# Hardware Reference Manual

# 64K RAM Board

Central Data Corporation

# 64K RAM Board Manual

Central Data Corporation
P.O. Box 2484, Station A
Champaign, IL 61820
Phone: (217) 359-8010

TABLE OF CONTENTS

## 1.0 Warrantee

The Central Data 64K RAM Board is fully warranteed for a period of one year following the date of shipment. All units returned to Central Oata Corporation postpaid during this period will be repaired and returned without charge.

This warrantee does not apply to boards which have been damaged, abused, or modified. Central Data reserves the right to change the design of the RAM Board without having to change any previously manufactured units.

## 2.0 Oescription

The Central Data 64K RAM Board is a random access read/write memory module designed to expand the memory capacity of S-100 computers. The Board is configured as four sections of 16K x 8 hits. The sections are seperately addressable.

The memory integrated circuits used on the 64K RAM board are 4116-25 16K (or equivalent) dynamic RAMs for 2MHz boards, 4116-20 RAMs for 4MHz boards. These RAM's use considerably less power than static RAM as well as using less PC board space.

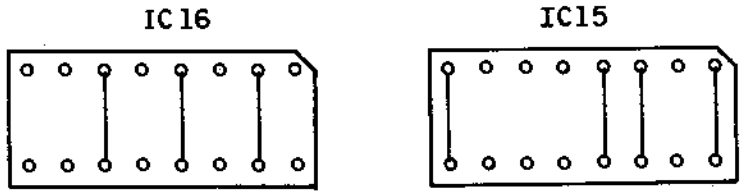
The circuitry has been designed to be independent of the processor type. This manual shows how to set up the board to be accessed by S-100 systems using the 8080, Z80, or 2650 microprocessors. It will work with other types of bus masters as long as they simulate the memory access timing of the processor for which the board is strapped.

## 3.0 Processor Selection

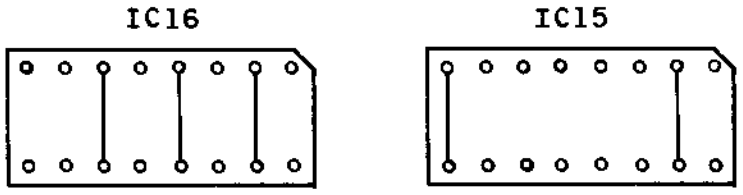
First, there are two straps numbered 14 and 15 soldered on the board which determine which S-100 line is used for reset in your system. Strap 14 selects pin 75 as the reset source, and strap 15 selects pin 99 as the reset source. Normally, the board will work as strapped at the factory (for pin 75), but if you find that your board will not hold data during reset operations, you can change the strap to fix the problem. The strap locations can be found on the parts placement diagram at the end of this manual.

Two 16 pin DIP headers are used to program the RAM Board for use with various processors. Wires must be soldered in place connecting across these headers. The correct connections are shown in figure 1.

8080 PROCESSOR:




Z-80 PROCESSOR:




2650 PROCESSOR:

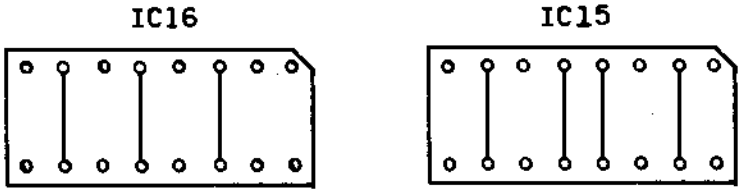



Figure 1

Two alternates for the Z-80 processor would be: on IC15, take out the wire between pins 8 and 9 and add a wire between pins 7 and 10. Then also add a wire between either pins 6 and 11 or pins 5 and 12, depending on which alternate you try. This changes the clock used in the refresh circuit.

## 4.0 Address Selection

When the RAM Board is used in an 8080 or Z80 system it will be addressable on 16K boundries. When strapped for use in a 2650 system, it will be addressable on 8K boundries. Two sets of address selection instructions are given below. Follow the instructions that correspond to the processor type being used.

## 4.1 Address Selection Instructions for 8080 or Z80 Systems

When strapped for 8080 or Z80 bus masters, the 64K RAM Board consists of up to four seperate 16K sections. Each section can be addressed independently and can occupy any 16K block of memory address space.

For each 16K section there will be one blue "minijump" connector bridging the address selection area (16K boards will use one address strap minijump, 32K boards will use two, etc.). Refer to Figure 2 for the location of this area on the board. The exact placement of the minjumps will vary depending on whether you have a board with 16K or 32K of memory or one with 48K or 64K of memory. The detail on how to address the board is presented below in two sections, depending on the size of board that you have.

### 4.1.1 Addressing 16K/32K Boards with 8080 or Z80 Systems

For a 16K or 32K board, you will need to place one or two minijumps (depending on the board size) in the address selection strap area. The minijump for the top row of chips should be placed in the top row of the address selection strap area (see Figure 2). The minijump should be placed over the set of pins marked 0, 1, 2, or 3 to address the top row of RAMs to start at 0, H4000, H8000, or HC000. For a 32K board the second row of RAMs can be addressed in the same way, but with the minijump placed in the bottom row of the address selection strap area. Figure 2 shows the addressing for a 32K board whose top row of RAM's is addressed to start at H8000, and its second row of RAMs is addressed to start at HC000. The deselection straps shown in the figure have no effect on the basic addressing of the board and are explained in section 5.

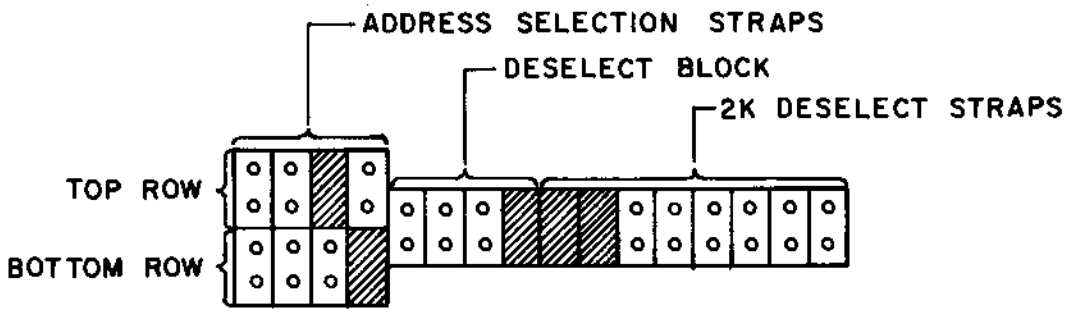



Figure 2

### 4.1.2 Addressing for 48K/64K Boards on 8080 or Z80 Systems

For a 48K or 64K board, you will have to place three or four minjumps (depending on the board size) in the address selection strap area. One minijump will be used for each 16K block that you address. To address the first 32K of the board (the top two rows of RAMs), you will place two minijumps on the top row of the address selection strap area (see Figure 3). Each minijump should be placed to select

where a corresponding 16K block of memory should be addressed. A minijump is placed in column 0, 1, 2, or 3 to start addressing for the 16K section at address 0, H4000, H8000, or HC000. There is one restriction: the two minijumps must be adjacent. This means that the allowable combinations for addressing the first 32K of your board are:

| | |
|---|---|
| 0,1 | (address 0 to H7FFF) |
| 1,2 | (address H4000 to HBFFF) |
| 2,3 | (address H8000 to HFFFF) |
| 3,0 | (address HC000 to H3FFF) |

At this point, the remaining memory on the board can be addressed. To address the last 16K of a 48K board, a single minijump is placed in the correct column of the bottom row of the address selection strap area. For a 64K board, two minijumps are placed on the bottom row, addressing the bottom 32K of memory on the board. Note that these two minijumps must be adjacent as in selecting the top 32K of memory. Figure 3 shows a 48K board addressed from 0 to HBFFF.

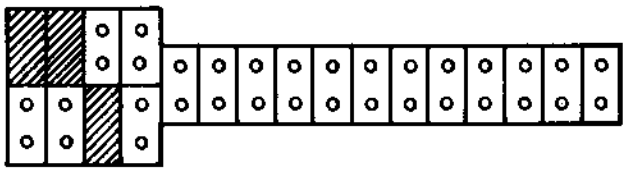


Figure 3

With regard to 48K boards only, the last 16K of memory corresponds to the eight memory chips located either in the third or fourth row of the socketed memory area. If you are addressing this 16K segment to begin on an even 32K boundary (to start at address 0 or H8000) then the row of memory chips must be in the fourth (bottom) row of the memory area on the board. If you are addressing this 16K segment to begin at either H4000 or HC000 then the memory devices must be placed in the third row of sockets in the memory area (adjacent to the top 32K of memory). When the boards are shipped from Central Data they contain the memory on the bottom row of the board.

## 4.2 Addressing Instructions for 2650 Processors

To address a 16K board to be used in a 2650 system, put a minijump in positions 1 and 2 of the top row of the address selection strap area (see Figure 4). This addresses the board to have a continuous 16K of memory starting at address H2000.

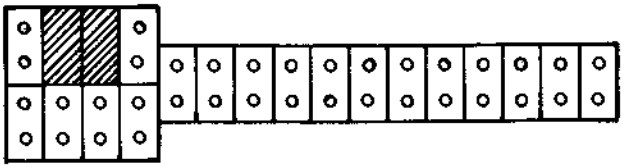


Figure 4

To address a 32K board for a 2650 system, you will have to deselect the portions of page zero which are used by the CPU board. The straps should be set up as shown in Figure 5 for this configuration.

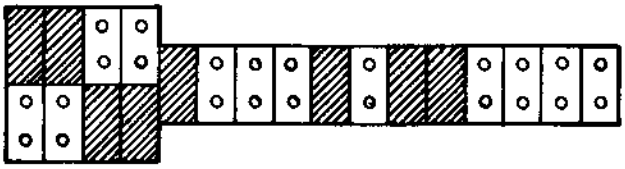


Figure 5

## 5.0 Address Deselect Straps

In order to have the RAM Board fit around ROM monitors or other dedicated areas of memory, from one to eight 2K segments of any 16K address space can be deselected. The instructions for setting these straps are given for all non-2650 systems. Any 2650 users which need this feature should consult the factory.

Use a minijump to select the 16K area in which the gap will occur. Place the minjump in the column of the "Deselect Block" area that corresponds to this block of memory. Now place a minijump in each column of the "Deselect Strap" area that you wish to have deselected.

Example:

You have a SDL-20 computer and wish to use the ALS-8 program which requrires 12K of memory from HD000 to HFFFF. A full 16K of RAM (HC000 to HFFFF) won't work because the monitor (SOLOS) occupies 2K of memory space between HC000 and HC7FF. Also, the SOL-20 has 2K of RAM between HC800 and HCFFF.

A Central Data RAM Board can be used for this purpose by deselecting the occupied areas. Put a minijump in the address selection area to address the 16K of the board from HC000 to HFFFF (see section 4.1). Since the area to be deselected is in the top 16K put a minijump in column 3 of

the "Deselect Block." Now put a minjump in both column 0 and column 1 of the "Deselect Strap" section of the board to deselect the lowest 4K of this 16K. The board is now strapped to occupy the 12K from HD000 to HFFFF. Figure 2 shows a RAM Board addressed this way.

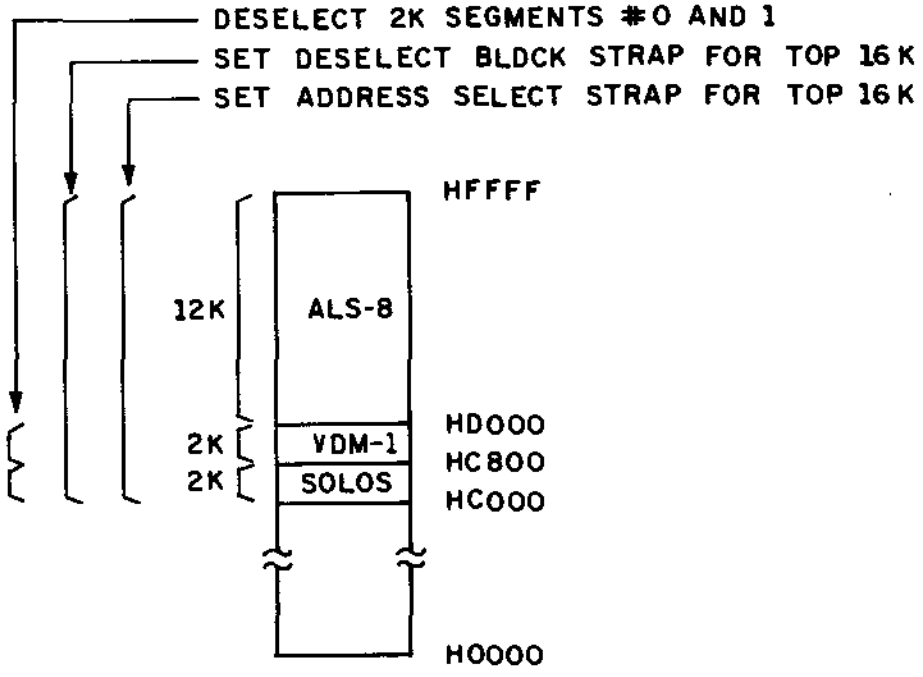



Figure 6

## 6.0 Installation

Before plugging the RAM Board into an S-100 card slot, make certain that the power has been off long enough to allow the power supply capacitors to discharge. Whenever any S-100 board is plugged in or removed, care should be taken to make sure that the edge connector lines up with the printed circuit board fingers.

Keeping these precautions in mind, put the board into the S-100 mainframe. Make sure the board faces the right direction. The power may now be applied. The RAM Board is ready for use.

## 7.0 Increasing your Memory Size

To increase the memory capacity of your board, you can purchase a set of memory ICs and capacitors from Central Data. The memory circuits should be carefully plugged into the next available row of sockets from the top of the board. The only exception to this is when you are upgrading to a 48K board where you may need to skip the third row in order to address the board correctly (see section 4.1.2 for

information concerning this).

The capacitors should be soldered in according to the parts placement diagram making sure of the polarity on the tantalum capacitors. The capacitors in the memory area of the parts placement diagram are numberd to indicate which ones are to be put in for the different sizes of boards. Capacitors with the number '1' should all be in for 16K boards, while 32K board should have capacitors for both numbers '1' and '2', etc.

Finally, if you upgraded a 16K or 32K board to a 48K or 64K board, you will have to change two jumper straps on the board. These are the straps numbered 1, 2, 3, and 4 on the parts placement diagram, and they should be wired as follows for the different sizes of boards:

**16K,32K** **48K,64K**

## 8.0 Specifications

| | |
|---|---|
| Storage Capacity | 16K, 32K, 48K, or 64K |
| Addressing | 16K boundaries |
| Buffering | On all I/O lines |
| Access Time - 2MHz Bd. | 450ns(max) |
| 4MHz Bd. | 250ns(max) |
| Cycle Time | 480ns(max) |
| Refresh Period | 15us(max) |
| Wait States Generated | None |
| DMA Rate | 1MHz Max |

Power Consumption (16K, typ.):

| | |
|---|---|
| +16 | 150ma |
| +8 | 300ma |
| -16 | 20ma |

Power Consumption (each add'l 16K, typ.):

| | |
|---|---|
| +16V. | 20ma |
| +8V. | 0ma |
| -16V. | 3ma |

## 9.0 8080 Memory Test Program

The memory test program listed below can be used to test a 16K memory segment addressed for Block 1 (H4000-H7FFF). When an error is found the program stores information about the error and then halts. The error information is stored as follows:

| Address | Information |
|---|---|
| H0003 | High byte of address |
| H0002 | Low byte of address |
| H0001 | Data written to byte |
| H0000 | Data read from byte |

A 32K memory segment can be tested by addressing the board for blocks 1 and 2 and changing the values at locations H16 and H2C to HC0.

This memory test program occupies memory starting at H0000 so a working RAM board must be at this part of memory.

```
                    BEG     EQU     4000    HEX STARTING
ADDRESS
                    END     EQU     80      TOP BYTE OF
END ADDR
                    *
                            ORG     4
0004 31 04 00       START   LXI     SP,4
0007 AF                     XRA     A
0008 47                     MOV     B,A     CLEAR A,B
0009 21 00 40       OVER    LXI     H,BEG   STARTING ADDR
000C 48                     MOV     C,B
000D 71             WRITE   MOV     M,C     STORE DATA
000E 23                     INX     H       BUMP ADDRESS
000F 0C                     INR     C       BUMP DATA
0010 C2 14 00               JNZ     SKIP
0013 0C                     INR     C       OFFSET PATTERN

0014 7C             SKIP    MOV     A,H
0015 FE 80                  CPI     END
0017 C2 0D 00               JNZ     WRITE
                    *
                    *       READ BACK LOOP
                    *
001A 78                     MOV     A,B
0018 21 00 40               LXI     H,BEG
001E 5E             READ    MOV     E,M
001F BB                     CMP     E
0020 C2 35 00               JNZ     ERROR
0023 23                     INX     H
0024 3C                     INR     A
0025 C2 29 00               JNZ     NDINC
0028 3C                     INR     A
0029 4F             NDINC   MOV     C,A
002A 7C                     MOV     A,H
002B FE 80                  CPI     END
002D 79                     MOV     A,C
002E C2 1E 00               JNZ     READ
0031 04                     INR     B
0032 C3 09 00               JMP     OVER
                    *
                    *       ERROR ROUTINE
                    *
0035 E5             ERROR   PUSH    H
0036 57                     MOV     D,A
0037 D5                     PUSH    D
0038 76                     HLT
```

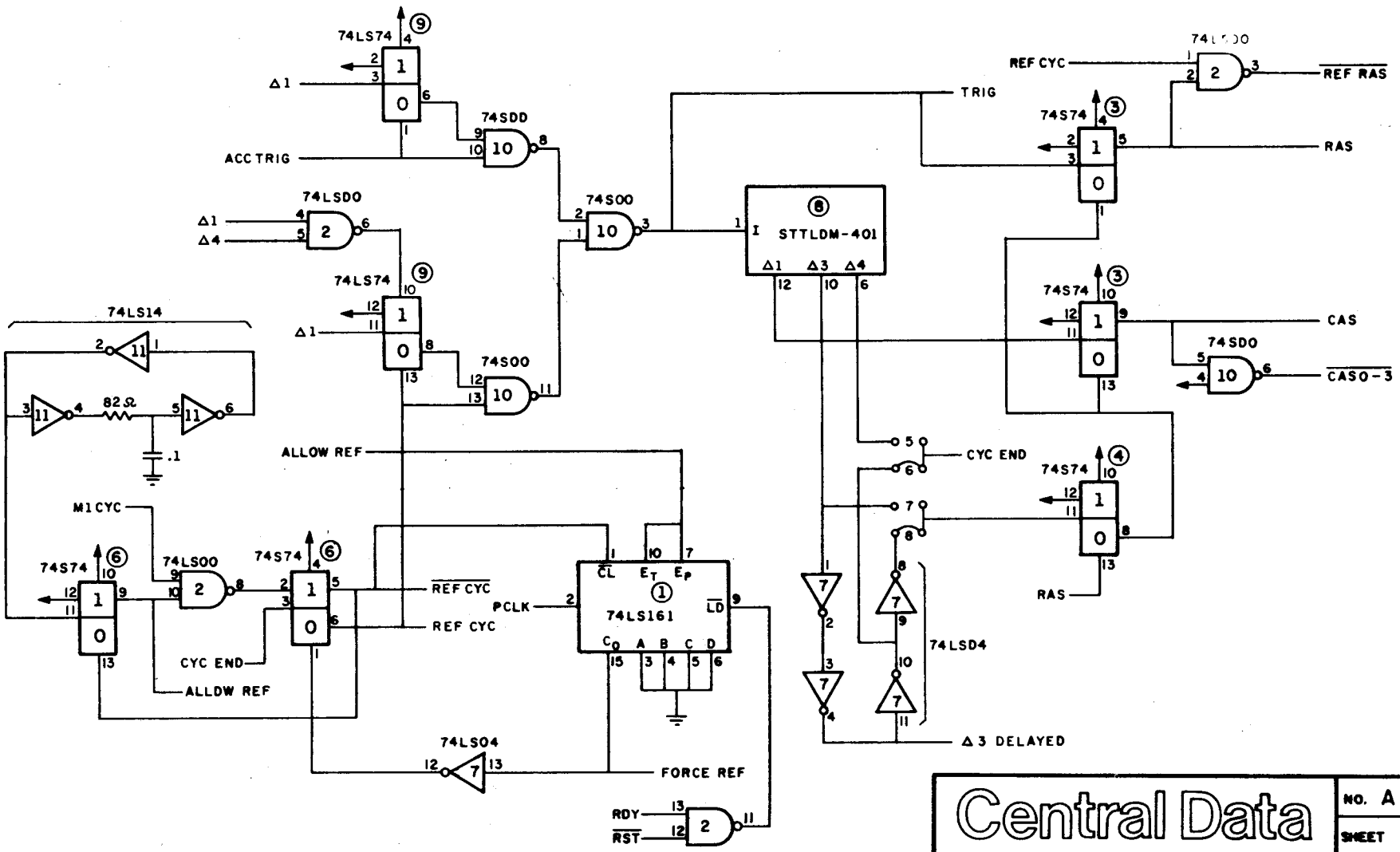
## 10.0 2650 Memory Test Program

A memory test program for the 2650 is listed here. The program loads into display memory at H1510 and occupies about HA0 bytes. To set the address range to test, simply set the two bytes labeled STARTT and ENDT to the high bytes of the starting and ending addresses. For example, if you wanted to test memory between H2000 and H5FFF, you would change STARTT to H20 and ENDT to H60. Then just execute at 1510, and the test will begin. Since the program occupies display RAM, the screen will be broken up when it is being executed. The program terminates when a key is pressed. Errors are displayed on the screen in the following format:

```
   +-------------------------Address of the error
   !   +---------------------Data that was written there
   !   !  +------------------Data that was read back
   !   !  !
 XXXX XX XX
```

```
0001  0000                        PRNT
0002  0000             *
0003  0000             *
0004  0000             *
0005  0000             *          MEMORY TEST
0006  0000             *
0007  0000             *
0008  0000             *
0009  0000             EQ         EQU        0          POSSIBLE CONDITIONS
0010  0000             GT         EQU        1
0011  0000             LT         EQU        2
0012  0000             UN         EQU        3
0013  0000             R0         EQU        0          REGISTER DEFINITIONS
0014  0000             R1         EQU        1
0015  0000             R2         EQU        2
0016  0000             R3         EQU        3
0017  0000             HXOT       EQU        006A
0018  0000             LFCR       EQU        0024
0019  0000             WCHR       EQU        0396
0020  0000             RETU       EQU        0083
0021  0000             *
0022  0000             *
0023  0000             *
0024  1510                        ORG        1510
0025  1510 1B 02                 BCTR,UN    START      BRANCH OVER POINTERS
0026  1512 00          STARTT     RES        1          HIGH BYTE OF START ADDRESS
0027  1513 00          ENDT       RES        1          HIGH BYTE OF END ADDRESS+1
0028  1514 04 00       START      LODI,R0    0
0029  1516 93                     LPSL                  SETUP PSL
0030  1517 3F 00 24               BSTA UN    LFCR       DO LINEFEED AT BEGINNING
0031  151A 20          STRT       EORZ,R0               GET READY TO CLEAR LOW BYTE OF
0032  151B 09 75                  LODR,R1    STARTT     GET HIGH BYTE OF START ADDRESS
0033  151D C9 29                  STRR,R1    RAMPTR     STORE INTO POINTER
0034  151F C8 28                  STRR,R0    RAMPTR+1   ZERO LOW BYTE OF POINTER
0035  1521 05 FF                  LODI,R1    FF         SETUP STARTING DATA
0036  1523 85 01       WRT        ADDI,R1    1          ADD 1 TO STARTING DATA
0037  1525 01                     LODZ,R1               PUT IT IN R0 FOR USE
0038  1526 06 00       WSTL       LODI,R2    0          SETUP INDEX REGISTER
0039  1528 CF F5 48    WRTLP      STRA,R2    *RAMPTR,I  STORE NEXT BYTE
0040  152B 84 01                  ADDI,R0    1          INCREMENT DATA
0041  152D 98 02                  BCFR,EQ    WTSKP      IF ZERO  DONT BRANCH
0042  152F 84 01                  ADDI,R0    1          DONT LET ZERO BE STORED
0043  1531 DA 75       WTSKP      BIRR,R2    WRTLP      DO 256 TIMES
0044  1533 72                     REDD,R2               READ THE KEYBOARD
0045  1534 9E 00 83               BCFA,LT    RETU       RETURN IF KEY IS PRESSED
0046  1537 0A 0F                  LODR,R2    RAMPTR     INCREMENT POINTER
0047  1539 86 01                  ADDI,R2    1
0048  153B CA 0B                  STRR,R2    RAMPTR
0049  153D EE 15 13               COMA,R2    ENDT       SEE IF END OF TEST AREA
0050  1540 98 64                  BCFR,EQ    WSTL       IF NOT, BRANCH
0051  1542 0A 4F                  LODR,R2    STARTT     SETUP FOR START OF TEST AREA
0052  1544 CA 02                  STRR,R2    RAMPTR
0053  1546 1B 02                 BCTR,UN    RD         START READ LOOP
```

```
0054  1548                  *
0055  1548                  *
0056  1548                  *
0057  1548 00 00    RAMPTR  RES       2
0058  154A                  *
0059  154A                  *
0060  154A                  *
0061  154A 01        RD      LODZ,R1                 LOAD THE STARTING DATA VALUE
0062  154B C3                STRZ,R3                 SAVE IT IN R3
0063  154C 06 00     RSTL    LODI,R2   0             SETUP INDEX REGISTER
0064  154E 0E F5 48  RDLP    LODA,R2   *RAMPTR,I     GET NEXT DATA BYTE
0065  1551 E3                COMZ,R3                 COMPARE TO WHAT IT SHOULD BE
0066  1552 B8 23             BSFR,EQ   ERROR         IF NOT THE SAME, GOTO ERROR
0067  1554 87 01             ADDI,R3   1             INCREMENT DATA
0068  1556 98 02             BCFR,EQ   RDSKP         DONT ALLOW ZERO AGAIN
0069  1558 87 01             ADDI,R3   1
0070  155A DA 72     RDSKP   BIRR,R2   RDLP          DO THIS LOOP 256 TIMES
0071  155C 72                REDD,R2                 READ KEYBOARD
0072  155D 9E 00 83          BCFA,LT   RETU          RETURN IF KEY IS PRESSED
0073  1560 0A 66             LODR,R2   RAMPTR        INCREMENT RAM POINTER
0074  1562 86 01             ADDI,R2   1
0075  1564 CA 62             STRR,R2   RAMPTR
0076  1566 EE 15 13          COMA,R2   ENDT          SEE IF END OF TEST AREA
0077  1569 98 E1             BCFR,EQ   RSTL          IF NOT, BRANCH
0078  156B 0E 15 12          LODA,R2   STARTT        REDO THE WHOLE THING AGAIN
0079  156E CA 58             STRR,R2   RAMPTR
0080  1570 1F 15 23          BCTA,UN   WRT
0081  1573                  *
0082  1573                  *
0083  1573                  *
0084  1573 00 00 00  TMP0    RES       4
0085  1577                  *
0086  1577                  *
0087  1577                  *
0088  1577 C8 7A     ERROR   STRR,R0   TMP0          SAVE THE REGISTERS
0089  1579 C9 79             STRR,R1   TMP0+1
0090  157B CA 78             STRR,R2   TMP0+2
0091  157D CB 77             STRR,R3   TMP0+3
0092  157F 0A 47             LODR,R2   RAMPTR        LOAD THE HIGH BYTE
0093  1581 3F 00 6A          BSTA,UN   HXOT          WRITE IT
0094  1584 0A 6F             LODR,R2   TMP0+2        WRITE THE LOW BYTE
0095  1586 3F 00 6A          BSTA,UN   HXOT
0096  1589 07 20             LODI,R3   20            WRITE A SPACE
0097  158B 3F 03 96          BSTA,UN   WCHR
0098  158E 0A 66             LODR,R2   TMP0+3        WRITE THE DATA WRITTEN
0099  1590 3F 00 6A          BSTA,UN   HXOT
0100  1593 07 20             LODI,R3   20            WRITE A SPACE
0101  1595 3F 03 96          BSTA,UN   WCHR
0102  1598 0A 59             LODR,R2   TMP0          WRITE THE DATA READ
0103  159A 3F 00 6A          BSTA,UN   HXOT
0104  159D 3F 00 24          BSTA,UN   LFCR
0105  15A0 08 51             LODR,R0   TMP0          RESTORE THE REGISTERS
0106  15A2 09 50             LODR,R1   TMP0+1
0107  15A4 0A 4F             LODR,R2   TMP0+2
0108  15A6 0B 4E             LODR,R3   TMP0+3
0109  15A8 17                RETC,UN                 RETURN
```



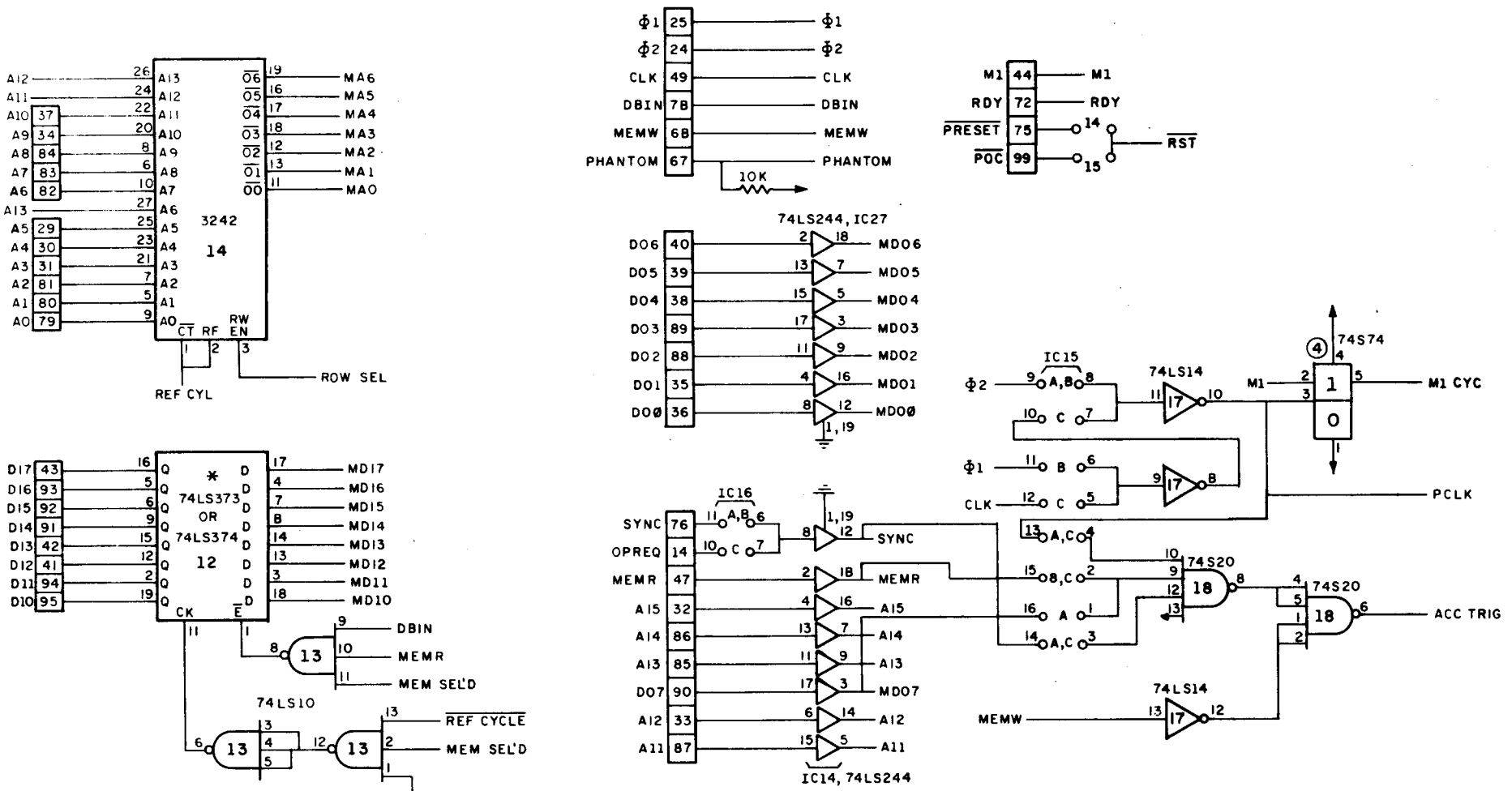

# Central Data

NO. A1065

SHEET 1 OF 5

64 K RAM BOARD

TIMING / REFRESH




NOTES: 1: 'A' STRAPS FOR 8080
'B' STRAPS FOR Z-80
'C' STRAPS FOR SPECIAL USE

2: ALL UNLABELED ARROWS ARE +5V

3: □ = S100 PIN

*: 4 MHz BOARDS REQUIRE 74S' PARTS

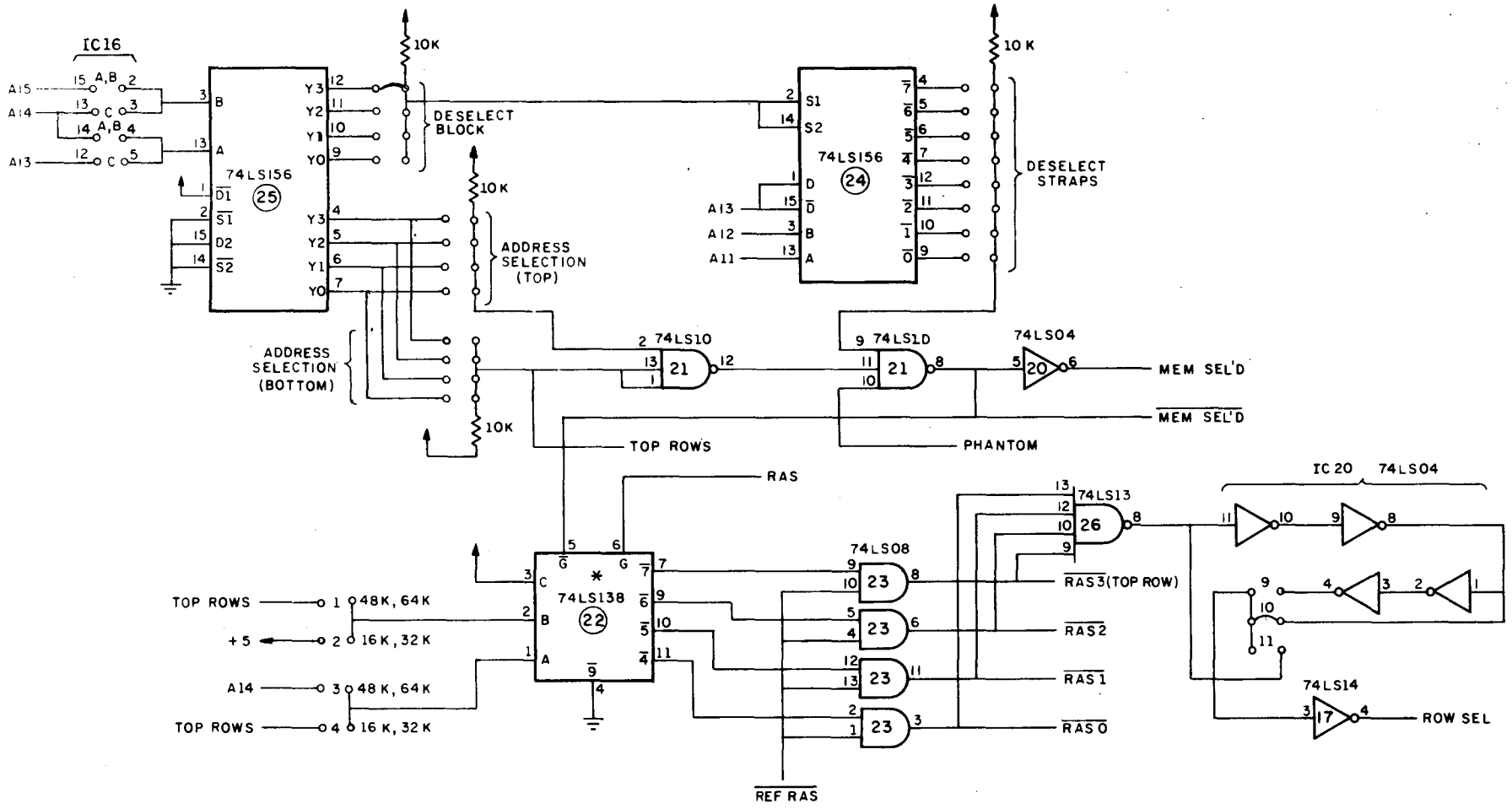
Central Data

NO. A1066

SHEET 2 OF 5

64 K RAM BOARD

BUFFERS




NOTE: 'A' STRAPS FOR 8080
'B' STRAPS FOR Z-80
'C' STRAPS FOR SPECIAL USE
*: 4 MHz BOARDS REQUIRE 74S' PARTS

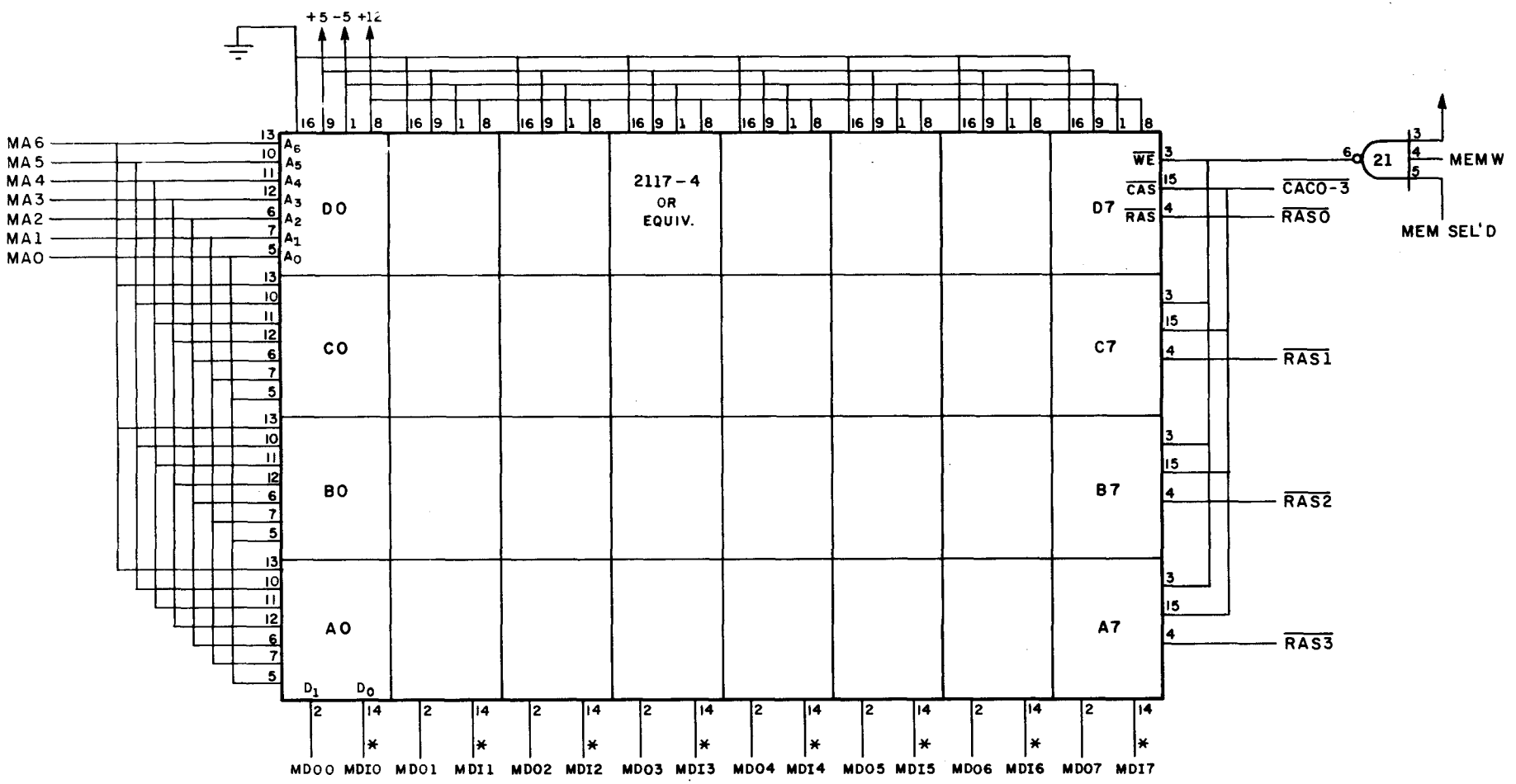
Central Data

NO. A1067

SHEET 3 OF 5

64K RAM BOARD

RAS GATING




* LINES PULLED-UP WITH 10K RESISTOR PAK, RA01

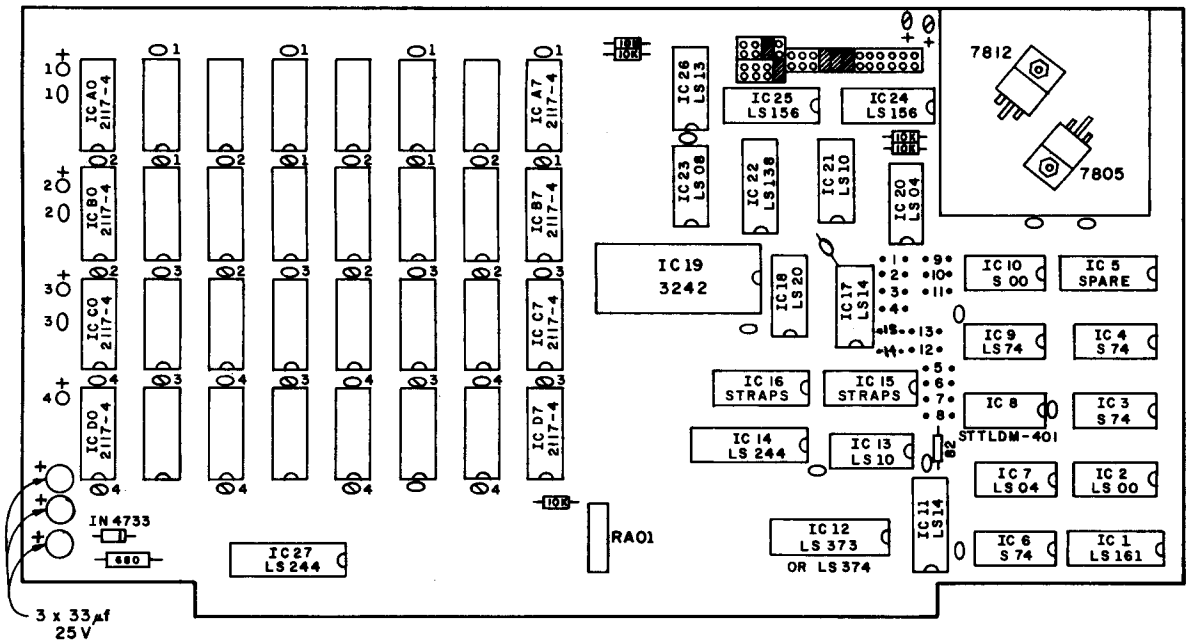
Central Data

NO. A1068

SHEET 4 OF 5

64 K RAM BOARD

MEMORY ARRAY




NOTES: ○ = .1 µf / 25 V
⊘ = .33 µf / 25 V
◯ = 4.7µf / 16 V

# Central Data

NO. A1069

SHEET 5 OF 5

64 K RAM BOARD

PARTS LOCATION