM for this socket with the 512 byte program, and any bytes that are not used shall be programmed with FF Hex. Now insert the MB-8A card with the one PROM into the computer and insert RAM at all addresses except where 512 bytes of ROM is located.

As you can see, any byte in ROM set to FF Hex will disable the MB-8A card.

3.3 Magic Mapping

This is the most interesting mode of the MB-8A card, but requires more software and care in implementing. The idea of magic mapping is the ability of having scratch RAM within the 1024 bytes of a 2708 ROM or have a couple of bytes (or more) of ROM right in the middle of a RAM card. If that sounds like what you have been looking for, then here is how to do it.

A. The RAM card that will be mixed with PROM must be equipped with Phantom disable. (Pin 67 on the bus going low will disable memory.)

B. Any area of a 2708 PROM you wanted to be changed into RAM must be programmed with FF Hex.

C. The area of RAM that a 2708 overlaps must be initialized to FF before the user program can be run. This initialization program can not have any FF bytes in its machine code, if it also overlaps RAM memory.

A simple initialization routine is listed below:

```
                ;THIS PROGRAM WILL FILL JUST ONE BLOCK OF
                ;MEMORY WITH FF HEX. WRITTEN BY MALCOLM WRIGHT,
                ;1-10-1978.

                ;"LOC" IS THE STARTING ADDRESS OF
                ;THE PROGRAM.
FC00            LOC     EQU     0FC00H
F021            XXXX    EQU     0F021H  ;TO USER'S MONITOR.

                ;DEMO.
4000            BYTES   EQU     4000H   ;NUMBER OF BYTES=16K
C000            START   EQU     0C000H  ;START ADDR.=TOP 16K

FC00                    ORG     LOC
                ;PLACE THE NUMBER OF BYTES TO BE CHANGED INTO B&C.
FC00 010040             LXI     B,BYTES ;* WARNING!
                ;PLACE THE STARTING ADDRESS INTO H&L.
FC03 2100C0             LXI     H,START ;* WARNING!
FC06 AF         NEXT:   XRA     A
FC07 2F                 CMA
FC08 77                 MOV     M,A
FC09 23                 INX     H
FC0A 0B                 DCX     B
FC0B 79                 MOV     A,C
FC0C B0                 ORA     B
FC0D C206FC             JNZ     NEXT    ;* WARNING!
                ;CHANGE XXXX TO THE ENTRY POINT OF YOUR MONITOR.
FC10 C321F0             JMP     XXXX

                ;*.....THIS JUMP SHOULD NOT HAVE ANY FF BYTES IN IT.
0000                    END
```

```
                ;THIS PROGRAM WILL FILL AREAS OF MEMORY WITH
                ;FF HEX AS SPECIFIED IN A ADDRESS TABLE SUPPLIED
                ;BY THE USER. WRITTEN BY MALCOLM WRIGHT,1-10-1976.

                ;"LOC" IS THE STARTING ADDRESS OF THE PROGRAM.
FC00            LOC     EQU     0FC00H
F021            XXXX    EQU     0F021H  ;TO USER'S MONITOR

FC00                    ORG     LOC

                ;THE ENTRY POINT OF THIS PROGRAM BEGINS WITH
                ;LXI H INSTRUCTION. THE TWO BYTES THAT ARE GOING
                ;INTO H&L CAN NOT BE FF HEX.
                ;IF EITHER BYTE IS FF , THEN TRY THIS:
                ;       MVI     H,0     ;IF REG.H IS FF.
                ;       DCR     H
                ;       MVI     L,LOW   ;SET LOW 1/2 OF ADDR.
                ;OR THIS:
                ;       MVI     H,HIGH  ;SET HIGH 1/2 OF ADDR.
                ;       MVI     L,0     ;IF REG.L IS FF.
                ;       DCR     L

FC00 212AFC             LXI     H,TABLE ;TABLE OF ADDRESSES
FC03 4E         GET:    MOV     C,M
FC04 23                 INX     H
FC05 46                 MOV     B,M     ;STARTING ADDRESS
FC06 23                 INX     H
FC07 5E                 MOV     E,M
FC08 23                 INX     H
FC09 56                 MOV     D,M     ;END ADDRESS
FC0A 23                 INX     H
FC0B 78                 MOV     A,B
FC0C B1                 ORA     C       ;CHECK FOR LAST
FC0D C218FC             JNZ     START   ;* WARNING!
FC10 B3                 ORA     E
FC11 C218FC             JNZ     START   ;* WARNING!
FC14 B2                 ORA     D
FC15 CA21F0             JZ      XXXX    ;SET XXXX TO THE ENTRY
                                        ;ADDRESS OF YOUR MONITOR.
                                        ;INITIALIZATION IS COMPLETE.
FC18 0B         START:  DCX     B
FC19 03         NEXT:   INX     B
FC1A AF                 XRA     A       ;CLEAR REG.A TO ZERO
FC1B 2F                 CMA
FC1C 02                 STAX    B       ;STORE "FF"
FC1D 78                 MOV     A,B
FC1E BA                 CMP     D
FC1F DA19FC             JC      NEXT    ;* WARNING!
FC22 79                 MOV     A,C
FC23 BB                 CMP     E
FC24 DA19FC             JC      NEXT    ;* WARNING!
FC27 C303FC             JMP     GET     ;NEXT AREA PLEASE.

FC2A            TABLE:  DS      12      ;* WARNING!
                                        ;SET WITH ADDRESS VALUES

                ;*....THIS BYTE OR INSTRUCTION CAN NOT BE FF HEX.
0000                    END
```

3.3 (continued)

(3) If individual IK blocks need to be initialized to FF Hex, then the routine on the previous page will help.

The lower part of the program labeled "Table" is where the starting & ending addresses of each block is stored. The format is:

```
Table:          Starting Address 1;  Two Bytes (low half first)
                Ending   Address 1;  Two Bytes

                Starting Address 2;  Two Bytes
                Ending   Address 2;  Two Bytes

                   '        '
                   '        '
                   '        '

                Starting Address Last
                Ending   Address Last

                NOP              ;  Four zero Bytes
                NOP              ;  Indicate end of table.
                NOP
                NOP
```

3.4 Wait State Selection

The MB-8A can be set for zero to eight wait states.

Prom Access Time* (n sec)	Dip Switch Settings WE WØ W1 W2	Wait Cycles	DIP Switch Notation
Less than 55Ø	1 Ø Ø Ø	Ø	Ø=switch closed
1Ø5Ø	Ø 1 1 1	1	1=switch open
15Ø	Ø Ø 1 1	2	
2Ø5Ø	Ø 1 Ø 1	3	
25Ø	Ø Ø Ø 1	4	
3Ø5Ø	Ø 1 1 Ø	5	
35Ø	Ø Ø 1 Ø	6	
4Ø5Ø	Ø 1 Ø Ø	7	
Less than 45Ø	Ø Ø Ø Ø	8	

*--Prime 2708's normally do not require wait cycles.

3.5 Jump Address Selection

The MB-8A is equipped with a Power-On Clear or Reset Vector Jump Circuit. To activate this function, switch S2-JE must be switched to the closed (on) position. The Jump Select Connection has already been connected for Power-On Jump, but this small PC Jumper can be cut to allow for a Jumper Wire Selecting the Reset option.

The Jump circuit generates four Bytes onto the bus which are:

F3------Disable Interrupts
C3------Jump Instruction, 1st Byte
ØØ------Low Address, 2nd Byte
YY------High Address, 3rd Byte

The last Byte (YY) can be set by switch S1 which controls the high address (A8 thru A15). A switch position set to open (off) is a logic Ø state and set to closed (on) as a logic 1. S1 can be set for any address from ØØØØH to FFØØH in 256 Byte increments.

Jump Address	Setting of S1 A15 A14 A13 A12 A11 A1Ø A9 A8
ØØØØ Hex	All Open (off)
Ø1ØØ Hex	All Open, Except A8=closed
Ø2ØØ Hex	All Open, Except A9=closed
'	
'	
'	
FFØØ Hex	All Closed (On)

4.0 Trouble Shooting Hints

a. Check for proper settings of DIP switches.

b. Verify that all ICs are in the correct sockets.

c. Visually inspect all ICs to be sure that leads are in the sockets and not bent under.

d. Verify that the output voltage of each regulator is correct.

e. Inspect back side of board for solder bridges, running a small sharp knife blade between traces that appear suspicious. A magnifying glass is a must for this.

f. If you have an addressing problem:

1) Check U35 (7465136) for addresses A14 & A15.

2) Check U28 (74367) for address A13.

3) Check U30 & U34 (74L138) for addresses A10 thru A12.

4) Check U27 & U28 (7465367) for addresses AØ thru A9.

g. If you have a problem with data output (consistent missing bits):

1) Check outputs of buffers U32 & U33 (74L5367) for shorts as well as proper operation.

2) Check signals on U23 (74L$ØØ).

h. If you have a wait cycle problem:

1) Check signals on U23 (74L$197)

2) Check signals on U31 (74L$04)

3) Check signals on U36 (75453)

i. If you have a vector jump problem:

1) Check U26, U2 & U14 if the jump code is wrong.

2) Check U26 & U29 if no code sequence is sent out. Also check switch S1-JE for proper operation.

j. If the "Magic Maping" feature doesn't work, then check out U11 & U29 for proper operation.

5.0 Theory of Operation

5.1 Useage

1) U11 (8-Input Nand, 74LS3Ø) is a detector for "FF" Hex Bytes.

2) U23 (Presettable Binary Counter, 74197) is used to count Phase 2 cycles to give a number of different Wait State cycles depending on the settings of S2.

3) U26 (Hex D Flip-Flops, 74LS174) is used as a shift register to sequence the Jump code on to the bus for Power-On Jump.

4) U27, U28 (Hex Tri-State Buffers, 74LS367) are to buffer (isolate) the lower address lines (AØ thru A9, A13) onto the card.

5) U29 (Quad 2-Input Nand, 74LSØØ) is used as control logic for address enable, PDBIN strobe and PRDY functions on the card.

6) U3Ø, U34 (3 to 8 Decoder, 74LS138) are used for decoding the address for driving the appropriate chip select pin of the bank of 2708's.

7) U31 (Hex Inverter, 74LSØ4) is used to buffer signal lines onto the card (Psync, Phase 2, A1Ø, A11 & A12).

8) U32, U33 (Hex Tri-State Buffers, 74LS367) are used for buffering the eight output data lines. These buffers are turn-on by PDBIN.

5.1 (continued)

9) U35 (Quad Exclusive-Or gates with open collectors, 74LS136) is used to compare the Dip Switch address selected with A14 & A15. This IC is also used as a phantom line driver and a enable gate for SMEMR status to the card.

10) U36 (Dual 2-Input or with open collectors, 75453) is used to drive the PRDY and PS bus lines.

11) U1, U13 (340T-12) positive 12 volt regulators for the PROMS.

12) U12, U24 (320T-5) negative 5 volt regulators for the PROMS.

13) U25 (340T-5) positive 5 volt regulator for logic power.

5.2 Operation

The MB-8 card uses U30 & U34 for chip selection of the 2708's. U30 & U34 decode the address bits A1Ø, A11, A12 & A13 for address steps in 1K increments. U30 & U34 are only disabled during the Vector Jump operation of the card. U35 is used to decode the address selection from S2 against A15 & A14 for a SMEMR state. If the address is valid, then a enable signal (logic 1) is sent to U29, pin 12, for a possible output data enable signal. U29 receives a address enable signal and a signal from U11 which is a "FF" code detector. The "FF" detector will output a logic zero to U29, pin 13 if all ones are detected. U29 will provide an enable signal (U29, pin 3=Ø) for output data to U32 & U33, if the address is valid (U29, pin 13=1), and PDBIN signal is valid (U29, pin 2=1).

The Wait State circuit uses a counter U23 to count out the number of wait cycles. PSYNC drives the load enable of U23, pin 1 through a inverter U31. PSYNC loads U23 with a binary number from S2 which activates the PRDY signal to U36 pin 7. (logic Ø) Phase 2 pulses the clock of U23 which counts up from the binary number until U23, pin 5 goes high to turn-on the PRDY signal.

The Vector Jump circuit is activated by a logic zero on pin 1 (clear) of U26 . The logic zero signal can be PRESET or POC depending on the user selected Jumper wire. Switch S2, position JE, must be closed for the Vector Jump to operate.

U26 is connected up as a shift register to sequence (five cycles) out the Jump instruction onto the Computer's Bus. U26 controls the Input Select, chip enable and some on the Input lines of the 74LS258 IC's (U2, U14). U26 is clocked through its sequence by a pulse form U29, pin 3 (PDBIN).

U2, U14 receive logic patterns on the A-select Inputs (pins 2, 5, 11, 4) for the Hex codes F3, C3, & ØØ from U26. These Hex codes give the computer a Disable Interrupt and 2/3 of a Jump Command. The Final Byte (high address) of the Vector Jump is received from Switch S1 (8 bits) to the B-Select Inputs of U2 & U14. After the four machine instruction Bytes,

5.2 (continued)

a fifth cycle is sequenced which turns off U2 & U14.

During the whole Vector Jump operation U30 & U34 are disable so that all the PROM's outputs are disable, and U2 & U14 are enable so their Tri-State Outputs are turned-on.

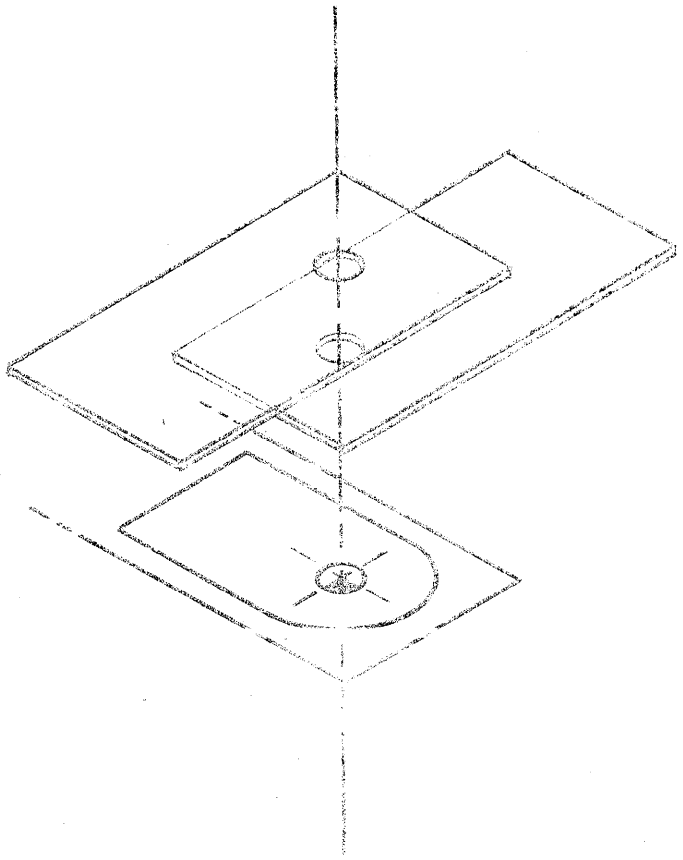
6.0 Warranty

Parts guaranteed to original purchaser for 90 days, unless failure is due to misuse or failure of purchaser to excercise caution in addembly and operation. Registration card must be returned at time of purchase to validate warranty.

Assembled boards may be returned for service. A service charge will be made unless, in our judgement, the problem is due to a defective board or parts.

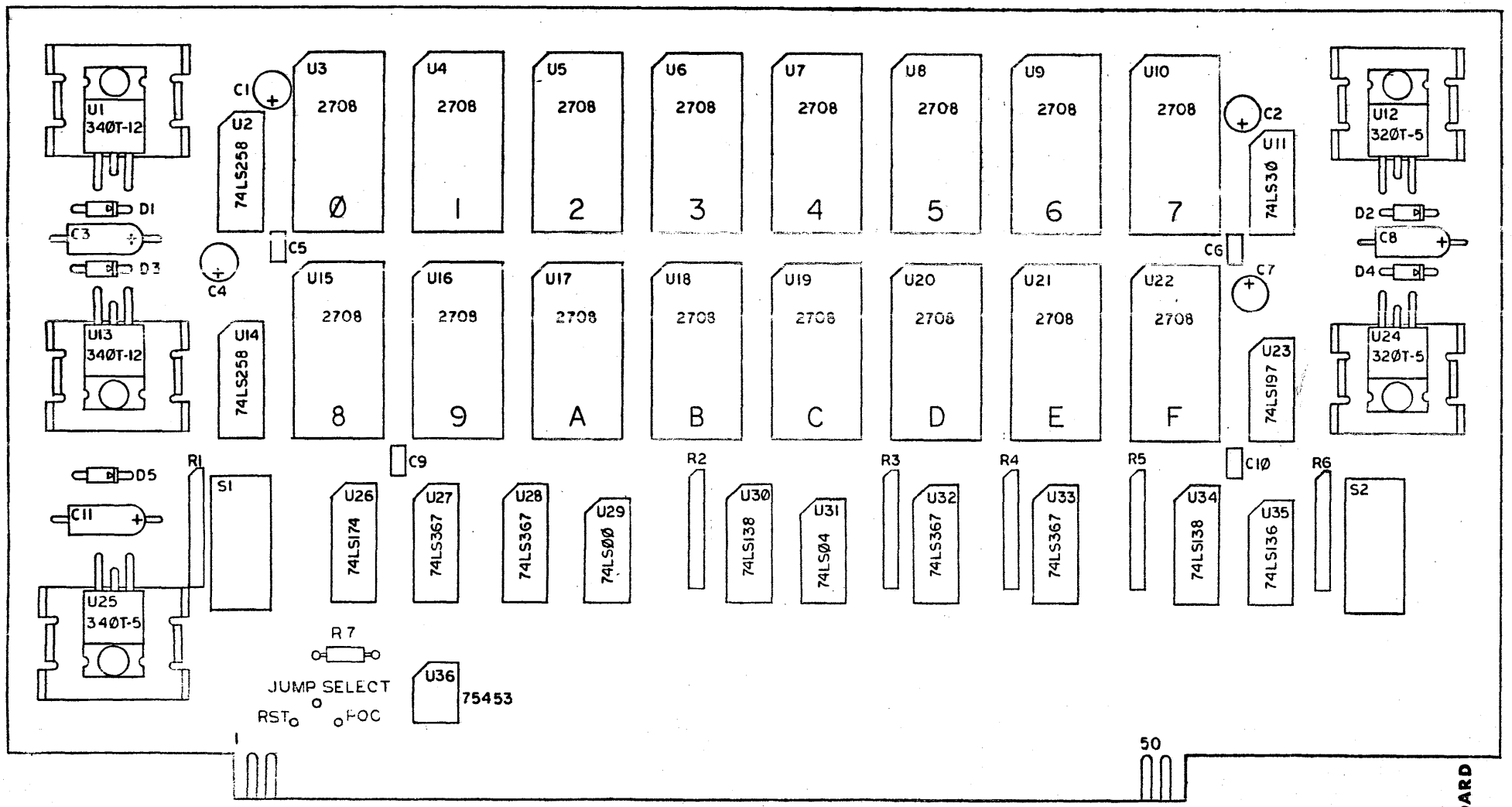
SSM

2116 Walsh Avenue
SANTA CLARA, CALIFORNIA 95050
(408) 246-2707

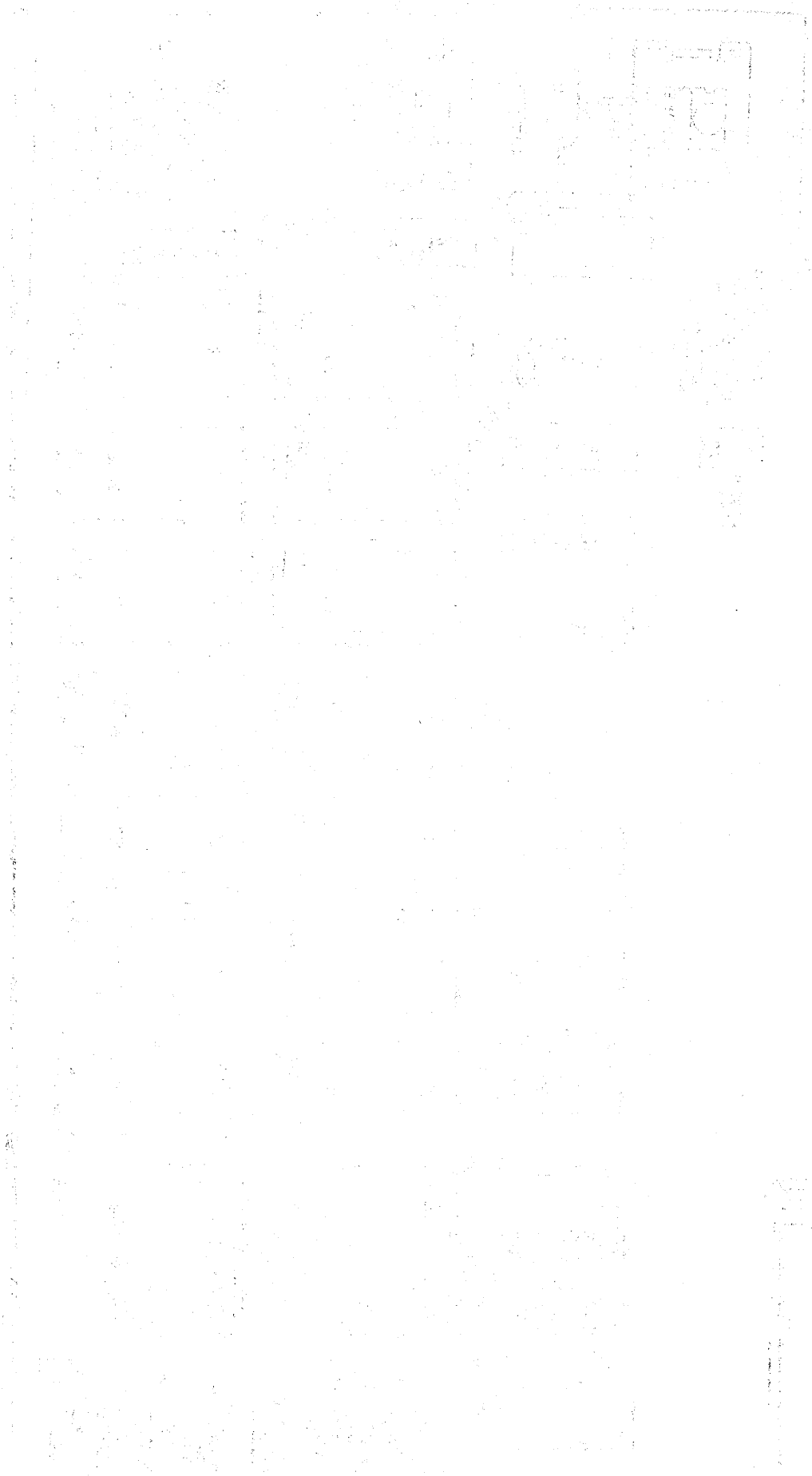


NOTE:

Place two plastic insulators sideways, as shown and underneath the regulator heat sink of the two negative voltage regulators U12 & U24.



ASSEMBLY
16K 2708 EPROM BOARD
© 1977
MB8A



MB8A Parts List

Chip Pack

2 - U2,14	74L$258
1 - U11	74L$30
1 - U23	74L$197/8291
1 - U26	74L$174
1 - U29	74L$00
2 - U30,34	74L$138
1 - U31	74L$04
1 - U36	75453
4 - U27,28,32,33	74367/8097
1 - U35	74L$136

Capacitor Pack

3 - C3,8,11	10uf,20v
4 - C5,6,9,10	0.1uf disc
4 - C1,2,4,7	4.7uf 20v drop tant

Resistor Pack

5 - D1-5	1N4001 to 1N4006
6 - R1-6	2.7K x 7 SIP pack
1 - R7	10K ¼W 5%

Hardware Pack

2 - U1,13	7812/340T-12
2 - U12,24	7905/320T-5
1 - U25	7805/340T-5
5	Heatsinks
4	Insulators
5	Sets #6 hardware

Socket Pack

1	24 pin socket
5	14 pin sockets
1	8 pin socket
2	8 position DIP switch

Misc.

15	24 pin sockets
9	16 pin sockets
1	PC board
1	Warranty card
1	Instruction manual

12-1-78

MB8A ADDENDUM

Magic Mapping vs. Wait States

The magic mapping function will disable the outputting of data and wait states from the MB8A, if there is no ROM in the socket or the ROM's code is FF (Hex). Another way a ROM can generate an FF code is to be slow in responding to a chip select on pin 20. A slow ROM can disable its own wait state signal back to the computer.

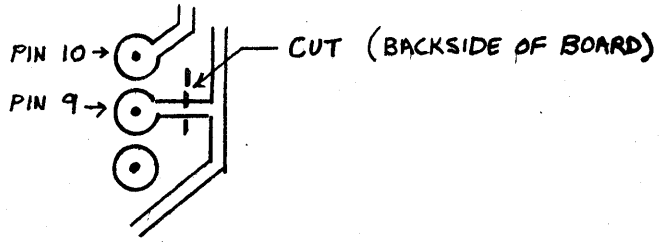
In most computer systems today, the ROM's speed is better than 550 nanoseconds for chip select and the main system clock is 2mhz, so no read problems occur. If the ROM is very slow or the computer is running at 4mhz, then do the following:

1. Wait States wanted, but no Magic Mapping!

   a. Remove U11 from the board to defeat Magic Mapping.
   b. The MB8A is now a full 16K ROM with/or without the ROMs in their sockets.

2. Wait States wanted and Magic Mapping!

   a. Cut the very short trace going to U29, pin 9.



   B. Jumper U29, pin 9 to U29, pin 10.

SCHEMATIC
16K 2708 EPROM BOARD

MB8A