PROM/RAM BOARD USERS MANUAL AND ASSEMBLY INSTRUCTIONS

S100
2K PROM
ADD RAM
LITTLE OLD

256 X 8 OUT PROM

VECTOR GRAPHIC INC.
31364 VIA COLINAS
WESTLAKE VILLAGE, CA 91361

PROM/RAM BOARD

TABLE OF CONTENTS

PROM/RAM BOARD USERS MANUAL

AND

ASSEMBLY INSTRUCTIONS

DESCRIPTION

CONGRATULATIONS ON YOUR PURCHASE OF A VECTOR GRAPHIC INC. PROM/RAM BOARD.

THIS UNIQUE PROM/RAM BOARD ANSWERS THE NEED FOR A MEANS OF STORING PROGRAMS SUCH AS BOOTSTRAP LOADERS, MONITOR PROGRAMS, AND VIDEO DRIVERS, ON NON-VOLATILE PROMS. SINCE SUCH PROGRAMS GENERALLY REQUIRE RAM FOR STACK OPERATIONS, 1K BYTES OF RAM ARE ALSO PROVIDED ON THE BOARD. WHILE RAM IS USUALLY AVAILABLE ELSEWHERE IN A SYSTEM, IT IS QUITE INCONVENIENT TO REPROGRAM THE PROMS TO RELOCATE THE STACK EACH TIME MORE MEMORY IS ADDED TO THE SYSTEM.

THE PROM/RAM BOARD WHEN USED IN CONJUNCTION WITH VECTOR GRAPHIC INC. 512 BYTE MONITOR PROGRAM, PROVIDES THE USER WITH A COMPLETE OPERATIONAL SYSTEM WITHOUT ADDITIONAL MEMORY. CIRCUITRY ON THE BOARD REPLACES THE MEMORY WRITE LOGIC FOUND ON THE FRONT PANEL BOARD OF IMSAI AND "ALTAIR"[T.M.] COMPUTERS. A JUMP ON RESET FEATURE ALLOWS A PROGRAM IN PROM TO BE EXECUTED STARTING AT ANY LOCATION IN MEMORY WITHOUT INTERFERING WITH PROGRAMS IN ANY OTHER PORTION OF MEMORY.

ASSEMBLY INSTRUCTIONS

PURPOSE

THE PURPOSE OF THESE INSTRUCTIONS IS TO HELP YOU PRODUCE THE BEST RESULTS IN THE SHORTEST TIME WITH NO DAMAGE TO THE VARIOUS COMPONENTS.

IF THERE IS ANYTHING THAT YOU DO NOT UNDERSTAND, PLEASE DO NOT HESITATE TO CALL OR WRITE US!

AFTER COMPLETING THE ASSEMBLY, PLEASE FILL OUT AND RETURN THE WARRANTY CARD SO THAT WE CAN ADD YOU TO OUR MAILING LIST FOR FUTURE PRODUCTS.

IMPORTANT PRECAUTIONS

POWER MUST BE OFF WHEN:

- INSERTING OR REMOVING BOARDS OR IC CHIPS
- CONNECTING OR DISCONNECTING WIRES
- SOLDERING

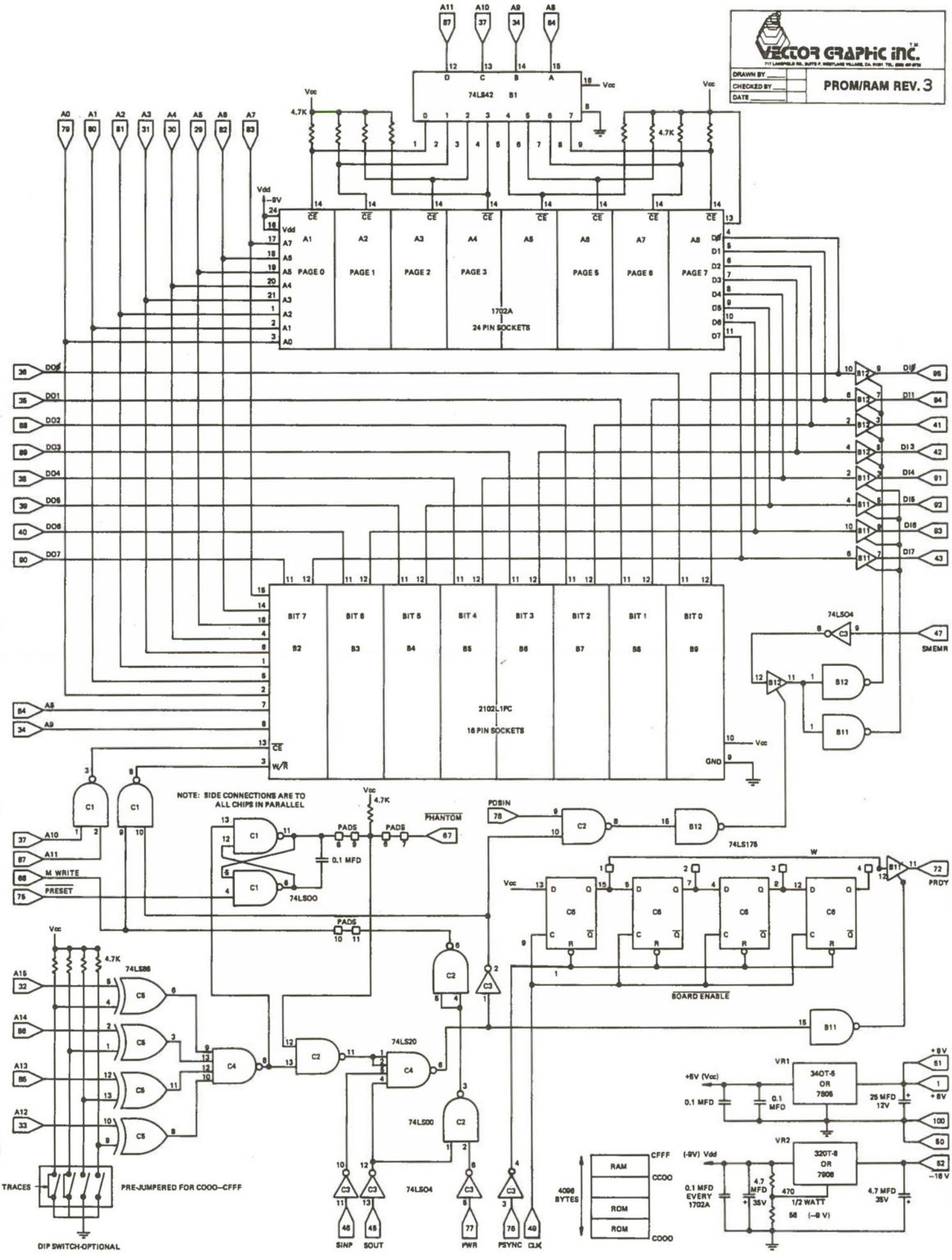
ONLY SOLDER WITH:

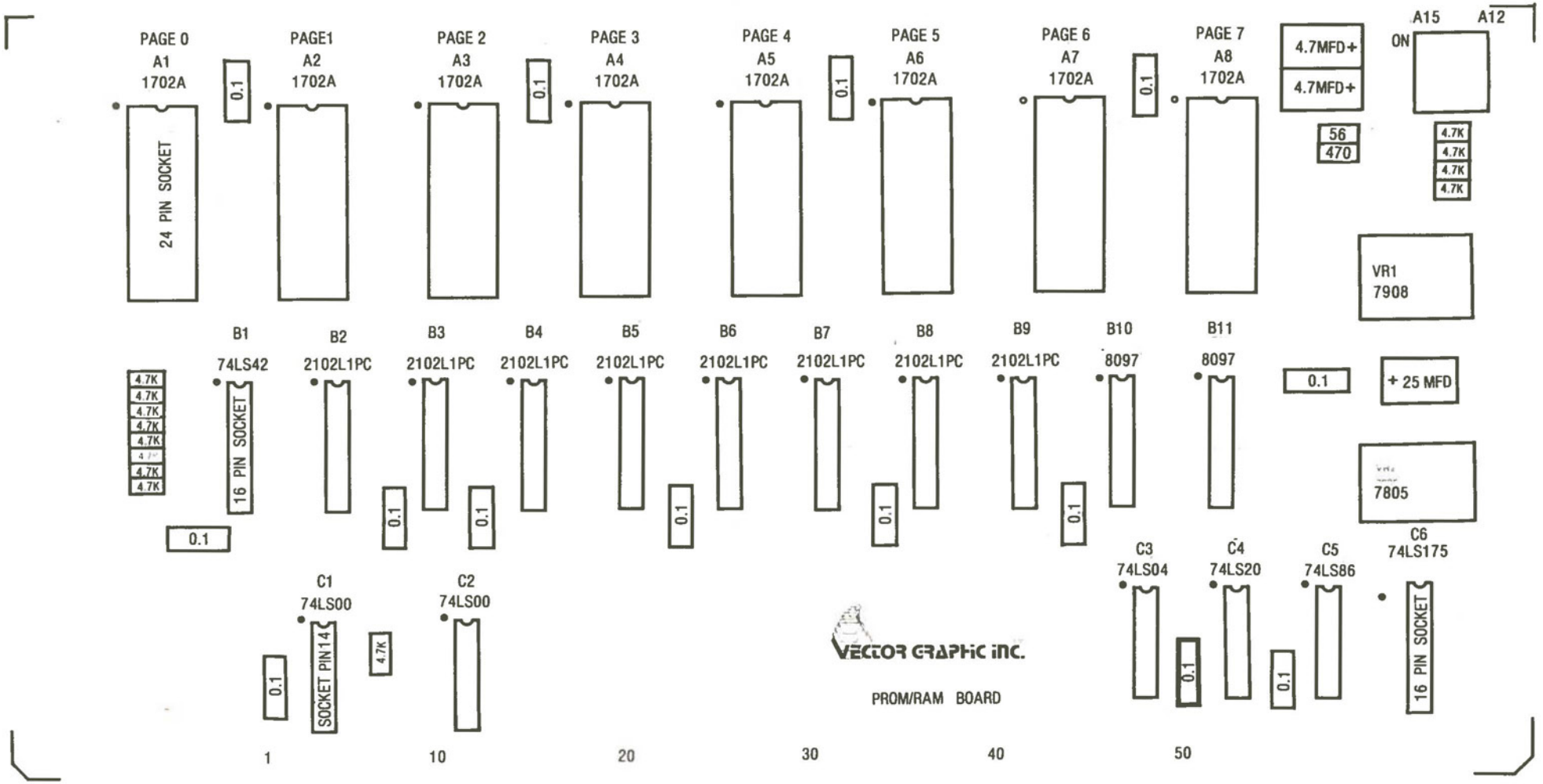
- 30 WATT MAXIMUM SOLDERING IRON
- 60/40 ROSIN CORE SOLDER

ALWAYS PROTECT MOS CHIPS FROM STATIC ELECTRICITY.

PROM/RAM BOARD KIT CONTENTS

QUANTITY	DESCRIPTION
	PRINTED CIRCUIT BOARD
8	24 PIN IC SOCKETS
12	16 PIN IC SOCKETS
5	14 PIN IC SOCKETS
14	0.1 MFD DISC CAPACITORS
13	4.7K RESISTORS 1/4 WATT (BANDS OF YELLOW, VIOLET, RED)
1	470 OHM RESISTOR 1/4 WATT (BANDS OF YELLOW, VIOLET, BROWN)
1	56 OHM RESISTOR 1/4 WATT (BANDS OF GREEN, BLUE, BLACK)
2	4.7 MFD 50 VOLT ELECTROLYTIC CAPACITORS
1	25 MFD 12 VOLT ELECTROLYTIC CAPACITOR
8	2102LIPC
2	74367/8097
2	74LS00
1	74LS04
1	74LS20
1	74LS42
1	74LS86
1	74LS175
1	7805 REGULATOR
1	7908 REGULATOR
2	HEAT SINKS
1	MICA INSULATOR FOR HEAT SINK
1	6-32 x 3/8 METAL SCREW, NUT AND LOCKWASHER
1	6-32 x 3/8 NYLON SCREW, NUT AND LOCKWASHER
1	USERS MANUAL AND ASSEMBLY INSTRUCTION
1	GENERAL TROUBLE SHOOTING GUIDE


VECTOR GRAPHIC INC.
DRAWN BY
CHECKED BY
DATE
PROM/RAM REV. 3
74LS42 B1
1702A
24 PIN SOCKETS
PAGE 0
PAGE 1
PAGE 2
PAGE 3
PAGE 5
PAGE 6
PAGE 7
2102L1PC
16 PIN SOCKETS
BIT 7
BIT 6
BIT 5
BIT 4
BIT 3
BIT 2
BIT 1
BIT 0
4.7K
0.1 MFD
74LS00
74LS04
74LS86
74LS20
74LS175
NOTE: SIDE CONNECTIONS ARE TO ALL CHIPS IN PARALLEL
PHANTOM
PRESET
M WRITE
BOARD ENABLE
TRACES
PRE-JUMPERED FOR C000–CFFF
DIP SWITCH-OPTIONAL
4096 BYTES
RAM
ROM
ROM
CFFF
CC00
C000
VR1
VR2
340T-5 OR 7805
320T-5 OR 7905
+5V (Vcc)
(-9V) Vdd
25 MFD 12V
4.7 MFD 35V
0.1 MFD EVERY 1702A
470 1/2 WATT
56 (–9 V)
SINP
SOUT
PWR
PSYNC
CLK
SMEMR
PRDY
PDBIN
W/R
CE
GND
Vcc
Vdd


PAGE 0
A1
1702A
24 PIN SOCKET
0.1
PAGE1
A2
1702A
PAGE 2
A3
1702A
0.1
PAGE 3
A4
1702A
PAGE 4
A5
1702A
0.1
PAGE 5
A6
1702A
PAGE 6
A7
1702A
0.1
PAGE 7
A8
1702A
4.7MFD+
4.7MFD+
56
470
A15
A12
ON
4.7K
4.7K
4.7K
4.7K
VR1
7908
B1
74LS42
16 PIN SOCKET
B2
2102L1PC
B3
2102L1PC
B4
2102L1PC
B5
2102L1PC
B6
2102L1PC
B7
2102L1PC
B8
2102L1PC
B9
2102L1PC
B10
8097
B11
8097
4.7K
4.7K
4.7K
4.7K
4.7K
4.7K
4.7K
0.1
0.1
0.1
0.1
0.1
0.1
0.1
0.1
0.1
+ 25 MFD
7805
C1
74LS00
SOCKET PIN14
C2
74LS00
4.7K
0.1
C3
74LS04
C4
74LS20
C5
74LS86
C6
74LS175
16 PIN SOCKET
0.1
0.1
VECTOR GRAPHIC INC.
PROM/RAM BOARD
1
10
20
30
40
50

TOOLS AND MATERIALS REQUIRED FOR ASSEMBLY

THE FOLLOWING MINIMUM SET OF TOOLS AND MATERIALS IS REQUIRED FOR THE ASSEMBLY OF VECTOR GRAPHIC INC. KITS:

DESCRIPTION	**COMMENT**
VOLT - OHMMETER	INEXPENSIVE
SCREWDRIVER - STRAIGHT SLOT	FOR #5 and #8 SCREWS
SCREWDRIVER - PHILLIPS HEAD*	FOR #8 SCREWS
CUTTERS - DIAGONAL	4", FLUSH CUTTING
PLIERS - NEEDLE NOSED	6"
PLIERS - REGULAR	MEDIUM
WIRE STRIPPER	FOR 8 AWG TO 20 AWG
SOLDERING IRON	30 WATTS MAXIMUM WITH CHISEL TIP
SOLDER	.030 GA. 60/40 TIN-LEAD ROSIN CORE
SPONGE	FOR CLEANING SOLDERING IRON
PEN KNIFE	OR 'X-ACTO KNIFE
CLEANING SOLVENT	TRICHLOROETHANE OR ISOPROPYL ALCOHOL. ***DO NOT USE ACETONE***
CARDBOARD	TO PROTECT TABLE TOP DURING SOLDERING
HEAT SINK GREASE	OR HIGH TEMPERATURE PLUMBERS GREASE
RULER*	TO MEASURE WIRE LENGTHS

*NOTE: REQUIRED FOR MAINFRAME CABINET ASSEMBLY ONLY

SOLDERING TECHNIQUE

THE SOLDER

USE A #20 GAUGE (.030") ROSIN CORE SOLDER WITH A RATIO OF AT LEAST 60% TIN AND 40% LEAD. "KESTER" AND "ERSIN" ARE TWO DEPENDABLE BRANDS OF SOLDER. ACID CORE SOLDERS OR ACID FLUX MUST NOT BE USED AS THEY WILL CORRODE THE PRINTED CIRCUIT BOARD.

THE SOLDERING IRON

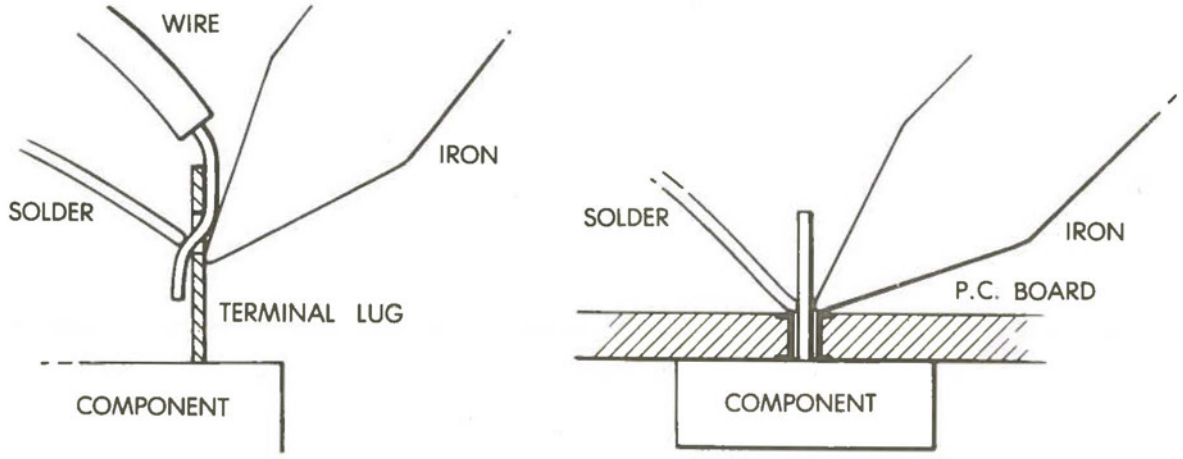
USE A SMALL, 30 WATT MAXIMUM IRON WITH A SMALL, CHISEL SHAPED TIP. TOO MUCH HEAT WILL DAMAGE BOTH COMPONENTS AND BOARDS. SOLDERING GUNS ARE TOO HOT AND SHOULD NOT BE USED.

HEAT THE IRON, WIPE ITS TIP QUICKLY ON THE DAMP SPONGE, AND APPLY A TINY AMOUNT OF SOLDER TO THE TIP - JUST ENOUGH TO MAKE IT SILVER IN COLOR BUT NOT SO MUCH THAT IT WILL DRIP OFF. THIS CLEANING PROCEDURE SHOULD BE REPEATED WHENEVER THE TIP OF THE SOLDERING IRON BEGINS TO TAKE ON A BROWNISH COLOR.

THE PROCEDURE

THE ENTIRE SOLDERING OPERATION SHOULD TAKE LITTLE MORE THAN TWO SECONDS PER JOINT. THE SEQUENCE IS AS FOLLOWS:

TOUCH THE TIP OF THE SOLDERING IRON TO THE JOINT, AS SHOWN BELOW, SO THAT BOTH CONDUCTORS TO BE JOINED ARE SIMULTANEOUSLY HEATED SUFFICIENTLY TO MELT THE SOLDER.



TOUCH THE SOLDER TO THE JOINT, AS SHOWN ABOVE, JUST LONG ENOUGH TO MELT ENOUGH SOLDER TO FORM A FILLET ON THE JOINT. TOO MUCH SOLDER MAY SHORT CIRCUIT THE BOTTOM OF THE BOARD OR FLOW THROUGH THE HOLES AND WICK INTO THE SOCKETS. THE MELTED SOLDER WILL APPEAR WET AND SHINY. IT WILL QUICKLY FLOW COMPLETELY AROUND THE WIRE AND OVER THE SURFACE TO WHICH THE WIRE IS ATTACHED.

REMOVE THE SOLDERING IRON AS SOON AS BOTH SURFACES HAVE BEEN COMPLETELY WETTED. REMEMBER, THE TOTAL TIME FROM APPLICATION TO REMOVAL OF THE SOLDERING IRON SHOULD BE ONLY TWO OR THREE SECONDS. REMOVAL OF THE SOLDERING IRON TOO SOON MAY RESULT IN A COLD SOLDER JOINT AND LEAVING THE SOLDERING IRON IN CONTACT TOO LONG MAY CAUSE HEAT DAMAGE TO EITHER THE COMPONENTS OR THE BOARD.

REMOVAL OF MULTI-PIN SOLDERED-IN PARTS

CAUTION

IF FOR ANY REASON, IT BECOMES NECESSARY TO REMOVE A SOLDERED-IN PART HAVING MORE THAN JUST TWO LEADS, DO NOT TRY TO REMOVE THE PART INTACT. IT CAN BE DONE BUT ONLY WITH RISK OF DAMAGING THE PRINTED CIRCUIT BOARD IN THE PROCESS.
HOLD THE PRINTED CIRCUIT BOARD IN A PADDED VISE TO AVOID DAMAGE.

REMOVAL OF SOLDERED-IN IC SOCKETS

CAREFULLY PRY UP THE PLASTIC BODY OF THE SOCKET USING A KNIFE OR SCREWDRIVER TO LEAVE THE PINS EXPOSED. GENTLY REMOVE THE PINS FROM THE TOP OF THE BOARD WITH NEEDLE NOSED PLIERS WHILE TOUCHING THE JOINT ON THE OTHER SIDE OF THE BOARD WITH THE TIP OF THE IRON. DO NOT USE FORCE. THE PIN WILL COME OUT QUITE EASILY ONCE THE SOLDER MELTS.
CLEAR THE HOLES OF ANY EXCESS SOLDER USING A SOLDER SUCKER OR WICK.

REMOVAL OF SOLDERED-IN INTEGRATED CIRCUIT CHIPS

CUT EACH PIN WITH A PAIR OF DIAGONAL CUTTERS AT A POINT BETWEEN THE CHIP AND THE PRINTED CIRCUIT BOARD WHICH IS AS CLOSE TO THE CHIP AS POSSIBLE SO THAT THERE IS ENOUGH OF THE PIN SHOWING ABOVE THE BOARD TO BE GRASPED BY NEEDLE NOSED PLIERS WHILE REMOVING AS DESCRIBED ABOVE.

PREPARATION FOR ASSEMBLY

WORKING AREA AND TOOLS

A WELL LIGHTED, CLEAN TABLE OR WORK BENCH AND THE PROPER TOOLS AND MATERIALS ARE MOST IMPORTANT FOR PRODUCING TROUBLE FREE ASSEMBLIES. THE WORK SURFACE SHOULD BE CLEAN AND FREE OF ALL ITEMS EXCEPT FOR THE TOOLS AND KIT COMPONENTS BEING USED. A CLEAN PIECE OF CARDBOARD OR HAND TOWEL IS SUGGESTED TO PROTECT THE TABLE TOP WHEN SOLDERING.

CHECK KIT CONTENTS

VERIFY THE CONTENTS OF YOUR KIT AGAINST THE KIT CONTENTS LIST IN THE FRONT OF THIS MANUAL. CHECK EACH PART VISUALLY FOR DAMAGE IN SHIPPING. IF THERE ARE ANY MISSING OR DAMAGED ITEMS, PLEASE NOTIFY THE DEALER FROM WHOM YOU BOUGHT YOUR KIT IMMEDIATELY. THERE MAY BE SLIGHT VARIATIONS FROM THE PARTS SPECIRED, BUT THE COMPONENTS SHOULD BE FUNCTIONALLY EQUIVALENT.

PARTS LAYOUT AND ASSEMBLY SEQUENCE

THE FRONT OF THE BOARD IS THE SIDE ON WHICH THE PARTS LAYOUT HAS BEEN SILK SCREENED. ALL PARTS WILL BE ON THE FRONT OF THE PRINTED CIRCUIT BOARD. THEIR LEADS OR PINS WILL PASS THROUGH THE BOARD AND BE SOLDERED ON THE REAR.

PLACE THE BOARD WITH ITS FRONT SIDE UP AND THE GOLD EDGE CONTACTS NEAREST YOU. IN THAT POSITION, WE WILL REFER TO THE UPPER PORTION OF THE BOARD AS BEING FURTHEST AWAY FROM YOU.

SHOULD YOU USE SOCKETS?

WE RECOMMEND THE USE OF SOCKETS FOR TWO REASONS. ONE IS THAT SOLDERED-IN CHIPS CANNOT BE RETURNED FOR REPLACEMENT. ANOTHER IS THAT, SHOULD YOU HAVE TO REPLACE A CHIP, IT IS POSSIBLE TO DO CONSIDERABLE DAMAGE TO THE P. C. BOARD, UNLESS YOU ARE EXPERIENCED AT IC REMOVAL AND HAVE THE PROPER TOOLS.

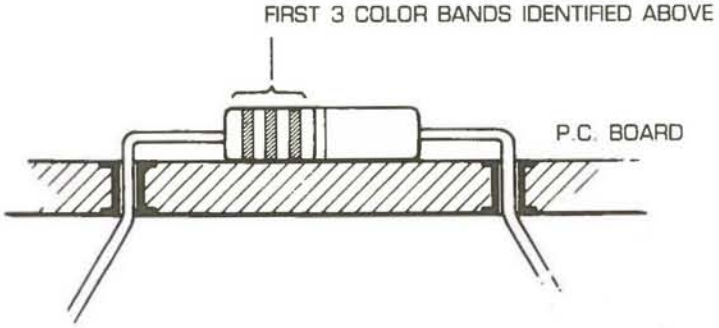
PROM/RAM BOARD ASSEMBLY SEQUENCE

CHECKING THE PRINTED CIRCUIT BOARD:

ALTHOUGH WE HAVE INSPECTED THE BOARD PRIOR TO SHIPMENT, A FURTHER ELECTRICAL CHECK FOR ETCH BRIDGES BETWEEN TRACES MAY BE PERFORMED WITH AN OHMMETER, USING THE LOW RESISTANCE RANGE. MEASURE THE RESISTANCE BETWEEN OPPOSITE PADS ON ONE OF THE 2102L1PC CHIP LOCATIONS, FIRST ONE THEN THE OTHER, LIKE CLIMBING A LADDER.

INSERTION OF RESISTORS

ORIENTATION IS OF NO CONCERN WITH RESISTORS, BUT BE SURE THAT THE STRIPED COLOR CODE WHICH IDENTIRES THE RESISTANCE VALUE IS AS SHOWN BELOW FOR THE PARTICULAR LOCATION.



AREA	LAYOUT SYMBOL	QUANTITY	DESCRIPTION	MARKINGS
VARIOUS	4.7K	13	4.7K OHM 1/4 WATT	YELLOW, VIOLET, RED
UPPER RIGHT	470	1	470 OHM 1/4 WATT	YELLOW, VIOLET, BROWN
UPPER RIGHT	56	1	56 OHM 1/4 WATT	GREEN, BLUE, BLACK

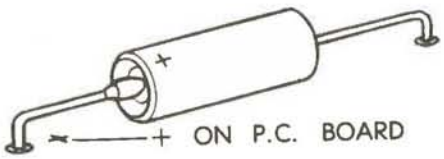
INSERT THE LEADS INTO THE PROPER HOLES, HOLD THE RESISTOR BODY FIRMLY AGAINST THE BOARD, AND THEN SLIGHTLY SPREAD THE LEADS ON THE OPPOSITE SIDE OF THE BOARD TO HOLD IT IN PLACE WHILE SOLDERING.

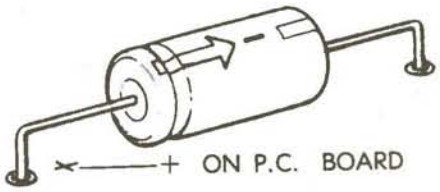
INSPECT FOR PROPER LOCATION AND FOR PROPER SOLDER JOINTS AND THEN CLIP OFF EXCESS LENGTH WITH DIAGONAL CUTTERS.

WHEN THIS PROM/RAM BOARD IS FOR USE WITH THE VECTOR 1 OR OTHER COMPUTERS THAT DO NOT HAVE FRONT PANEL LOGIC, A JUMPER MUST BE INSERTED BETWEEN SOLDER PADS 10 AND 11 ON THE LOWER LEFT HAND PORTION OF THE BOARD. BEND A LEAD CLIPPING FROM ONE OF THE PREVIOUSLY INSTALLED RESISTORS AND INSERT ITS ENDS THROUGH HOLES 10 AND 11 RESPECTIVELY. SOLDER IN PLACE AS YOU WOULD A RESISTOR.

INSERTION OF AXIAL CAPACITORS

AXIAL ELECTROLYTIC CAPACITORS HAVE SPECIAL POLARITY REQUIREMENTS, THE REVERSAL OF WHICH WILL CAUSE DAMAGE TO THE CAPACITOR. MOST SMALL, AXIAL ELECTROLYTICS WILL BE MARKED WITH A "+" AND/OR HAVE A GROOVE AT THE PLUS END. SOME HAVE AN ARROW POINTING TO THE OPPOSITE END WHICH IS "-". THE LEAD FROM THE "+" END IS TO BE INSERTED IN THE HOLE MARKED "+" ON THE PRINTED CIRCUIT BOARD.





INSERT THE AXIAL ELECTROLYTIC CAPACITORS IN THE LOCATION INDICATED BELOW AND ON THE PARTS LAYOUT AND SOLDER IN PLACE IN THE SAME MANNER AS DESCRIBED ABOVE FOR RESISTORS.

AREA	LAYOUT SYMBOL	QUANTITY	DESCRIPTION	MARKINGS
UPPER RIGHT	4.7 MFD	2	4.7 MFD 50 Volt	4.7 MFD
MIDDLE RIGHT	25 MFD	1	25 MFD 12 Volt	25 MFD

IC SOCKET INSERTION

1. CHECK THE PINS OF IC SOCKET TO INSURE THAT NONE ARE MISSING AND THAT EACH IS IN LINE. IF THERE ARE ANY CONTACTS MISSING, THE SOCKET IS DEFECTIVE AND MUST BE REPLACED. IF ANY CONTACTS ARE OUT OF LINE, GENTLY STRAIGHTEN THEM WITH NEEDLE NOSED PLIERS.
2. THE SOCKETS ARE TO BE LOCATED AS FOLLOWS:

AREA	LAYOUT SYMBOL	QUANTITY	DESCRIPTION
UPPER ROW	A-1 - A-8	8	24 PIN SOCKET
MIDDLE ROW	B-1 - B-11	11	16 PIN SOCKET
LOWER ROW	C-1 - C-5	5	14 PIN SOCKET
LOWER ROW	C-6	1	16 PIN SOCKET

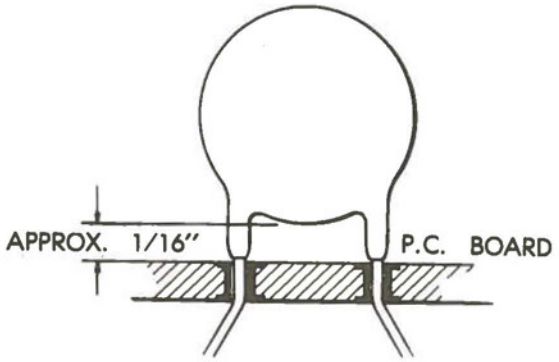
3. CAREFULLY INSERT EACH IC SOCKET IN ITS PROPER LOCATION MAKING SURE THAT ALL ITS PINS ENTER THEIR ASSIGNED HOLES SIMULTANEOUSLY TO AVOID BENDING. CHECK THE BACK OF THE BOARD TO INSURE THAT ALL THE PINS HAVE STARTED THROUGH. PRESS IN AND HOLD THE SOCKET FIRMLY AGAINST THE BOARD WHILE SOLDERING.
4. SOLDER THE DIAGONALLY OPPOSITE PINS OF THE SOCKET FIRST AND THEN HOLD THE BOARD UP TO THE LIGHT TO INSURE THAT EACH SOCKET IS FIRMLY SEATED. THEN SOLDER THE REMAINING PINS.

DO NOT INSERT IC CHIPS UNTIL AFTER ALL OTHER PARTS HAVE BEEN SOLDERED IN AND THE BOARD HAS BEEN CLEANED.

INSERTION OF DISC CAPACITORS

DISC CAPACITORS DO NOT REQUIRE SPECIAL ORIENTATION. HOWEVER, THEY OFTEN HAVE THEIR COATING EXTENDING DOWN FROM THEIR BODY ALONG THEIR LEADS. IF TOO FAR ALONG THE LEAD, IT MAY BE CRACKED OFF BY SQUEEZING IT WITH PLIERS. IN ANY EVENT, BE SURE THAT THIS INSULATIVE COATING DOES NOT EXTEND INTO THE PRINTED CIRCUIT BOARD HOLE.

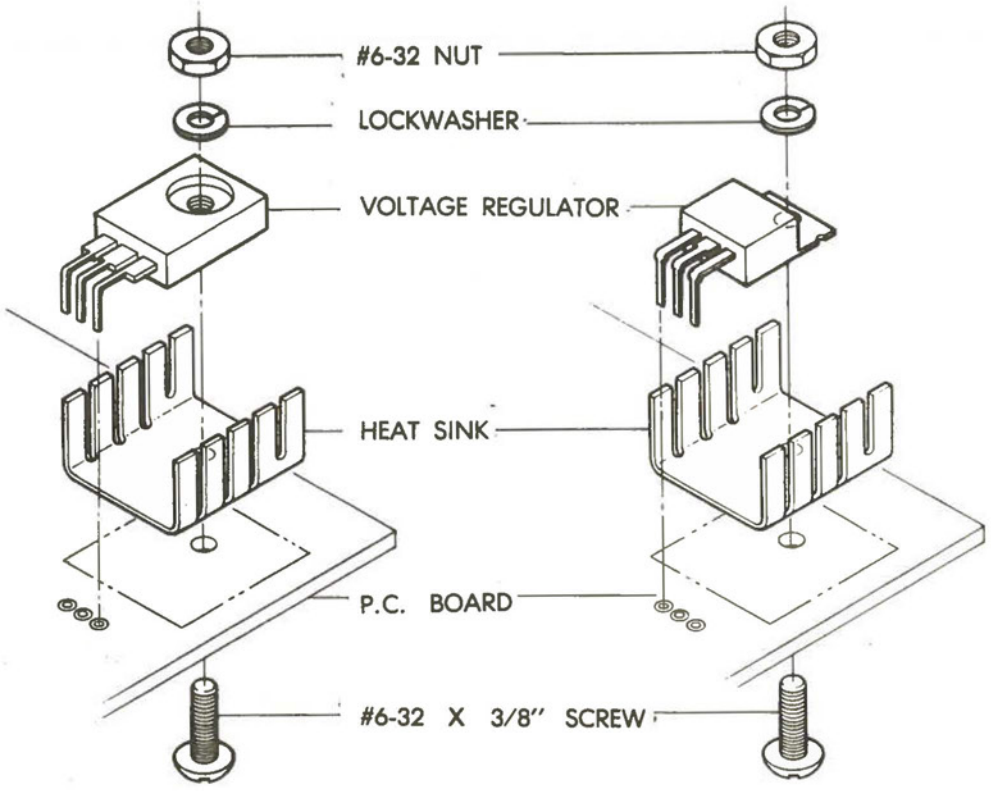
INSERT THE LEADS OF THE 14 DISC CAPACITORS THROUGH THE PROPER HOLES AS INDICATED ON THE PARTS LAYOUT. BEND THE LEADS SLIGHTLY OUTWARD TO HOLD THE CAPACITOR IN POSITION WHILE SOLDERING. THE DISC CAPACITORS SHOULD BE SPACED UNIFORMLY ABOVE THE PRINTED CIRCUIT BOARD ABOUT 1/16" SO AS TO GIVE A NEAT APPEARANCE OF THE FINISHED BOARD. SOLDER IN PLACE WHILE HOLDING IN THIS POSITION.



INSPECT FOR PROPER LOCATION AND FOR PROPER SOLDER JOINTS, AND THEN CLIP OFF EXCESS LEAD LENGTH WITH DIAGONAL CUTTER.

INSTALLATION OF VOLTAGE REGULATORS AND HEAT SINKS

THERE ARE TWO VOLTAGE REGULATORS ON THE PROM/RAM BOARD, A 7805 AND A 7908, EACH TO BE USED WITH A HEAT SINK. ***POSITION THE HEATSINK TO ALLOW CLEARANCE AT THE EDGE OF THE BOARD.*** THE 7908 MUST BE INSULATED.



MEASURE THE REGULATOR LEADS AGAINST THE P.C. BOARD, AND USING NEEDLE NOSED PLIERS, CAREFULLY BEND THE LEADS DOWN TO FORM A RIGHT ANGLE AS SHOWN ABOVE.

ASSEMBLY OF VOLTAGE REGULATORS

FIRST ASSEMBLE THE 7805 REGULATOR ON THE FRONT OF THE BOARD IN THE LOCATION NOTED ON THE PARTS LAYOUT.

1. INSERT THE 6-32 x 3/8" METAL SCREW FROM THE BACK OF THE PRINTED CIRCUIT BOARD.
2. APPLY A THIN COAT OF HEAT SINK GREASE OR PLUMBERS GREASE TO BOTH SIDES OF THE HEAT SINK. THIS WILL GREATLY IMPROVE THE CONDUCTION OF HEAT BETWEEN COMPONENTS.
3. PLACE THE HEAT SINK ON THE TOP OF THE BOARD OVER THE PROTRUDING SCREW.
4. PLACE THE VOLTAGE REGULATOR OVER THE SCREW WHILE CAREFULLY INSERTING ITS LEADS INTO THEIR PROPER HOLES.
5. PLACE THE LOCKWASHER OVER THE END OF THE SCREW AND FINALLY THE METAL NUT.
6. CAREFULLY TIGHTEN THE SCREW FROM THE BACK WITH A SCREWDRIVER WHILE HOLDING BOTH THE HEAT SINK TO INSURE THE PROPER ALIGNMENT AND THE REGULATOR TO PREVENT ANY STRAIN ON THE LEADS CAUSED BY TURNING PRESSURE.
7. SOLDER THE LEADS ON THE BACK OF THE BOARD. INSPECT FOR PROPER SOLDER JOINTS AND THEN CLIP OFF EXCESS LEAD LENGTH WITH DIAGONAL CUTTERS.

ASSEMBLE THE 7908 AND HEAT SINK IN THE LOCATION NOTED ON THE FRONT OF THE BOARD IN THE SAME MANNER, EXCEPT THAT A NYLON SCREW IS TO BE USED AND THE THIN INSULATING WAFER MUST BE PLACED BETWEEN THE REGULATOR AND ITS HEAT SINK. APPLY THE HEAT SINK GREASE OR PLUMBERS GREASE LIGHTLY TO BOTH SIDES OF THE MICA INSULATOR.

TESTING THE VOLTAGE REGULATORS

CAUTION

SHORTED REGULATORS HAVE BEEN KNOWN TO EXPLODE. STAY CLEAR OF REGULATOR SIDE OF BOARD WHILE TESTING. APPLY POWER TO THE BOARD BY PLUGGING IT INTO YOUR COMPUTER AND THEN TURNING THE POWER ON.

MEASURE THE REGULATED OUTPUT OF EACH REGULATOR. ON THE 7805 REGULATOR, THE MIDDLE PIN IS GROUND AND THE LOWER PIN IS THE 5 VOLT REGULATED OUTPUT. ON THE 7908 REGULATOR, THE TOP PIN IS GROUND AND THE BOTTOM PIN IS THE 9 VOLT REGULATED OUTPUT. IF EITHER VOLTAGE VARIES BY MORE THAN ±5%, THE REGULATOR MAY NEED TO BE REPLACED.

INSPECTION AND CLEANING

CAREFULLY INSPECT THE ACTUAL LAYOUT OF THE PARTS ON THE BOARD WITH THE PARTS LAYOUT DRAWING. DO NOT INSERT IC CHIPS YET.

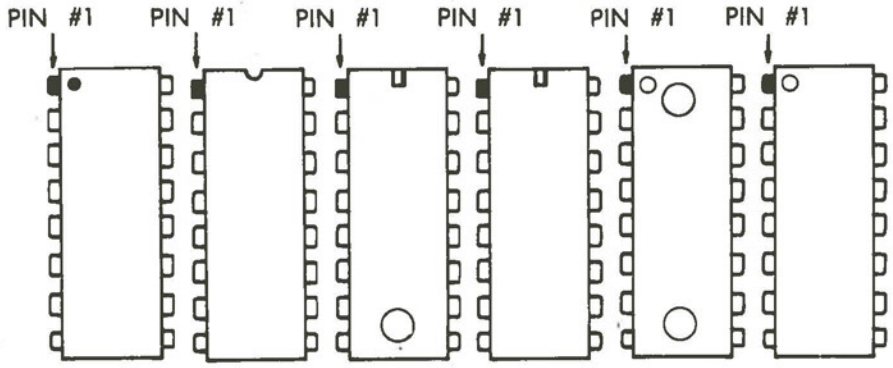
AFTER HAVING SOLDERED ALL COMPONENTS ON THE BOARD, REINSPECT EACH JOINT AREA TO INSURE THAT ALL JOINTS HAVE BEEN SOLDERED AND ARE SHINY AND THAT NO TINY ETCH OR SOLDER BRIDGES HAVE BEEN LEFT BETWEEN TRACES. LETTING A BRIGHT LIGHT SHINE THROUGH THE BOARD MAY HELP YOU LOCATE TINY SOLDER BRIDGES BETWEEN HOLES OR TRACES. IF ANY JOINTS HAVE A "MILKY" COLOR OR "SUGARY" TEXTURE, THEY MUST BE REHEATED WITH THE IRON TO ACHIEVE THE SHINY LOOK.

THE BOARD CAN BE CLEANED BY RINSING IN A SUITABLE SOLVENT SUCH AS ISOPROPYL ALCOHOL. ***DO NOT USE ACETONE.*** [RINSING IS OPTIONAL AS THE ROSIN HAS NO ELECTRICAL EFFECT.] THE BOARD CAN THEN BE WASHED IN HOT WATER USING A MILD DETERGENT. RINSE IN CLEAN HOT WATER AND LET DRY.

ORIENTATION OF INTEGRATED CIRCUIT CHIPS

CARE MUST BE TAKEN TO INSURE THAT EACH INTEGRATED CIRCUIT CHIP IS SO ORIENTED, PRIOR TO INSERTION IN ITS SOCKET, THAT PIN #1 IS AT THE LOCATION SO DESIGNATED ON THE PRINTED CIRCUIT BOARD OR IN THE INDIVIDUAL ASSEMBLY INSTRUCTIONS FOR THE KIT.

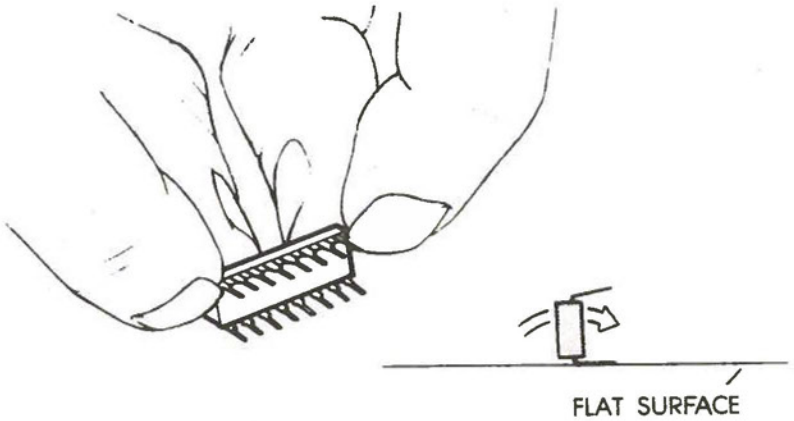
PIN #1 IS, UNFORTUNATELY, DESIGNATED IN A VARIETY OF WAYS DEPENDING UPON THE INTEGRATED CIRCUIT MANUFACTURER. SEVERAL METHODS ARE INDICATED IN THE DRAWING BELOW. WITH THE LEADS OF THE CHIP POINTING AWAY FROM THE VIEWER, PIN #1 IS IN THE POSITION INDICATED WITH RESPECT TO THE VARIOUS END NOTCHES OR TINY CIRCULAR MARKINGS OR DEPRESSIONS IN ONE CORNER.



INSERTION OF INTEGRATED CIRCUIT CHIPS

BE SURE ALL LEADS ARE STRAIGHT AND PARALLEL. IF NOT, GENTLY STRAIGHTEN AND ALIGN THE BENT PINS WITH NEEDLE NOSED PLIERS.

INTEGRATED CIRCUIT CHIPS USUALLY COME FROM THE MANUFACTURER WITH THEIR ROWS OF LEADS SPREAD WIDER THAN THE SOCKET. TO BEND THE PINS IN A UNIFORM MANNER, PLACE THE CHIP ON ITS SIDE ON A FLAT SURFACE SO THAT ONE ROW OF PINS IS FLAT AGAINST THE SURFACE AS SHOWN ON THE FOLLOWING PAGE.



HOLDING EACH SIDE OF THE CHIP FIRMLY AGAINST THE FLAT SURFACE WITH BOTH HANDS, ROTATE IT A SHORT DISTANCE UNTIL THE PINS ARE BENT PERPENDICULAR TO THE BODY.

PARTIALLY INSERT ALL ICS WITH THE PIN #1 ORIENTED AS SHOWN ON THE BOARD. THE LAYOUT SYMBOL FOR IC PIN #1 IS DESIGNATED BY A WHITE DOT. RECHECK TO INSURE THAT EACH PIN IS IN ITS HOLE AND HAS NOT BEEN FOLDED UNDER THE CHIP OR BENT OUTSIDE THE SOCKET. COMPLETE INSERTION EVENLY AND FIRMLY.

POWER ON

PLUG THE BOARD INTO YOUR COMPUTER AND CHECK IT OUT IN ACCORDANCE WITH THE USERS MANUAL FOLLOWING THESE ASSEMBLY INSTRUCTIONS.

MEMORY TEST PROGRAM

THERE ARE NUMEROUS MEMORY TEST PROGRAMS AVAILABLE IN THE LITERATURE FOR ANY LEVEL OF SYSTEM SOPHISTICATION. IF YOU HAVE 8K BASIC UP AND RUNNING, OR KNOW SOMEONE WHO DOES, THE FOLLOWING PROGRAM WILL DO A THOROUGH JOB OF TESTING YOUR MEMORY WITH A RANDOM PATTERN USING THE RND FUNCTION. TO USE THE PROGRAM, A SYSTEM WITH AT LEAST 8K OF MEMORY IS REQUIRED, NOT COUNTING THE BOARD TO BE TESTED. SET THE BOARD ADDRESS TO SOME RANGE ABOVE THE EXISTING MEMORY BUT BELOW 32K. LOAD BASIC AND INITIALIZE MEMORY AT 8192 BYTES, SO BASIC WILL NOT LOAD A PROGRAM IN THE BOARD TO BE TESTED. LOAD THE TEST PROGRAM USING THE KEYBOARD, PAPER TAPE, OR CASSETTE. RUN THE PROGRAM AND ENTER THE STARTING AND ENDING MEMORY LOCATIONS TO BE TESTED (IN DECIMAL). IT TAKES SEVERAL MINUTES TO TEST A BOARD AFTER WHICH THE PROGRAM TYPES CHECK OK AND CONTINUES TESTING. A THOROUGH TEST REQUIRES ABOUT 10 PASSES. IF AN ERROR OCCURS, THE LOCATION IS PRINTED OUT ALONG WITH THE NUMBER WRITTEN INTO MEMORY AND READ FROM MEMORY.

PROGRAM LISTING (MITS BASIC)

```
30 INPUT"HIGH MEMORY ADD.";H
70 INPUT"LOW MEMORY ADD.";L
121 PRINT"LOCATION","WROTE","READ"
122 A=RND(1)
125 B=RND(-A)
130 FOR N=L TO H
140 POKE N,INT(256*RND(1))
150 NEXT
160 B=RND (-A)
170 FOR N=L TO H
180 IF PEEK(N)=INT(256*RND(1) ) GOTO 200
190 PRINT N,INT(256*RND(0)),PEEK(N)
200 NEXT
210 PRINT"CHECK OK"
220 GOTO 122
OK
```

EXAMPLE RUN

```
RUN
HIGH MEMORY ADD.? 20479
LOW MEMORY ADD.? 8192
LOCATION        WROTE           READ
CHECK OK
CHECK OK
CHECK OK
```

THEORY OF OPERATION

THE BOARD OCCUPIES A 4K ADDRESS SLOT, THEREFORE ADDRESS LINES A12 TO A15 ARE DECODED TO ENABLE THE BOARD. EXCLUSIVE OR GATE C5 INVERTS THE ADDRESS LINES IF THE DIP SWITCH CONTACTS ARE OPEN, SO THAT FOR THE SELECTED ADDRESS RANGE, C4 PIN 8 GOES LOW. (IF OPTIONAL DIP SWITCH IS NOT INSTALLED, TRACES ON THE BOARD SELECT ADDRESS C000). THE SECOND HALF OF C4 GATES THE INVERTED BOARD SELECT SIGNAL WITH SINP AND SOUT TO ENABLE THE BOARD. THIS SIGNAL ACTIVATES THE TRI-STATE BUS DRIVER TO PULL THE PRDY LINE LOW FOR A SELECTABLE NUMBER OF CLOCK CYCLES DETERMINED BY C6 CAUSING THE MPU TO ENTER A WAIT STATE. THE BOARD ENABLE SIGNAL IS GATED WITH PDBIN AND SMEMR TO ACTIVATE THE BUS DRIVERS, PLACING DATA FROM THE ROM OR RAM ON THE DATA IN BUS.

ADDRESS LINES A0 - A7 ARE CONNECTED TO BOTH THE PROM AND RAM. A8 AND A9 ARE ALSO CONNECTED TO THE RAM WHICH HAS 1024 LOCATIONS, BUT SINCE THE PROMS HAVE ONLY 256 ADDRESSABLE LOCATIONS, B1 IS USED TO SELECT ONE OF EIGHT CHIPS, COVERING 2K OF MEMORY. C1 PIN 3 GOES LOW IF A10 AND A11 ARE BOTH HIGH TO ENABLE RAM IN THE TOP 1K ADDRESS SLOT. IT WAS NOT CONSIDERED NECESSARY TO BUFFER THE ADDRESS LINES SINCE THERE ARE ONLY ONE FOURTH AS MANY CHIPS AS ON AN 8K MEMORY BOARD, AND MORE THAN ONE OF THESE BOARDS IS RARELY USED IN A SYSTEM. THE DATA OUT BUS IS CONNECTED TO THE DATA IN PINS OF THE APPROPRIATE RAM CHIP.

THE JUMP-ON-RESET FEATURE IS CONTROLLED BY THE JUMP FLIP-FLOP FORMED BY C1 (PIN 6 AND 11). WHEN THE PRESET LINE GOES LOW, C1 PIN 11 GOES LOW, CAUSING THE BOARD TO BE ENABLED AT ANY ADDRESS. AT THE SAME TIME, BUS LINE 67 IS PULLED LOW, DISABLING THE BUS DRIVERS OF THE VECTOR GRAPHIC 8K RAM BOARDS, WHICH MUST HAVE THE OUTPUT DISABLE JUMPER IN PLACE. SINCE THE PRESET CAUSES THE MPU TO ZERO THE PROGRAM COUNTER, PROGRAM EXECUTION BEGINS AT LOCATION ZERO WHEN THIS LINE GOES HIGH. SINCE THE PROM/RAM BOARD IS ENABLED, THE INSTRUCTION FETCHED IS THE FIRST CONTAINED IN THE PAGE 0 PROM. THIS INSTRUCTION SHOULD BE JMP X003, WHERE X CORRESPONDS TO THE SETTING OF THE DIP SWITCH OR JUMPERS. THE BOARD IS NORMALLY PRE-JUMPERED FOR C000. RESPONSE TO THIS FIRST INSTRUCTION CAUSES THE MPU TO SUBSTITUTE X003 IN THE PROGRAM COUNTER, AND FETCH THE NEXT INSTRUCTION AT X003, WHICH, OF COURSE, IS THE NEXT INSTRUCTION IN PROM. C4 PIN 8 DECODES THIS ADDRESS AND GOES LOW, CAUSING THE JUMP FLIP-FLOP (C1 PINS 6 AND 11) TO RESET, RESTORING NORMAL OPERATION OF THE 8K RAM BUS DRIVERS AND THE PROM/RAM ADDRESS DECODING. PROGRAM EXECUTION CONTINUES IN PROM AT THE NORMAL ADDRESS FOR WHICH THE PROGRAM IS ASSEMBLED. NOTE THAT THIS JUMP TECHNIQUE DOES NOT INTERFERE WITH PROGRAM STORED IN RAM AT LOCATION 0, AND IT IS NOT RESTRICTED TO A PARTICULAR OP CODE SET AS ARE THE USUAL HARDWIRED JAM TECHNIQUES. IF YOU DESIRE TO USE THIS FEATURE WITH ANOTHER TYPE OF MICROPROCESSOR, THE PROM CAN BE REPLACED WITH ONE CONTAINING ITS OP CODES.

THE ONLY LOGIC ON THE FRONT PANEL OF IMSAI AND "ALTAIR"™ COMPUTERS FOR NORMAL OPERATION OF THE COMPUTER IS GATING OF THE PWR SIGNAL AND SOUT TO PRODUCE THE MWRITE SIGNAL. THIS LOGIC IS PROVIDED AT C2 PIN 6 AND CAN OPTIONALLY BE CONNECTED BY JUMPERING BETWEEN PADS 10 AND 11 (THE BOARD IS NOT PREJUMPERED BETWEEN THESE PADS). THIS FEATURE SHOULD NOT BE USED WITH A COMPUTER HAVING FRONT PANEL LOGIC, SINCE IT WILL CONFLICT WITH OPERATION OF THE FRONT PANEL.

IF MORE THAN ONE PROM/RAM BOARD IS USED IN A SYSTEM, THE JUMP FEATURE MUST BE DISABLED ON ALL BUT ONE OF THE BOARDS BY CUTTING THE TRACES BETWEEN PADS 6 AND 7 AND 8 AND 9.

THE NUMBER OF WAIT STATES IS PREJUMPERED AT 1. THIS SHOULD BE ADEQUATE FOR VIRTUALLY ALL 1702 A'S. HOWEVER IF YOU WISH TO INCREASE THE NUMBER OF WAIT STATES, CUT THE TRACE BETWEEN PAD W AND PAD 1 IN THE LOWER RIGHT HAND CORNER AND CONNECT A JUMPER BETWEEN W AND THE APPROPRIATE WAIT STATES. THE BOARD MUST HAVE AT LEAST 1 WAIT STATE.

A VARIETY OF PROGRAMS ON PROM ARE AVAILABLE FROM VECTOR GRAPHIC INC. PLEASE SEE YOUR DEALER FOR OUR CATALOG.

POWER SUPPLY CONSIDERATION

FOR RELIABLE OPERATION, AN ADEQUATE, UNREGULATED 8 VOLT SUPPLY MUST BE PROVIDED. THE REGULATORS ON THE PROM/RAM REQUIRE AT LEAST 2 VOLTS DROP TO REGULATE PROPERLY. THIS MEANS THAT THE TROUGH OF THE UNREGULATED SUPPLY WAVEFORM MUST BE AT LEAST 7 VOLTS. TO ALLOW FOR NORMAL LINE VOLTAGE FLUCTUATIONS, AT LEAST 10% MARGIN SHOULD BE MAINTAINED ABOVE THIS. THUS WITH 1 VOLT PEAK-PEAK RIPPLE, THE AVERAGE UNREGULATED SUPPLY VOLTAGE SHOULD BE AT LEAST 8.2 VOLTS. TO MAINTAIN LESS THAN 1 VOLT P-P RIPPLE, AT LEAST 8000 MFD OF FILTER CAPACITANCE SHOULD BE PROVIDED PER AMPERE OF TOTAL CURRENT DRAIN. IF YOUR COMPUTER SUPPLY IS NOT ADEQUATE, WE OFFER A REPLACEMENT POWER TRANSFORMER WHICH WILL PRODUCE +8V, 18A, ±16V, 2.5A CONTACT US FOR FURTHER INFORMATION.

LINE TRANSIENTS

MOST OF US HAVE EXPERIENCED THE FRUSTRATION OF SPENDING A LOT OF TIME WORKING ON A PROGRAM, ONLY TO HAVE A POWER LINE TRANSIENT CAUSE THE PROGRAM TO BOMB. THIS PROBLEM IS USUALLY DUE TO HIGH FREQUENCY TRANSIENTS CAUSED BY MOTOR STARTING CONTACTORS OR INDUCTIVE ENERGY STORAGE SOMEWHERE ON THE POWER DISTRIBUTION SYSTEM. ACTUAL POWER OUTAGES ARE RELATIVELY RARE. MEMORY WRITE PROTECTION OR STANDBY POWER SOURCES WILL NOT PREVENT THIS PROBLEM. IT IS RECOMMENDED THAT A POWER LINE FILTER BE INSTALLED IN YOUR COMPUTER AS CLOSE TO THE LINE CORD ENTRY POINT AS POSSIBLE. A CORCOM MODEL 3B1 OR EQUIVALENT IS VERY EFFECTIVE. THE VECTOR 1 HAS A POWER LINE FILTER.

VENTILATION

IT IS RECOMMENDED THAT ADEQUATE FORCED VENTILATION BE PROVIDED IN ENCLOSED CABINETS. IF THE COMPUTER IS OPERATED WITHOUT A COVER, ALLOW 2 SLOTS SEPARATION OR 1.5" BETWEEN BOARDS. IF YOU CAN'T HOLD YOUR FINGER ON THE HEAT SINK FOR AT LEAST A FEW SECONDS, THE VENTILATION IS NOT ADEQUATE.

PROM/RAM BOARD TROUBLE SHOOTING HINTS

ASSUMING YOU HAVE CHECKED THE +5V AND -9V REGULATORS FOR PROPER OPERATION, TURN OFF POWER, AND INSTALL THE MONITOR PROMS IN LOCATION A1 AND A2. IF THE COMPUTER FAILS TO RESPOND WITH A PROMPT WITH POWER-ON-RESET, THEN REVIEW THE GENERAL TROUBLE SHOOTING GUIDE FOR THE COMPUTER. IF THE PROBLEM CAN BE ISOLATED TO THE PROM/RAM BOARD, THE JUMPER BETWEEN PADS 10 AND 11 IS IN PLACE, AND THE JUMPER TO PIN 67 OF THE RAM BOARD AT ADDRESS ZERO IS IN PLACE, YOU MAY HAVE A DEFECTIVE CHIP. IF YOU HAVE ACCESS TO ANOTHER PROM/RAM BOARD, CHANGE THE ADDRESS JUMPERING TO E000H ON THE DEFECTIVE BOARD BY INSTALLING A JUMPER IN THE A13 POSITION. IT SHOULD NOW BE POSSIBLE TO DISPLAY THE MONITOR PROGRAM IN THE DEFECTIVE BOARD USING THE GOOD BOARD AND TO COMPARE THE CHECKSUM USING THE W COMMAND. THE RAM ON THE DEFECTIVE BOARD CAN BE TESTED FROM EC00H TO EFFFH USING THE T COMMAND (T EC00 EFFF). IF THIS FAILS TO REVEAL THE PROBLEM, ANOTHER TECHNIQUE IS TO REMOVE THE 8097 BUS DRIVERS AND THE JUMPER BETWEEN PAD 10 TO 11 FROM THE DEFECTIVE BOARD, ADDRESS IT IN THE SAME LOCATION AS THE GOOD BOARD, AND THEN COMPARE WAVEFORMS AT DIFFERENT NODES ON EACH BOARD. DUE TO THE SIMPLICITY OF THE CIRCUIT, PROBLEMS BEYOND THIS POINT ARE VERY UNUSUAL.

MACHINE LANGUAGE TEST PROGRAM

THE MACHINE LANGUAGE MEMORY TEST PROGRAM ON THE FOLLOWING PAGES IS ABSTRACTED FROM THE VECTOR I MONITOR PROGRAM, AND ASSEMBLED TO RUN IN THE LOWEST 256 BYTES OF MEMORY. START EXECUTION AT ADDRESS 0000H. A "*" WILL BE TYPED IF YOU HAVE PROPERLY PATCHED THE I/O ROUTINES FOR YOUR SYSTEM. PTCN IS THE OUTPUT ROUTINE FOR A 3P+S BOARD WITH STATUS INVERTED. (OR MITS REV I SIO) RDCN IS THE INPUT ROUTINE. IF YOU ARE USING A BOARD WITH A PROGRAMMABLE USART, YOU WILL HAVE TO INITIALIZE IT IN ADDITION TO CHANGING THE MASK, JUMP CONDITION, AND PORT.

AFTER *, TYPE IN FOUR HEX CHARACTERS FOR THE LENGTH OF THE MEMORY BLOCK TO BE TESTED (2000 FOR 8K) AND FOUR CHARACTERS FOR THE STARTING ADDRESS OF THE BLOCK. SPACE IS AUTOMATIC, AND IF YOU TYPE ANY CHARACTERS OTHER THAN 0-9, A-F THE PROGRAM WILL DO STRANGE THINGS. A RESET WILL TERMINATE THE TEST. THE PROGRAM GENERATES A 2^{16}-1 BYTE PSEUDORANDOM NUMBER SEQUENCE, WRITES A PORTION OF IT IN THE BLOCK OF MEMORY AND THEN REGENERATES THE SEQUENCES FROM THE SAME POINT TO COMPARE WITH WHAT IS READ FROM MEMORY. IF THE PASS IS CORRECT, A NEW PORTION OF THE SEQUENCE IS WRITTEN INTO MEMORY. ERRORS ARE PRINTED OUT WITH THE ADDRESS, WHAT WAS WRITTEN, AND WHAT WAS READ. USE THE ADDRESS LOCATIONS ON THE COMPONENT PLACEMENT DIAGRAM TO LOCATE THE BAD ROW, AND THE INCORRECT BIT TO LOCATE THE COLUMN. AN OUTPUT OF FF MEANS NO MEMORY, MORE THAN ONE BIT WRONG IS USUALLY CAUSED BY CHIPS IN BACKWARDS (WHICH DOES NOT DESTROY THE MEMORY CHIPS, CONTRARY TO TTL) OR A SOLDER BRIDGE. BENT UNDER ADDRESS PINS CAUSE MANY ERRORS TO BE PRINTED OUT IN ONE 1K BLOCK.

THE MOST DIFFICULT PROBLEM TO ISOLATE IS A SHORT CIRCUITED ADDRESS LINE TO THE MEMORY ARRAY. THIS WILL USUALLY CAUSE ALL MEMORY LOCATIONS TO INDICATE ERROR WITH ALL BITS BAD. THE SHORT CAN BE CAUSED BY A SOLDER BRIDGE, AN ETCH BRIDGE (ALTHOUGH EACH BOARD IS ELECTRICALLY TESTED FOR THIS), OR A DEFECTIVE CHIP. IF YOU CAN NOT LOCATE THE PROBLEM VISUALLY, REMOVE HALF OF THE ROWS OF CHIPS AND TEST WITH A SMALLER BLOCK LENGTH. REPEAT THIS UNTIL ALL CHIPS HAVE BEEN ELIMINATED AS TROUBLE MAKERS. THEN TEST BETWEEN MEMORY SOCKET PINS USING A LOW VOLTAGE OHMMETER ON THE XI OHMS SCALE AT ONE CHIP LOCATION. IF THIS FAILS TO REVEAL THE PROBLEM, SOME EXPERIENCE IN TROUBLESHOOTING ELECTRONIC CIRCUITS BECOMES VERY USEFUL.

MEMORY TEST PROGRAM FOR LOCATION 0000H TO 00FFH

```
0000                    0010 CONC   EQU   0              CONSOLE STAT PORT
0000                    0020 COND   EQU   1              CONSOLE DATA PORT
0000                    0030 SPTR   EQU   0100H          STACK POINTER
0000 31 00 01           0040 START  LXI   SP,SPTR
0003 CD 37 00           0050        CALL  CRLF
0006 3E 2A              0060       MVI   A,'*'          PRINT "*"
0008 CD 2B 00           0070        CALL  PTCN
000B C3 4F 00           0080        JMP   TMEM
000E                    0090 *
000E                    0100 *** CONVERT UP TO 4 HEX DIGITS TO BIN
000E                    0110 *
000E 21 00 00           0120 AHEX   LXI   H,0            GET 16 BIT ZERO
0011 0E 04              0130        MVI   C,4            COUNT OF 4 DIGITS
0013 CD 41 00           0140 AHE1   CALL  RDCN           READ A BYTE
0016 29                 0150        DAD   H              SHIFT 4 LEFT
0017 29                 0160        DAD   H
0018 29                 0170        DAD   H
0019 29                 0180        DAD   H
001A D6 30              0190        SUI   48             ASCII BIAS
001C FE 0A              0200        CPI   10             DIGIT 0-10
001E DA 23 00           0210        JC    ALF
0021 D6 07              0220        SUI   7              ALPHA BIAS
0023 85                 0230 ALF    ADD   L
0024 6F                 0240        MOV   L,A
0025 0D                 0250        DCR   C              4 DIGITS?
0026 C2 13 00           0260        JNZ   AHE1           KEEP READING
0029 3E 20              0270 SPCE   MVI   A,20H         PRINT SPACE
002B F5                 0280 PTCN   PUSH  PSW            SAVE REG A
002C DB 00              0290 PTLOP  IN    CONC           READ PRTR STATUS
002E E6 80              0300        ANI   80H            IF BIT 7 NOT 0,
0030 C2 2C 00           0310        JNZ   PTLOP          WAIT TILL TIS
0033 F1                 0320        POP   PSW            THEN RECOVER A
0034 D3 01              0330        OUT   COND           AND PRINT IT
0036 C9                 0340        RET   RETURN         FROM PTCN
0037 3E 0D              0350 CRLF   MVI   A,0DH          PRINT CR
0039 CD 2B 00           0360        CALL  PTCN
003C 3E 0A              0370        MVI   A,0AH
003E C3 2B 00           0380        JMP   PTCN
0041                    0390 *
0041                    0400 *** READ FROM CONSOLE TO REG A ***
0041                    0410 *
0041 DB 00              0420 RDCN   IN    CONC           READ KB STATUS
0043 E6 01              0430        ANI   1              IF BIT 1 NOT 0
0045 C2 41 00           0440        JNZ   RDCN           REPEAT UNTIL IT IS
0048 DB 01              0450        IN    COND           READ FROM KB
004A E6 7F              0460        ANI   7FH            STRIP OFF MSB
004C C3 2B 00           0470        JMP   PTCN           ECHO ONTO PRINTER
004F                    0480 *
004F                    0490 *** MEMORY TEST ROUTINE ***
004F                    0500 *
004F CD 0E 00           0510 TMEM   CALL  AHEX           READ BLK LEN
```

```
0052 EB                  0520          XCHG                    PUT IN D,E
0053 CD 0E 00            0530          CALL  AHEX              READ ST ADD
0056 01 5A 5A            0540          LXI   B,5A5AH           INI B,C
0059 CD 83 00            0550 CYCL     CALL  RNDM
005C C5                  0560          PUSH  B                 KEEP ALL REGS
005D E5                  0570          PUSH  H
005E D5                  0580          PUSH  D
005F CD 83 00            0590 TLOP     CALL  RNDM
0062 70                  0600          MOV   M,B               WRITE IN MEM
0063 23                  0610          INX   H                 INC POINTER
0064 1B                  0620          DCX   D                 DECR COUNTER
0065 7A                  0630          MOV   A,D               CHECK D,E
0066 B3                  0640          ORA   E                 FOR ZERO
0067 C2 5F 00            0650          JNZ   TLOP              REPEAT LOOP
006A D1                  0660          POP   D
006B E1                  0670          POP   H                 RESTORE ORIG
006C C1                  0680          POP   B                 VALUES OF
006D E5                  0690          PUSH  H
006E D5                  0700          PUSH  D
006F CD 83 00            0710 RLOP     CALL  RNDM              GEN NEW SEQ
0072 7E                  0720          MOV   A,M               READ MEM
0073 B8                  0730          CMP   B                 COMP MEM
0074 C4 A4 00            0740          CNZ   ERR               CALL ERROR ROUT
0077 23                  0750          INX   H
0078 1B                  0760          DCX   D
0079 7A                  0770          MOV   A,D
007A B3                  0780          ORA   E
007B C2 6F 00            0790          JNZ   RLOP
007E D1                  0800          POP   D
007F E1                  0810          POP   H
0080 C3 59 00            0820          JMP   CYCL
0083                     0830 *** THIS ROUTINE GENERATES RANDOM NOS ***
0083 78                  0840 RNDM     MOV   A,B               LOOK AT B
0084 E6 B4               0850          ANI   0B4H              MASK BITS
0086 A7                  0860          ANA   A                 CLEAR CY
0087 EA 8B 00            0870          JPE   PEVE              JUMP IF EVEN
008A 37                  0880          STC
008B 79                  0890 PEVE     MOV   A,C               LOOK AT C
008C 17                  0900          RAL                     ROTATE CY IN
008D 4F                  0910          MOV   C,A               RESTORE C
008E 78                  0920          MOV   A,B               LOOK AT B
008F 17                  0930          RAL                     ROTATE CY IN
0090 47                  0940          MOV   B,A               RESTORE B
0091 C9                  0950          RET                     RETURN W NEW B,C
0092                     0960 *
0092                     0970 *** ERROR PRINT OUT ROUTINE
0092                     0980 *
0092 CD 37 00            0990 PTAD     CALL  CRLF              PRINT CR,LF
0095 7C                  1000          MOV   A,H               PRINT
0096 CD B3 00            1010          CALL  PT2               ASCII
0099 7D                  1020          MOV   A,L               CODES
009A CD B3 00            1030          CALL  PT2               FOR
009D CD 29 00            1040          CALL  SPCE              ADDRESS
00A0 CD 29 00            1050          CALL  SPCE
00A3 C9                  1060          RET
00A4 F5                  1070 ERR      PUSH  PSW               SAVE ACC
00A5 CD 92 00            1080          CALL  PTAD              PRINT ADD.
00A8 78                  1090          MOV   A,B               DATA
00A9 CD B3 00            1100          CALL  PT2               WRITTEN
00AC CD 29 00            1110          CALL  SPCE
00AF CD 29 00            1120          CALL  SPCE
```

```
00B2 F1                  1130         POP   PSW          DATA READ
00B3 F5                  1140 PT2     PUSH  PSW
00B4 CD BB 00            1150         CALL  BINH
00B7 F1                  1160         POP   PSW
00B8 C3 BF 00            1170         JMP   BINL
00BB 1F                  1180 BINH    RAR
00BC 1F                  1190         RAR
00BD 1F                  1200         RAR
00BE 1F                  1210         RAR
00BF E6 0F               1220 BINL    ANI   0FH          LOW 4 BITS
00C1 C6 30               1230         ADI   48           ASCII BIAS
00C3 FE 3A               1240         CPI   58           DIGIT 0-9
00C5 DA 2B 00            1250         JC    PTCN
00C8 C6 07               1260         ADI   7            DIGIT A-F
00CA C3 2B 00            1270         JMP   PTCN

SYMBOL TABLE

AHE1  0013   AHEX  000E   ALF   0023   BINH  00BB   BINL  00BF   CONC  0000
COND  0001   CRLF  0037   CYCL  0059   ERR   00A4   PEVE  008B   PT2   00B3
PTAD  0092   PTCN  002B   PTLOP 002C   RDCN  0041   RLOP  006F   RNDM  0083
SPCE  0029   SPTR  0100   START 0000   TLOP  005F   TMEM  004F


D 0000 00CF
0000 31 00 01 CD 37 00 3E 2A CD 2B 00 C3 4F 00 21 00
0010 00 0E 04 CD 41 00 29 29 29 29 D6 30 FE 0A DA 23
0020 00 D6 07 85 6F 0D C2 13 00 3E 20 F5 DB 00 E6 80
0030 C2 2C 00 F1 D3 01 C9 3E 0D CD 2B 00 3E 0A C3 2B
0040 00 DB 00 E6 01 C2 41 00 DB 01 E6 7F C3 2B 00 CD
0050 0E 00 EB CD 0E 00 01 5A 5A CD 83 00 C5 E5 D5 CD
0060 83 00 70 23 1B 7A B3 C2 5F 00 D1 E1 C1 E5 D5 CD
0070 83 00 7E B8 C4 A4 00 23 1B 7A B3 C2 6F 00 D1 E1
0080 C3 59 00 78 E6 B4 A7 EA 8B 00 37 79 17 4F 78 17
0090 47 C9 CD 37 00 7C CD B3 00 7D CD B3 00 CD 29 00
00A0 CD 29 00 C9 F5 CD 92 00 78 CD B3 00 CD 29 00 CD
00B0 29 00 F1 F5 CD BB 00 F1 C3 BF 00 1F 1F 1F 1F E6
00C0 0F C6 30 FE 3A DA 2B 00 C6 07 C3 2B 00 2B 00 C6
```

EXPERIMENTING WITH YOUR NEW COMPUTER

NOW THAT YOUR SHINY NEW COMPUTER IS ASSEMBLED AND CHECKED OUT, WHAT IS THE NEXT STEP? IF YOU HAVE NOT ALREADY DONE SO, YOU SHOULD READ THE INTEL 8080 MICROCOMPUTER SYSTEMS USER'S MANUAL AND BECOME FAMILIAR WITH THE INSTRUCTION SET AND EXACTLY WHAT GOES ON IN THE CPU CHIP FROM A PROGRAMMERS POINT OF VIEW. THE NEXT STEP WOULD BE TO TRY YOUR HAND AT SOME SIMPLE ASSEMBLY LANGUAGE PROGRAMS. LENGTHY PROGRAMS ARE USUALLY WRITTEN WITH THE AID OF AN ASSEMBLER PROGRAM WHICH ENORMOUSLY SIMPLIRES THE TASK OF MAKING CHANGES IN THE PROGRAM, SUCH AS ESP-1 WHICH IS AVAILABLE FROM VECTOR GRAPHIC INC. AT A NOMINAL CHARGE.

SHORT PROGRAMS CAN BE CODED BY HAND USING AN 8080 PROGRAMMING CARD AND THEN ENTERED IN THE COMPUTER MEMORY USING THE VECTOR 1 MONITOR. ASSEMBLY LANGUAGE PROGRAMMING CONSISTS OF BUILDING A PROGRAM USING GENERAL PURPOSE SUBROUTINES AS BUILDING BLOCKS. MOST PROGRAMS HAVE ROUTINES THAT READ THE KEYBOARD, OUTPUT TO A PRINTER, CONVERT FROM HEX TO BINARY AND BACK, COMPARE ADDRESSES AND SO ON. AN EXPERIENCED PROGRAMMER WILL HAVE A COLLECTION OF THESE ROUTINES IN HIS "BAG OF TRICKS" THAT HE CAN INSERT IN A PROGRAM WHEN NEEDED. THE DIFFICULT PART IS TO BE ABLE TO QUICKLY SCAN THROUGH THE ROUTINE AND UNDERSTAND EXACTLY WHAT IT DOES, HOW DATA IS PASSED BACK AND FORTH, AND WHICH REGISTERS ARE USED TO SEE IF IT INTERFERES WITH THE USE OF REGISTERS IN THE CALLING ROUTINE. IF THERE IS A CONFLICT, THE REGISTER CONTENTS MUST BE PUSHED ON THE STACK BEFORE THE ROUTINE IS CALLED AND POPPED BACK AFTER A RETURN.

A USEFUL COLLECTION OF SUBROUTINES IS CONTAINED IN THE VECTOR 1 MONITOR, AND THEY CAN BE CALLED BY ANY PROGRAM YOU WISH TO WRITE. AN EXAMPLE OF A SHORT PROGRAM CALLED SRCH IS SHOWN IN RGURE 1. THE PURPOSE OF SRCH IS TO LOOK FOR SPECIRC INSTRUCTIONS SUCH AS INPUT OR OUTPUT COMMANDS IN A LARGE PROGRAM. THIS PROGRAM WAS ASSEMBLED USING ESP-1 TO RUN IN RAM ON THE PROM/RAM BOARD AND CALLS SUBROUTINES FROM THE MONITOR. THE PROGRAM IS TYPED IN USING LINE NUMBERS TO IDENTIFY LINES IN THE RLE. THE RRST INSTRUCTION IN CALL AHEX, A SUBROUTINE IN THE MONITOR THAT INPUTS FOUR HEX DIGITS FROM THE KEYBOARD, ECHOES THEM TO THE PRINTER, CONVERTS THEM TO A 16 BIT BINARY ADDRESS IN REGISTERS H & L AND EXCHANGES H & L WITH D & E (REFER TO MONITOR LISTING). TWO SUCCESSIVE CALLS TO AHEX RESULT IN A STARTING ADDRESS IN H & L, AND AN ENDING ADDRESS IN D & E. THE NEXT INSTRUCTIONS SAVE H, SET UP REGISTERS TO CONVERT ONLY 2 CHARACTERS TO BINARY AND THEN CALL A PORTION OF AHEX TO INPUT A TWO DIGIT INSTRUCTION CODE FROM THE KEYBOARD. THIS CODE IS PUT IN REGISTER B, AND H IS RESTORED.

THE NEXT BLOCK OF INSTRUCTIONS IS REPEATED OVER AND OVER, SO A LABEL CONT IS GIVEN TO THIS POINT IN THE PROGRAM. MEMORY IS READ USING THE ADDRESS IN H & L AND COMPARED TO THE DESIRED OP CODE. IF THEY ARE NOT THE SAME, THE PROGRAM JUMPS TO SKP. IF THEY ARE THE SAME, PROGRAM EXECUTION PROCEEDS BY READING THE NEXT MEMORY LOCATION AND CALLING ERR WHICH PRINTS THE ADDRESS, OP CODE AND NEXT CODE IN THE PROPER FORMAT. BMP COMPARES THE CURRENT ADDRESS WITH THE RNISH ADDRESS IN D & E TO SEE IF IT IS TIME TO STOP, AND IF NOT, THE PROGRAM JUMPS BACK TO CONT TO CONTINUE THE SEARCH.

STARTING AT LINE 0200 ARE FOUR INSTRUCTIONS CALLED PSEUDO OP CODES THAT SERVE TO GIVE THE ASSEMBLER ADDITIONAL INFORMATION IT NEEDS, NAMELY WHERE THE SUBROUTINES ARE ACTUALLY LOCATED. THE PARTICULAR ASSEMBLER USED REQUIRES THAT THE ADDRESSES IN HEX BE PRECEDED BY A 0 AND FOLLOWED BY H TO DENOTE HEX. NO OBJECT CODE IS GENERATED BY THESE INSTRUCTIONS. THE CODE PRODUCED BY THE ASSEMBLER IS SHOWN ON THE LEFT OF THE LISTING FOLLOWING THE 4 DIGIT HEX MEMORY LOCATION. MANY OF THE INSTRUCTIONS GENERATE MULTIBYTE CODES, AND THESE ARE LOADED IN SUBSEQUENT MEMORY LOCATIONS.

THE ASSEMBLER PRINTS AN ALPHABETICAL TABLE OF ALL THE LABELS USED IN THE PROGRAM FOLLOWED BY THE CORRESPONDING ADDRESS, SO THAT THESE POINTS CAN BE REFERENCED IN SUBSEQUENT PROGRAMS. BELOW THE SYMBOL TABLE, THE PROGRAM WAS EXECUTED BY TYPING G CC00 FROM THE MONITOR. THE ADDRESS RANGE OF C000 TO CIFF (THE MONITOR PROGRAM) WAS ENTERED AND THEN D3, THE 8080 CODE FOR "OUT". THE PROGRAM RESPONDED BY PRINTING OUT ALL LOCATIONS WHERE THE OUTPUT INSTRUCTION OCCURRED IN THE MONITOR PROGRAM FOLLOWED BY THE PORT NUMBER. YOU CAN TRY THIS ON YOUR SYSTEM BY ENTERING THE OBJECT CODE IN THE PROPER MEMORY LOCATION USING THE "P" MONITOR COMMAND.

FIGURE 1 EXPERIMENTING WITH YOUR NEW COMPUTER

```
A CC00
  MEM  LOC                LINE
                          NO.   LABEL
CC00 CD 57 C0             0010  SRCH    CALL  AHEX        START
CC03 CD 57 C0             0020          CALL  AHEX        FINISH(S=H,F=D)
CC06 E5                   0030          PUSH  H           SAVE H
CC07 2E 00                0040          MVI   L,0
CC09 0E 02                0050          MVI   C,2         COUNT OF 2
CC0B CD 5C C0             0060          CALL  AHE1        READ 2 DIGITS
CC0E EB                   0070          XCHG              H=CODE,D=F
CC0F 45                   0080          MOV   B,L         PUT CODE IN B
CC10 E1                   0090          POP   H           RESTORE H
CC11 7E                   0100  CONT    MOV   A,M         READ MEMORY
CC12 B8                   0110          CMP   B           COMPARE TO CODE
CC13 C2 1C CC             0120          JNZ   SKP         SKIP IF NO COMP
CC16 23                   0130          INX   H           INCR ADDRESS
CC17 7E                   0140          MOV   A,M         READ NEXT BYTE
CC18 2B                   0150          DCX   H           DECR ADDRESS
CC19 CD 68 C1             0160          CALL  ERR         PRINT CODES
CC1C CD F5 C1             0170  SKP     CALL  BMP         CHECK IF DONE
CC1F C2 11 CC             0180          JNZ   CONT        BACK FOR MORE
CC22 C9                   0190          RET
CC23                      0200  BMP     EQU   0C1F5H
CC23                      0210  ERR     EQU   0C168H
CC23                      0220  AHE1    EQU   0C05CH
CC23                      0230  AHEX    EQU   0C057H

SYMBOL TABLE

AHE1  C05C   AHEX  C057   BMP   C1F5   CONT  CC11   ERR   C168   SKP   CC1C
SRCH  CC00

G C000

*G CC00 C000 C1FF D3
C008 D3 10
C00C D3 10
C07E D3 01
C0C8 D3 6F
C0CE D3 6E
*G CC00 C000 C1FF DB
C076 DB 00
C08B DB 00
C092 DB 01
C0C0 DB 6E
C0EE DB C0
C10F DB 6E
C116 DB 6F
*
```

VECTOR 1 MONITOR - VERSION 1.2

THE 512 BYTE MONITOR FOR VECTOR 1 IS DESIGNED AS A MINIMUM OPERATING SYSTEM TO ALLOW RAPID SYSTEM CHECKOUT, TAPE LOADING AND CONSOLE PROGRAMMING. NINE COMMANDS ARE AVAILABLE WITH THE FORMAT SHOWN ON THE PROGRAM LISTING. THE MONITOR RESPONDS WITH A "*" ON RESET, AND ONE OF NINE LETTERS MAY BE TYPED. IF THE MONITOR RECOGNIZES THE LETTER, A FOUR DIGIT HEX ADDRESS MAY BE ENTERED AFTER WHICH A SPACE IS AUTOMATICALLY TYPED. EXAMPLES OF THE USE OF THE COMMANDS ARE SHOWN BELOW.

G GOES TO A LOCATION AND EXECUTES THE PROGRAM. IF THE PROGRAM ENDS IN RET, EXECUTION REVERTS BACK TO THE MONITOR.

D DISPLAYS MEMORY CONTENTS FROM SSSS TO FFFF IN HEX FORMAT. TO TERMINATE A DUMP, PUSH THE RESET BUTTON.

P RESPONDS BY PRINTING THE CONTENTS OF MEMORY LOCATION LLLL AND THEN A DASH. TYPING TWO HEX DIGITS WILL CAUSE THAT NUMBER TO BE SUBSTITUTED IN MEMORY AND THE NEXT MEMORY LOCATION TO BE PRINTED OUT. A BACK SLASH WILL TERMINATE THE SEQUENCE, WHILE A CARRIAGE RETURN WILL ONLY HAVE THE USUAL EFFECT.

T WILL TEST MEMORY BETWEEN THE SPECIRED LOCATIONS USING A PSEUDORANDOM SEQUENCE. ANY ERRORS WILL BE PRINTED OUT WITHIN A FEW SECONDS. ANY MEMORY LOCATION CAN BE TESTED EXCEPT THE AREA USED FOR THE MONITOR STACK JUST BELOW CFFF.

THE TAPE CASSETTE ROUTINES ARE FOR THE TARBELL CASSETTE INTERFACE AND ARE DERIVED FROM THOSE SUPPLIED WITH THE INTERFACE. R WILL READ A BLOCK OF DATA INTO MEMORY BETWEEN THE SPECIRED LOCATIONS. THE CHECKSUM IS PRINTED OUT AFTER THE TAPE IS READ, AND E IS PRINTED IF THE CHECKSUM IS NOT CORRECT. NOTE THAT THE ADDRESS FORMAT IS DIFFERENT THAN FOR THE TARELL ROUTINES. A TAPE DUMPED WITH O 1300 ED00 USING THE TARBELL PROGRAM WILL BE READ CORRECTLY USING R ED00 FFFF, I.E. ADD THE BLOCK LENGTH LESS 1 TO THE STARTING ADDRESS TO OBTAIN THE ENDING ADDRESS. THE SAME DATA CAN BE WRITTEN ON CASSETTE USING W ED00 FFFF WITH THE VECTOR 1 MONITOR. THE CHECKSUM IS PRINTED OUT AFTER THE DATA IS RECORDED, AND THIS FEATURE IS USEFUL TO VERIFY THE INTEGRITY OF DATA IN MEMORY WHILE DEVELOPING ASSEMBLY LANGUAGE PROGRAM. FOR EXAMPLE, ASSUME THAT A PROGRAM HAS GONE HAYWIRE AND YOU WISH TO SEE IF A RLE OR ASSEMBLER HAS BEEN DESTROYED, SIMPLY OUTPUT THE BLOCK OF DATA TO CASSETTE WITHOUT STARTING THE RECORDER. IF THE CHECKSUM IS THE SAME AS WHEN THE DATA WAS READ IN, YOU ARE 99 AND 61/100 PERCENT SURE IT IS INTACT. THIS FEATURE CAN ALSO BE USED TO COMPARE TWO BLOCKS OF IDENTICAL DATA. NOTE THAT DATA WRITTEN ON CASSETTE CAN BE READ BACK INTO ANY LOCATION, EQUIVALENT TO THE MOVE DATA COMMAND OF SOME MONITORS.

L WILL LOAD DATA THE SAME AS R, BUT WILL EXECUTE THE PROGRAM AS SSSS IF THE CHECKSUM IS CORRECT.

V READS A TAPE AND COMPARES THE CHECKSUM WITH THAT RECORDED ON THE TAPE; A BYTE BY BYTE COMPARISON IS NOT MADE WITH MEMORY.

A RESULTS IN AN ASCII DUMP OF MEMORY. THIS IS USEFUL FOR EXAMINING FILES OR FOR DISPLAING COMMAND TABLES.

VIDEO DRIVER DEMONSTRATION - MONITOR V 1.2 D

SOME PROGRAMS SUCH AS BASIC DO NOT ECHO CONTROL CHARACTERS; THEY MUST BE OUTPUT USING A CHR$( ) COMMAND. TO DEMONSTRATE THE FEATURES OF THE VIDEO DRIVER, ENTER THE FOLLOWING CODE AT CC00 AND EXECUTE IT FROM THE MONITOR WITH G CC00.

CC00-CD 8B C0 C3 00 CC

THIS ROUTINE CALLS RDCN WHICH INPUTS AN ASCII CODE FROM THE KEYBOARD AND ECHOES IT TO THE VIDEO DRIVER. THE FOLLOWING CHARACTERS ARE USED FOR SPECIAL PURPOSES:

CONTROL D = CLEAR SCREEN
H = HOME CURSOR
L = CURSOR LEFT
N = GRAHICS ON
O = GRAPHICS OFF
R = CURSOR RIGHT
U = CURSOR UP

CARRIAGE RETURN (CONTROL M) AND LINE FEED (CONTROL J) HAVE THE USUAL EFFECTS.

THE VIDEO DRIVER CAN BE CALLED BY ANOTHER PROGRAM AT C700, WITH AN ASCII CODE IN THE ACCUMULATOR (MSB MUST BE 0) AND ALL REGISTERS WILL BE SAVED AND RESTORED ON RETURN. THE POLY VIDEO BOARD MUST BE ADDRESSED AT D000H, AND THE STATUS PORT MODIRCATION MUST BE MADE TO THE BOARD TO PROVIDE A STATUS PORT AT DI WITH KEYSTROKE AND VERTICAL RETRACE STATUS BITS. THE VIDEO INTERFACE MEMORY CAN BE WRITTEN TO DIRECTLY; TRY T D000 D3FF.

VECTOR GRAPHIC INC.
717 LAKEFIELD ROAD, SUITE F • WESTLAKE VILLAGE, CA 91361 • (805) 497-0733

```
C000                    0010 CONC   EQU   0              CONSOLE STAT PORT
C000                    0020 COND   EQU   1              CONSOLE DATA PORT
C000                    0030 CASD   EQU   6FH            CASSETE DATA PORT
C000                    0040 CASC   EQU   6EH            CASS STAT PORT
C000                    0050 SPTR   EQU   0D000H         STACK POINTER
C000                    0051 *
C000                    0052 *** VECTOR ONE MONITOR - VERSION 1.2(A)
C000                    0053 *FOR SIO REV. 1 AND 3P+S W. INV. STATUS
C000                    0054 ********* COMMAND FORMAT *********
C000                    0055 *G LLLL GO TO LOC LLLL AND EXEC
C000                    0056 *D SSSS FFFF DISPLAY MEMORY
C000                    0057 *P LLLL PROGRAM MEMORY
C000                    0058 *T SSSS FFFF TEST MEMORY
C000                    0059 *R SSSS FFFF READ CASSETTE
C000                    0060 *W SSSS FFFF WRITE CASSETTE
C000                    0061 *V SSSS FFFF VERIFY CASSETTE
C000                    0062 *L SSSS FFFF LOAD AND GO
C000                    0063 *A SSSS FFFF ASCII DUMP
C000                    0064 ***********************************
C000                    0070 *
C000 C3 03 C0           0080        JMP   INIT
C003                    0090 INIT   DS    8
C00B 31 00 D0           0100 START  LXI   SP,SPTR
C00E CD 81 C0           0105        CALL  CRLF
C011 3E 2A              0110        MVI   A,'*'          PRINT "*"
C013 CD 75 C0           0120        CALL  PTCN
C016 CD 8B C0           0130        CALL  RDCN           READ KEYBOARD
C019 F5                 0140        PUSH  PSW            SAVE INPUT
C01A CD 73 C0           0150        CALL  SPCE
C01D F1                 0160        POP   PSW            RESTORE ACC
C01E FE 47              0170        CPI   'G'            IF G
C020 CC 4E C0           0180        CZ    EXEC           EXECUTE A PROGRAM
C023 FE 56              0190        CPI   'V'            IF V,
C025 CC CB C0           0200        CZ    CINR           GOTO INPUT ROUTINE
C028 FE 57              0230        CPI   'W'            IF W
C02A CA 99 C0           0240        JZ    COUTR          GO TO CASS OUT
C02D FE 44              0250        CPI   'D'            IF D
C02F CC 8E C1           0260        CZ    DISP           GO TO MEM DISP
C032 FE 50              0270        CPI   'P'            IF P
C034 CC C6 C1           0280        CZ    PGM            GO TO PROG MEM
C037 FE 52              0290        CPI   'R'            IF R
C039 CC CB C0           0300        CZ    CINR           GOTO CASS IN
C03C FE 4C              0310        CPI   'L'            IF L
C03E CC CB C0           0320        CZ    CINR           DO A LOAD AND GO
C041 FE 54              0330        CPI   'T'            IF T
C043 CC 19 C1           0340        CZ    TMEM           TEST MEMORY
C046 FE 41              0342        CPI   'A'            IF A
C048 CC 8E C1           0344        CZ    DISP           DUMP ASCII
C04B C3 0B C0           0350        JMP   START          START OVER
C04E                    0360 *
C04E                    0370 *** EXECUTE THE PROGRAM AT THE ADDRESS ***
C04E                    0380 *
C04E CD 57 C0           0390 EXEC   CALL  AHEX           READ ADD FROM KB
```

```
C051 EB                  0392          XCHG
C052 11 0B C0            0394          LXI   D,START
C055 D5                  0396          PUSH  D
C056 E9                  0400          PCHL  JUMP            TO IT
C057                     0410 *
C057                     0420 *** CONVERT UP TO 4 HEX DIGITS TO BIN
C057                     0430 *
C057 21 00 00            0440 AHEX     LXI   H,0             GET 16 BIT ZERO
C05A 0E 04               0450          MVI   C,4             COUNT OF 4 DIGITS
C05C CD 8B C0            0460 AHE1     CALL  RDCN            READ A BYTE
C05F 29                  0470          DAD   H               SHIFT 4 LEFT
C060 29                  0480          DAD   H
C061 29                  0490          DAD   H
C062 29                  0500          DAD   H
C063 D6 30               0510          SUI   48              ASCII BIAS
C065 FE 0A               0520          CPI   10              DIGIT 0-10
C067 DA 6C C0            0530          JC    ALF
C06A D6 07               0540          SUI   7               ALPHA BIAS
C06C 85                  0550 ALF      ADD   L
C06D 6F                  0560          MOV   L,A
C06E 0D                  0570          DCR   C               4 DIGITS?
C06F C2 5C C0            0580          JNZ   AHE1            KEEP READING
C072 EB                  0585          XCHG
C073 3E 20               0590 SPCE     MVI   A,20H           PRINT SPACE
C075 F5                  0600 PTCN     PUSH  PSW             SAVE REG A
C076 DB 00               0610 PTLOP    IN    CONC            READ PRTR STATUS
C078 E6 80               0620          ANI   80H             IF BIT 7 NOT 0,
C07A C2 76 C0            0630          JNZ   PTLOP           WAIT TILL TIS
C07D F1                  0640          POP   PSW             THEN RECOVER A
C07E D3 01               0650          OUT   COND            AND PRINT IT
C080 C9                  0660          RET   RETURN          FROM PTCN
C081 3E 0D               0670 CRLF     MVI   A,0DH           PRINT CR
C083 CD 75 C0            0680          CALL  PTCN
C086 3E 0A               0690          MVI   A,0AH
C088 C3 75 C0            0700          JMP   PTCN
C08B                     0710 *
C08B                     0720 *** READ FROM CONSOLE TO REG A ***
C08B                     0730 *
C08B DB 00               0740 RDCN     IN    CONC            READ KB STATUS
C08D E6 01               0750          ANI   1               IF BIT 1 NOT 0
C08F C2 8B C0            0760          JNZ   RDCN            REPEAT UNTIL IT IS
C092 DB 01               0770          IN    COND            READ FROM KB
C094 E6 7F               0780          ANI   7FH             STRIP OFF MSB
C096 C3 75 C0            0790          JMP   PTCN            ECHO ONTO PRINTER
C099                     0860 *
C099                     0870 *** CASSETTE INTERFACE OUTPUT ROUTINE ***
C099                     0880 *
C099 CD 57 C0            0890 COUTR    CALL  AHEX            READ BLOCK LENGTH
C09C CD 57 C0            0910          CALL  AHEX            READ STARTING ADD
C09F 06 00               0920          MVI   B,0             START CHECKSUM = 0
C0A1 CD BF C0            0930          CALL  COUT            START BYTE OUT
C0A4 3E E6               0940          MVI   A,0E6H          SEND SYNC BYTE
C0A6 CD BF C0            0950          CALL  COUT            TO CASSETTE
C0A9 7E                  0960 COLOP    MOV   A,M             GET DATA FROM MEM
C0AA CD BF C0            0970          CALL  COUT            SEND TO CASSETTE
C0AD 80                  0980          ADD   B               ADD TO CHECKSUM
C0AE 47                  0990          MOV   B,A
C0AF CD F5 C1            1000          CALL  BMP
C0B2 C2 A9 C0            1040          JNZ   COLOP           REPEAT LOOP
C0B5 78                  1050          MOV   A,B             GET CHECKSUM
C0B6 CD BF C0            1060          CALL  COUT            OUTPUT IT
```

```
C0B9 CD 74 C1          1065          CALL  PT2             PRINT CHECKSUM
C0BC C3 0B C0          1070          JMP   START           GET ANOTH COMMND
C0BF F5                1080 COUT     PUSH  PSW             SAVE A AND FLAGS
C0C0 DB 6E             1090 CLOP     IN    CASC            READ CASS STATUS
C0C2 E6 20             1100          ANI   20H             LOOK AT BIT 5
C0C4 C2 C0 C0          1110          JNZ   CLOP            TRY AGAIN?
C0C7 F1                1120          POP   PSW             RESTORE A
C0C8 D3 6F             1130          OUT   CASD            SEND DATA TO CASS
C0CA C9                1140          RET   RETURN          FROM COUT
C0CB                   1150 *
C0CB                   1160 *** CASSETTE INPUT ROUTINE ***
C0CB                   1170 *
C0CB F5                1180 CINR     PUSH  PSW             SAVE CONTROL CHAP
C0CC 3E 10             1190          MVI   A,10H           USE BIT 4 IN REG A
C0CE D3 6E             1200          OUT   CASC            TO RESET CASS INT
C0D0 CD 57 C0          1210          CALL  AHEX            READ BLOCK LENGTH
C0D3 CD 57 C0          1230          CALL  AHEX            READ STARTING ADD
C0D6 F1                1240          POP   PSW             GET CONTPOL CHAP
C0D7 E5                1250          PUSH  H               SAVE START ADD
C0D8 F5                1260          PUSH  PSW             UNDER CONTROL CHAR
C0D9 06 00             1270          MVI   B,0             SET CHECKSUM = 0
C0DB CD 0F C1          1280 CILOP    CALL  CIN             READ FM CONS
C0DE 4F                1290          MOV   C,A             SAVE IT IN REG C
C0DF F1                1300          POP   PSW             GET CONTPOL CHAR
C0E0 F5                1310          PUSH  PSW             SAVE IT BACK
C0E1 FE 56             1320          CPI   'V'             IS IT A V?
C0E3 79                1330          MOV   A,C             GET BACK DATA BYTE
C0E4 CA E8 C0          1340          JZ    CINO            IF C, DON'T STOPE
C0E7 77                1350          MOV   M,A             IF NOT , STORE
C0E8 80                1360 CINO     ADD   B               ADD TO CHECKSUM
C0E9 47                1370          MOV   B,A
C0EA CD F5 C1          1380          CALL  BMP
C0ED C2 DB C0          1420          JNZ   CILOP           READ MORE
C0F0 CD 0F C1          1430          CALL  CIN             READ LAST BYTE
C0F3 F5                1431          PUSH  PSW
C0F4 CD 74 C1          1432          CALL  PT2             PRINT CHECKSUM
C0F7 CD 73 C0          1434          CALL  SPCE            SPACE OVER
C0FA F1                1435          POP   PSW
C0FB B8                1440          CMP   B               COMP TO CHKSUM
C0FC 3E 45             1450          MVI   A,'E'           PRINT E FOP ERROR
C0FE C2 09 C1          1460          JNZ   CERR            PRINT NOW IF ERROP
C101 F1                1470          POP   PSW             RECOVER CTL CHAR
C102 FE 4C             1480          CPI   'L'             IF NOT L
C104 C2 09 C1          1490          JNZ   CERR            DON'T EXECUTE
C107 E1                1500          POP   H               OTHERWISE, EXECJTE
C108 E9                1510          PCHL  AT              STARTING ADDPESS
C109 CD 75 C0          1520 CERR     CALL  PTCN            PRINT V,E, OR R
C10C C3 0B C0          1530          JMP   START
C10F DB 6E             1540 CIN      IN    CASC            READ STATUS
C111 E6 10             1550          ANI   10H             LOOK AT BIT 4
C113 C2 0F C1          1560          JNZ   CIN             WAIT UNTIL LOW
C116 DB 6F             1570          IN    CASD            READ DATA FM CASS
C118 C9                1580          RET   RETURN          FROM CIN
C119                   1590 *
C119                   1600 *** MEMORY TEST ROUTINE ***
C119                   1610 *
C119 CD 57 C0          1620 TMEM     CALL  AHEX            READ BLK LEN
C11C CD 57 C0          1640          CALL  AHEX            READ ST ADD
C11F 01 5A 5A          1650          LXI   B,5A5AH         INI B,C
C122 CD 4A C1          1660 CYCL     CALL  RNDM
C125 C5                1670          PUSH  B               KEEP ALL REGS
```

```
C126 E5                 1680          PUSH  H
C127 D5                 1690          PUSH  D
C128 CD 4A C1           1700 TLOP     CALL  RNDM
C12B 70                 1710          MOV   M,B           WRITE IN MEM
C12C CD F5 C1           1720          CALL  BMP
C12F C2 28 C1           1760          JNZ   TLOP          REPEAT LOOP
C132 D1                 1770          POP   D
C133 E1                 1780          POP   H             RESTORE ORIG
C134 C1                 1790          POP   B             VALUES OF
C135 E5                 1800          PUSH  H
C136 D5                 1810          PUSH  D
C137 CD 4A C1           1820 RLOP     CALL  RNDM          GEN NEW SEQ
C13A 7E                 1830          MOV   A,M           READ MEM
C13B B8                 1840          CMP   B             COMP MEM
C13C C4 68 C1           1850          CNZ   ERR           CALL ERROR ROUT
C13F CD F5 C1           1860          CALL  BMP
C142 C2 37 C1           1930          JNZ   RLOP
C145 D1                 1940          POP   D
C146 E1                 1950          POP   H
C147 C3 22 C1           1960          JMP   CYCL
C14A                    1970 *** THIS ROUTINE GENERATES RANDOM NOS ***
C14A 78                 1980 RNDM     MOV   A,B           LOOK AT B
C14B E6 B4              1990          ANI   0B4H          MASK BITS
C14D A7                 2000          ANA   A             CLEAR CY
C14E EA 52 C1           2010          JPE   PEVE          JUMP IF EVEN
C151 37                 2020          STC
C152 79                 2030 PEVE     MOV   A,C           LOOK AT C
C153 17                 2040          RAL                 ROTATE CY IN
C154 4F                 2050          MOV   C,A           RESTORE C
C155 78                 2060          MOV   A,B           LOOK AT B
C156 17                 2070          RAL                 ROTATE CY IN
C157 47                 2080          MOV   B,A           RESTORE B
C158 C9                 2090          RET                 RETURN W NEW B,C
C159                    2100 *
C159                    2110 *** ERROR PRINT OUT ROUTINE
C159                    2120 *
C159 CD 81 C0           2130 PTAD     CALL  CRLF          PRINT CR,LF
C15C 7C                 2140          MOV   A,H           PRINT
C15D CD 74 C1           2150          CALL  PT2           ASCII
C160 7D                 2160          MOV   A,L           CODES
C161 CD 74 C1           2170          CALL  PT2           FOR
C164 CD 73 C0           2180          CALL  SPCE          ADDRESS
C167 C9                 2200          RET
C168 F5                 2210 EPR      PUSH  PSW           SAVE ACC
C169 CD 59 C1           2220          CALL  PTAD          PRINT ADD.
C16C 78                 2230          MOV   A,B           DATA
C16D CD 74 C1           2240          CALL  PT2           WRITTEN
C170 CD 73 C0           2250          CALL  SPCE
C173 F1                 2270          POP   PSW           DATA READ
C174 F5                 2280 PT2      PUSH  PSW
C175 CD 7C C1           2290          CALL  BINH
C178 F1                 2300          POP   PSW
C179 C3 80 C1           2310          JMP   BINL
C17C 1F                 2320 BINH     RAR
C17D 1F                 2330          RAR
C17E 1F                 2340          RAR
C17F 1F                 2350          RAR
C180 E6 0F              2360 BINL     ANI   0FH           LOW 4 BITS
C182 C6 30              2370          ADI   48            ASCII BIAS
C184 FE 3A              2380          CPI   58            DIGIT 0-9
```

```
C186 DA 75 C0          2390          JC    PTCN
C189 C6 07             2400          ADI   7              DIGIT A-F
C18B C3 75 C0          2410          JMP   PTCN
C18E                   2420 *
C18E                   2430 *** DISPLAY MEMORY CONTENTS ***
C18E                   2440 *
C18E 47                2450 DISP     MOV   B,A            SAVE CONTROL
C18F CD 57 C0          2455          CALL  AHEX           START
C192 CD 57 C0          2470          CALL  AHEX           FINISH
C195 0E 10             2480 ENT1     MVI   C,16           LOC/LINE
C197 CD 59 C1          2490          CALL  PTAD
C19A 78                2492 LP2      MOV   A,B
C19B FE 41             2500          CPI   'A'            IS IT "A"?
C19D 7E                2505          MOV   A,M
C19E CA B2 C1          2507          JZ    ASCD           DUMP ASCII
C1A1 CD 74 C1          2510          CALL  PT2            PRINT OUT
C1A4 CD 73 C0          2515          CALL  SPCE
C1A7 CD F5 C1          2520 LP3      CALL  BMP
C1AA C8                2525          RZ
C1AB 0D                2530          DCR   C
C1AC CA 95 C1          2540          JZ    ENT1           END OF LINE
C1AF C3 9A C1          2600          JMP   LP2            CONTINUE LOOP
C1B2 E6 60             2601 ASCD     ANI   60H            MASK FOR CONTROL
C1B4 C2 BD C1          2602          JNZ   NCON
C1B7 CD 73 C0          2603          CALL  SPCE
C1BA C3 A7 C1          2604          JMP   LP3
C1BD 7E                2605 NCON     MOV   A,M
C1BE E6 7F             2606          ANI   7FH            MASK FOR ASCII
C1C0 CD 75 C0          2607          CALL  PTCN
C1C3 C3 A7 C1          2608          JMP   LP3
C1C6                   2610 *
C1C6                   2620 *** PROGRAM MEMORY *****
C1C6                   2630 *
C1C6 CD 57 C0          2640 PGM      CALL  AHEX           READ ADD.
C1C9 EB                2645          XCHG
C1CA CD 81 C0          2650          CALL  CRLF
C1CD 7E                2660 PGLP     MOV   A,M            READ MEMORY
C1CE CD 74 C1          2670          CALL  PT2            PRINT 2 DIG.
C1D1 3E 2D             2680          MVI   A,'-'          LOAD DASH
C1D3 CD 75 C0          2690          CALL  PTCN           PRINT DASH
C1D6 CD 8B C0          2700 CRIG     CALL  RDCN
C1D9 FE 2F             2710          CPI   '/'
C1DB C8                2720          RZ                   QUIT ON SLASH
C1DC FE 0D             2730          CPI   0DH
C1DE C2 E7 C1          2740          JNZ   CON1           SKIP IF CR
C1E1 CD 81 C0          2750          CALL  CRLF           PRINT CR,LF
C1E4 C3 D6 C1          2760          JMP   CRIG           BACK FO MO
C1E7 EB                2770 CON1     XCHG                 H,L>D,E
C1E8 21 00 00          2780          LXI   H,0            GET 16 BIT ZERO
C1EB 0E 02             2790          MVI   C,2            COUNT 2 DIG.
C1ED CD 5F C0          2800          CALL  AHE1+3         CONV TO HEX
C1F0 73                2820          MOV   M,E            WRITE IN MEM
C1F1 23                2830          INX   H              INC POINTER
C1F2 C3 CD C1          2840          JMP   PGLP           KEEP GOING
C1F5 7B                3000 BMP      MOV   A,E
C1F6 95                3010          SUB   L
C1F7 C2 FC C1          3020          JNZ   GOON
C1FA 7A                3030          MOV   A,D
C1FB 9C                3040          SBB   H
C1FC 23                3050 GOON     INX   H
C1FD C9                3060          RET
```

SYMBOL TABLE

AHE1	C05C	AHEX	C057	ALF	C06C	ASCD	C1B2	BINH	C17C	BINL	C180
BMP	C1F5	CASC	006E	CASD	006F	CERR	C109	CILOP	C0DB	CIN	C10F
CINO	C0E8	CINR	C0CB	CLOP	C0C0	COLOP	C0A9	CON1	C1E7	CONC	0000
COND	0001	COUT	C0BF	COUTR	C099	CRIG	C1D6	CRLF	C081	CYCL	C122
DISP	C18E	ENT1	C195	ERR	C168	EXEC	C04E	GOON	C1FC	INIT	C003
LP2	C19A	LP3	C1A7	NCON	C1BD	PEVE	C152	PGLP	C1CD	PGM	C1C6
PT2	C174	PTAD	C159	PTCN	C075	PTLOP	C076	RDCN	C08B	RLOP	C137
RNDM	C14A	SPCE	C073	SPTR	D000	START	C00B	TLOP	C128	TMEM	C119

D 3000 31FF

```
3000 C3 03 C0 00 00 00 00 00 00 00 00 31 00 D0 CD 81
3010 C0 3E 2A CD 75 C0 CD 8B C0 F5 CD 73 C0 F1 FE 47
3020 CC 4E C0 FE 56 CC CB C0 FE 57 CA 99 C0 FE 44 CC
3030 8E C1 FE 50 CC C6 C1 FE 52 CC CB C0 FE 4C CC CB
3040 C0 FE 54 CC 19 C1 FE 41 CC 8E C1 C3 0B C0 CD 57
3050 C0 EB 11 0B C0 D5 E9 21 00 00 0E 04 CD 8B C0 29
3060 29 29 29 D6 30 FE 0A DA 6C C0 D6 07 85 6F 0D C2
3070 5C C0 EB 3E 20 F5 DB 00 E6 80 C2 76 C0 F1 D3 01
3080 C9 3E 0D CD 75 C0 3E 0A C3 75 C0 DB 00 E6 01 C2
3090 8B C0 DB 01 E6 7F C3 75 C0 CD 57 C0 CD 57 C0 06
30A0 00 CD BF C0 3E E6 CD BF C0 7E CD BF C0 80 47 CD
30B0 F5 C1 C2 A9 C0 78 CD BF C0 CD 74 C1 C3 0B C0 F5
30C0 DB 6E E6 20 C2 C0 C0 F1 D3 6F C9 F5 3E 10 D3 6E
30D0 CD 57 C0 CD 57 C0 F1 E5 F5 06 00 CD 0F C1 4F F1
30E0 F5 FE 56 79 CA E8 C0 77 80 47 CD F5 C1 C2 DB C0
30F0 CD 0F C1 F5 CD 74 C1 CD 73 C0 F1 B8 3E 45 C2 09
3100 C1 F1 FE 4C C2 09 C1 E1 E9 CD 75 C0 C3 0B C0 DB
3110 6E E6 10 C2 0F C1 DB 6F C9 CD 57 C0 CD 57 C0 01
3120 5A 5A CD 4A C1 C5 E5 D5 CD 4A C1 70 CD F5 C1 C2
3130 28 C1 D1 E1 C1 E5 D5 CD 4A C1 7E B8 C4 68 C1 CD
3140 F5 C1 C2 37 C1 D1 E1 C3 22 C1 78 E6 B4 A7 EA 52
3150 C1 37 79 17 4F 78 17 47 C9 CD 81 C0 7C CD 74 C1
3160 7D CD 74 C1 CD 73 C0 C9 F5 CD 59 C1 78 CD 74 C1
3170 CD 73 C0 F1 F5 CD 7C C1 F1 C3 80 C1 1F 1F 1F 1F
3180 E6 0F C6 30 FE 3A DA 75 C0 C6 07 C3 75 C0 47 CD
3190 57 C0 CD 57 C0 0E 10 CD 59 C1 78 FE 41 7E CA B2
31A0 C1 CD 74 C1 CD 73 C0 CD F5 C1 C8 0D CA 95 C1 C3
31B0 9A C1 E6 60 C2 BD C1 CD 73 C0 C3 A7 C1 7E E6 7F
31C0 CD 75 C0 C3 A7 C1 CD 57 C0 EB CD 81 C0 7E CD 74
31D0 C1 3E 2D CD 75 C0 CD 8B C0 FE 2F C8 FE 0D C2 E7
31E0 C1 CD 81 C0 C3 D6 C1 EB 21 00 00 0E 02 CD 5F C0
31F0 73 23 C3 CD C1 7B 95 C2 FC C1 7A 9C 23 C9 00 00
```

VECTOR 1 MONITOR V 1.2

B,C,D,E Patches

Option B

```
0090 INIT   MVI   A,03H
0091        OUT   10H
0092        MVI   A,11H
0093        OUT   10H

P 0600
0600 PTCN   PUSH  PSW
0610 PTLOP  IN    10H
0620        ANI   02
0630        JZ    PTLOP
0640        POP   PSW
0650        OUT   11H
0660        RET   RETURN

P 0740
0740 RDCN   IN    10H
0750        ANI   1
0760        JZ    RDCN
0770        IN    11H
0780        ANI   7FH
0790        JMP   PTCN
```

Option C

```
0090 INIT   MVI   A,0CEH
0091        OUT   03
0092        MVI   A,37H
0093        OUT   03

P 0600
0600 PTCN   PUSH  PSW
0610 PTLOP  IN    03
0620        ANI   01
0630        JZ    PTLOP
0640        POP   PSW
0650        OUT   02
0660        RET   RETURN

P 0740
0740 RDCN   IN    03
0750        ANI   02
0760        JZ    RDCN
0770        IN    02
0780        ANI   7FH
0790        JMP   PTCN
```

Option D

```
0600 PTCN   JMP   0C700H
0620        ANI   01
0630        JMP   RDCN
0640        POP   PSW
0650        OUT   02
0660        RET   RETURN

P 0740
0740 RDCN   IN    0D0H
0750        ANI   81H
0760        JNZ   RDCN
0770        IN    0D1H
0780        ANI   7FH
0790        JMP   PTCN
```

Option E

```
P 0600
0600 PTCN   PUSH  PSW
0610 PTLOP  IN    CONC
0620        ANI   80H
0630        JZ    PTLOP
0640        POP   PSW
0650        OUT   COND
0660        RET   RETURN

P 0740
0740 RDCN   IN    CONC
0750        ANI   40H
0760        JZ    RDCN
0770        IN    COND
0780        ANI   7FH
0790        JMP   PTCN
```

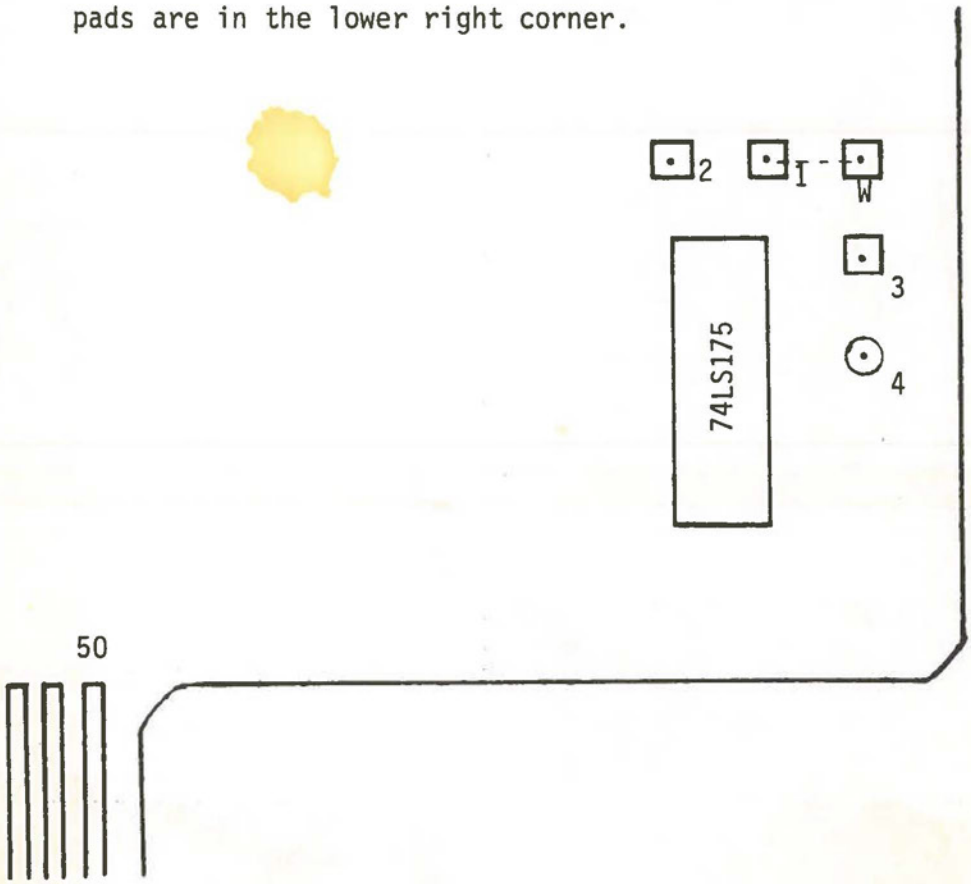
Option B - MITS 2 SIO
Option C - IMSAI SIO 2
Option D - Polymorphic Video Interface
Option E - 3 P + S without inverted status bits

VECTOR GRAPHIC INC.

ERRATA FOR VECTOR GRAPHIC INC. "RESET & GO"
PROM/RAM REV. 3

On page 11, paragraph 3 in the User's Manual and Assembly instructions it states "the number of wait states is prejumpered at 1." THIS IS NOT THE CASE ON THE REV. 3 BOARDS.

To achieve this, a jumper should be installed on the back of the board between PADS #1 and W. Looking at the front (silk-screened side) of the board these pads are in the lower right corner.



790 HAMPSHIRE ROAD, A + B • WESTLAKE VILLAGE, CA. 91361 • TEL: (805) 497-0733