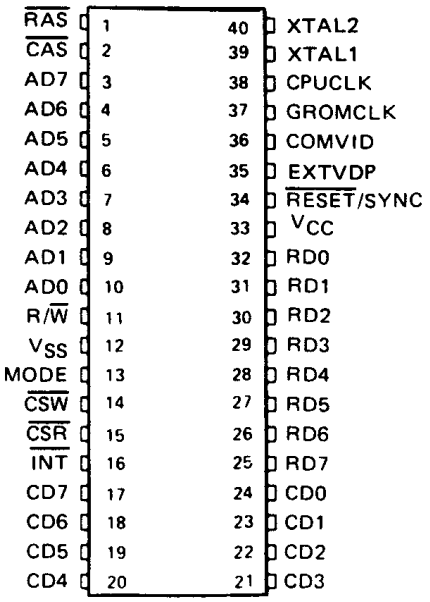
3.7 VDP TERMINAL ASSIGNMENTS

3.7.1 TMS9918A Terminal Assignments

SIGNATURE	TERMINAL	I/O	DESCRIPTION
XTAL1, XTAL2	40,39	I	10.7 + MHz crystal inputs*
CPUCLK	38	O	VDP color burst frequency clock. Typically not used on the TMS9918A, this is the color burst frequency clock.
GROMCLK	37	O	VDP output clock = XTAL/24. Typically not used.
COMVID	36	O	Composite video output for the TMS9918A.
EXTVDP	35	I/O	On the TMS9918A, this is the external VDP input.
$\overline{\text{RESET}}$/ SYNC	34	I	The $\overline{\text{RESET}}$ pin is a trilevel input pin. When it is below 0.8 volts, RESET initializes the VDP. When it is above 9 volts, RESET is the synchronizing input for external video.
V_{CC}	33	I	+5 volt supply
RD0 MSB	32	I	VRAM read data bus
RD1	31	I	
RD2	30	I	
RD3	29	I	
RD4	28	I	
RD5	27	I	
RD6	26	I	
RD7	25	I	
CD0 MSB	24	I/O	CPU data bus; (CD0) is the most significant bit
CD1	23	I/O	
CD2	22	I/O	
CD3	21	I/O	
CD4	20	I/O	
CD5	19	I/O	
CD6	18	I/O	
CD7 LSB	17	I/O	
$\overline{\text{INT}}$	16	O	CPU interrupt output.
$\overline{\text{CSR}}$	15	I	CPU-VDP read strobe
$\overline{\text{CSW}}$	14	I	CPU-VDP write strobe
MODE	13	I	CPU interface mode select; usually a processor address line



* When driven externally, both inputs must be driven.

** The least significant address bit (AD7) is wired to A0 of the dynamic RAMs. Likewise, AD6 is wired to A1 of the RAMs. Care must be exercised in assuring proper orientation of the TMS 9918A address outputs to the dynamic RAM address inputs.

TMS9918A Terminal Assignments (continued)

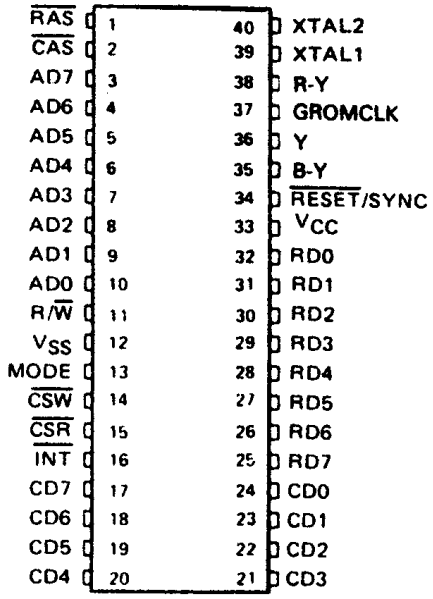
SIGNATURE	TERMINAL	I/O	DESCRIPTION
V_{SS}	12	I	Ground References
R/$\overline{W}$	11	O	VRAM write strobe
AD0 MSB	10	O	VRAM address/data bus (multiplexed high and low order VRAM address and output data bytes)
AD1	9	O	AD0 is the most significant bit and is used only for data and not for addressing.**
AD2	8	O	
AD3	7	O	
AD4	6	O	
AD5	5	O	
AD6	4	O	
AD7	3	O	
$\overline{CAS}$	2	O	VRAM column address strobe
$\overline{RAS}$	1	O	VRAM row address strobe

* When driven externally, both inputs must be driven.

** The least significant address bit (AD7) is wired to A0 of the dynamic RAMs. Likewise, AD6 is wired to A1 of the RAMs. Care must be exercised in assuring proper orientation of the TMS 9918A address outputs to the dynamic RAM address inputs.

3.7.2 TMS9928A/9929A Terminal Assignments

SIGNATURE	TERMINAL	I/O	DESCRIPTION
XTAL1, XTAL2	40,39	I	10.7 + MHz crystal inputs*
R-Y	38	O	VDP color burst frequency clock. On the TMS9928A/9929A, this is the R-Y color difference output.
GROMCLK	37	O	VDP output clock = XTAL/24. Typically not used.
Y	36	O	Composite video output. On the TMS9928A/9929A, this is the Y (black/white luminance and composite sync) output.
B-Y	35	I/O	External VDP input. On the TMS9928A/9929A, this is the B-Y color difference output.
$\overline{\text{RESET}}$/ SYNC	34	I	The $\overline{\text{RESET}}$ pin is a trilevel input pin. When it is below 0.8 volts, RESET initializes the VDP. When it is above 9 volts, RESET is the synchronizing input for external video.
V_{CC}	33	I	+5 volt supply
RD0 MSB	32	I	VRAM read data bus
RD1	31	I	
RD2	30	I	
RD3	29	I	
RD4	28	I	
RD5	27	I	
RD6	26	I	
RD7	25	I	
CD0 MSB	24	I/O	CPU data bus; (CD0) is the most significant bit
CD1	23	I/O	
CD2	22	I/O	
CD3	21	I/O	
CD4	20	I/O	
CD5	19	I/O	
CD6	18	I/O	
CD7 LSB	17	I/O	
$\overline{\text{INT}}$	16	O	CPU interrupt output.
$\overline{\text{CSR}}$	15	I	CPU-VDP read strobe
$\overline{\text{CSW}}$	14	I	CPU-VDP write strobe
MODE	13	I	CPU interface mode select; usually a processor address line



* When driven externally, both inputs must be driven.

** The least significant address bit (AD7) is wired to A0 of the dynamic RAMs. Likewise, AD6 is wired to A1 of the RAMs.

SIGNATURE	TERMINAL	I/O	DESCRIPTION
V_{SS}	12	I	Ground References
R/$\overline{W}$	11	O	VRAM write strobe
AD0 MSB	10	O	VRAM address/data bus (multiplexed high and low order VRAM address and output data bytes)
AD1	9	O	AD0 is the most significant bit and is used only for data and not for addressing.**
AD2	8	O	
AD3	7	O	
AD4	6	O	
AD5	5	O	
AD6	4	O	
AD7	3	O	
$\overline{CAS}$	2	O	VRAM row address strobe
$\overline{RAS}$	1	O	VRam row address strobe

* When driven externally, both inputs must be driven.
** The least significant address bit (AD7) is wired to A0 of the dynamic RAMs. Likewise, AD6 is wired to A1 of the RAMs.

3.7.3 TMS9918A/9928A/9929A Crystals

Crystals for the TMS9918A/9928A/9929A can be purchased from the following:

NDK
10080 North Wolfe Rd
Suite 220
Cuppertino, CA 95014
Telephone:
(408)255-0831
Telex: 352057

CTS Knights, Inc.
400 Reiman Ave
Sandwich, Ill 60548
Telephone:
(815)786-8411

4. DEVICE APPLICATIONS

This section describes the hardware and software interface between a TMS9918A/9928A/9929A VDP and a TMS9900 microprocessor. Some considerations in the use of the VDP for text and graphics applications are also described.

4.1 VDP TO TMS9900 INTERFACE

The circuit shown in Figure 4-1 illustrates a very simple interface between a TMS9900 microprocessor and a TMS9918A/9928A/9929A. In this circuit, the VDP 8-bit data bus is connected to the 8 MSBs of the TMS9900 16-bit data bus. For mode selection, A14 of the TMS9900 is connected to the mode input pin. Read and write signals to the VDP are as follows:

CSR =A0*A13*DBIN
CSW =A0*WE

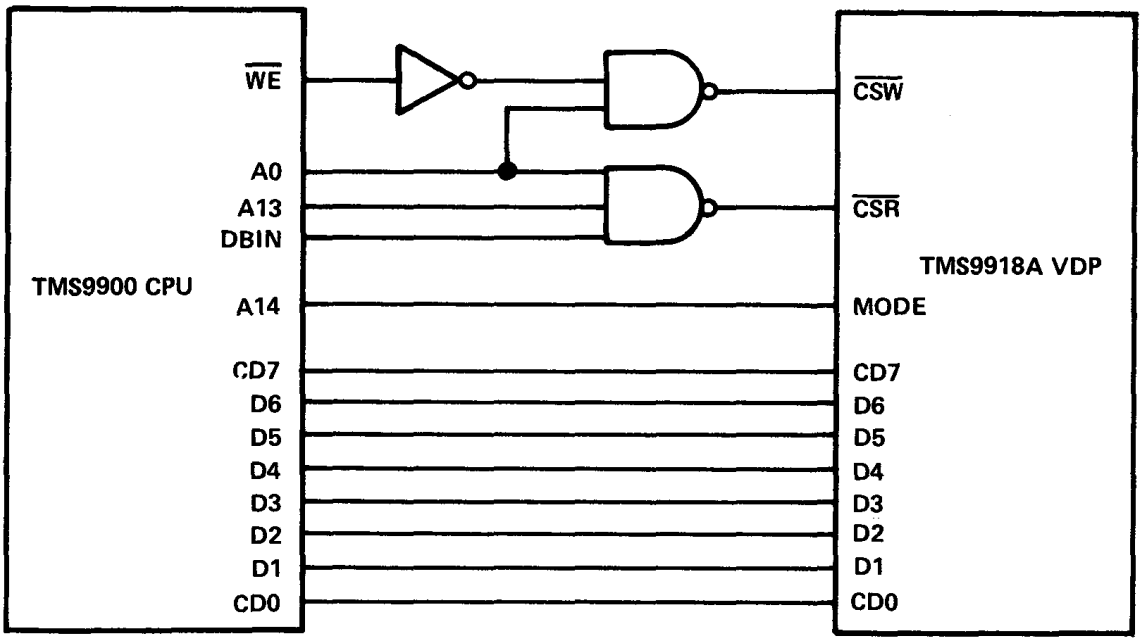


FIGURE 4-1 – MINIMUM SYSTEM INTERFACE TO TMS9900

DBIN and $\overline{WE}$ are signals from the TMS9900 which indicate direction flow on the data bus. DBIN is high when the CPU is attempting to do a read data operation, while $\overline{WE}$ is low when the CPU is outputting data onto the data bus.

A0 is used as a VDP select signal. Thus, the VDP is activated whenever the CPU is reading or writing data in the upper half of its address space (>8000 and above). All addresses above >8000 then become VDP port addresses. However, in a more sophisticated design, more decoding of the address lines would be done to select only those unique addresses required by the VDP. The purpose of A13 and decoding logic is to generate unique addresses for read and write operations and to block out the read data operation that occurs on the TMS9900 before a write data operation. Without this blockout logic, a pulse on the CSR input would occur before any desired pulsing of the CSW input, thus causing unwanted operation of the VDP. Referring to Table 4-1 and Figure 4-1, the following port addresses can be defined.

TABLE 4-1 – VDP PORT ADDRESSES FOR FIGURE 4-1

OPERATION	$\overline{CSW}$	$\overline{CSR}$	MODE	PORT
Write data to VRAM	0	1	0	>8000
Write address to VRAM or Write to VDP register	0	1	1	>8002
Read data from VRAM	1	0	0	>8004
Read VDP status	1	0	1	>8006

4.2 TMS9918A/9928A/9929A INTERFACE

Figures 4-2 and 4-3 show the hardware components necessary to make the VDP operate with a typical TM990 16-bit bus application. The CPU can be connected as shown to any general-purpose 8-bit data bus and control signals that work with most microprocessors. The VDP interface timing is similar to that of static memories and occupies eight unique memory address locations within the CPU memory address space.

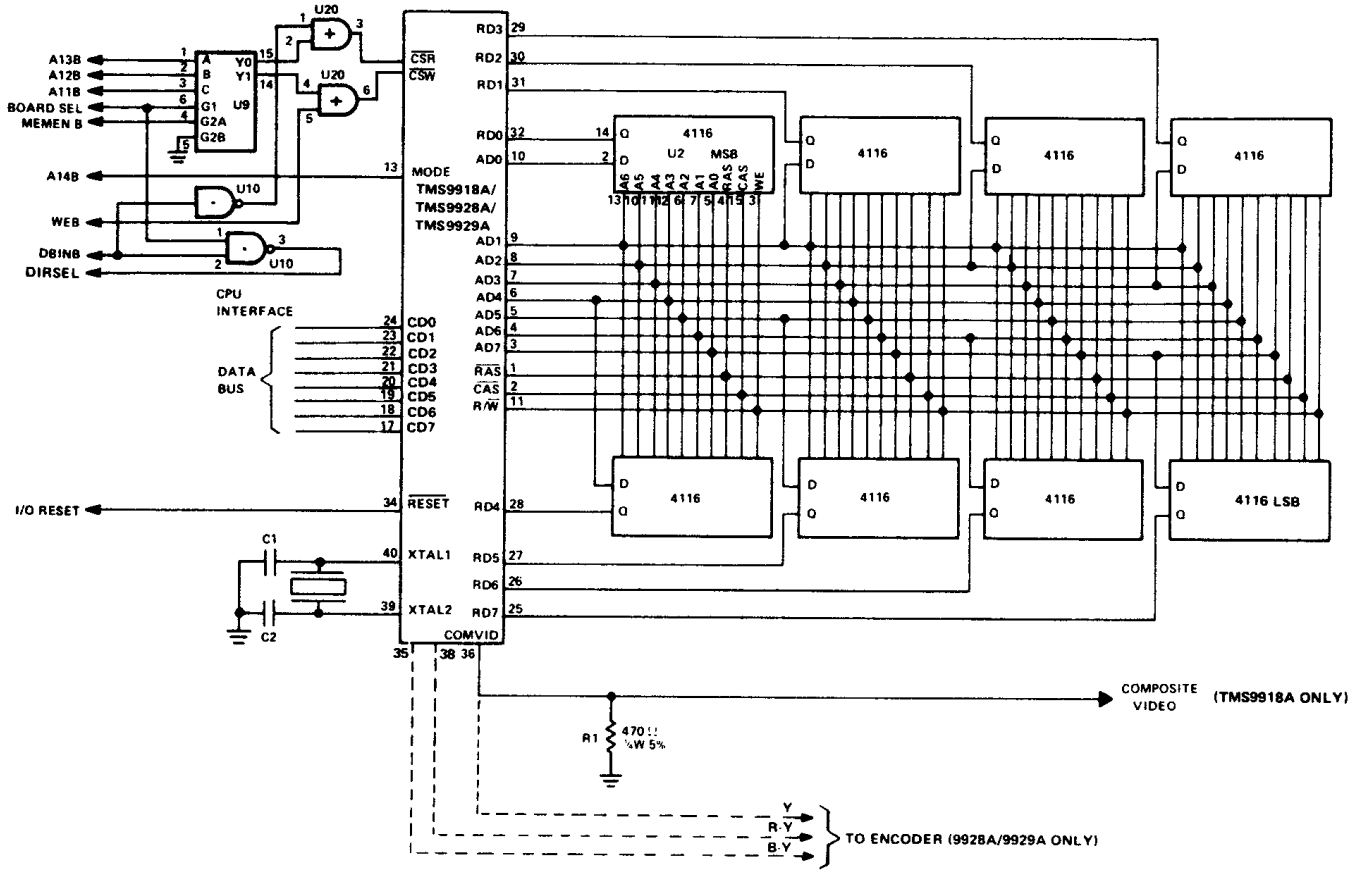


FIGURE 4-2 — TMS9918A/9928A/9929A INTERFACE

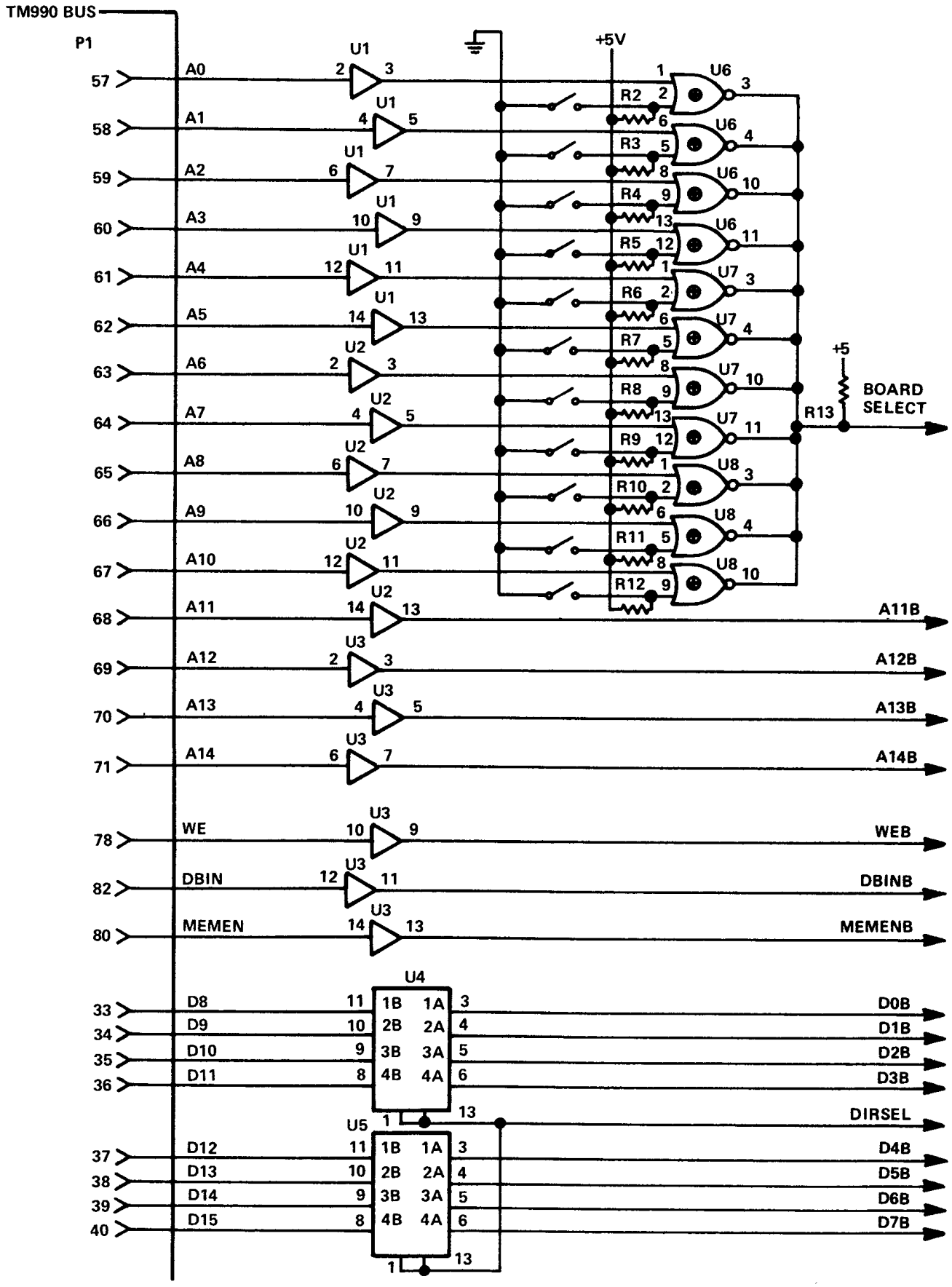


FIGURE 4-3 — TM990 (TMS9918A/9928A/9929A) DEMO BOARD

4.2.1 TM990 (TMS9918A/9928A/9929A) Parts List

U1,2,3	74LS367
U4,5	74LS243
U6,7,8	74LS266
U9	74LS138
U10	74LS00
U11	TMS9918A/9928A/9929A
U12-19	TMS4116
C1,2	33 pF
Y1	10.738635 MHz Crystal
SW1-3	4-position DIP Switches
R1	470 Ω 5% 1/4 W
R2-R13	Bourns XXXX or equivalent

NOTE: All power supply pins of each IC should be bypassed with a .1μF capacitor.

4.2.2 Composite Video Output

The TMS9918A composite video output pin (36), is driven by a source-follower MOS transistor that requires an external pull-down resistor to V_{SS}. A 470-ohm resistor is typically used to provide a 1.9 volt peak-to-peak signal on the output. This output will drive most color directly, although in some cases it may be necessary to provide a simple interface circuit to match the monitor's input requirements. If a color video monitor is not available, an RF modulator can be used to drive the antenna terminals of a standard color television, as shown in Figure 4-4.

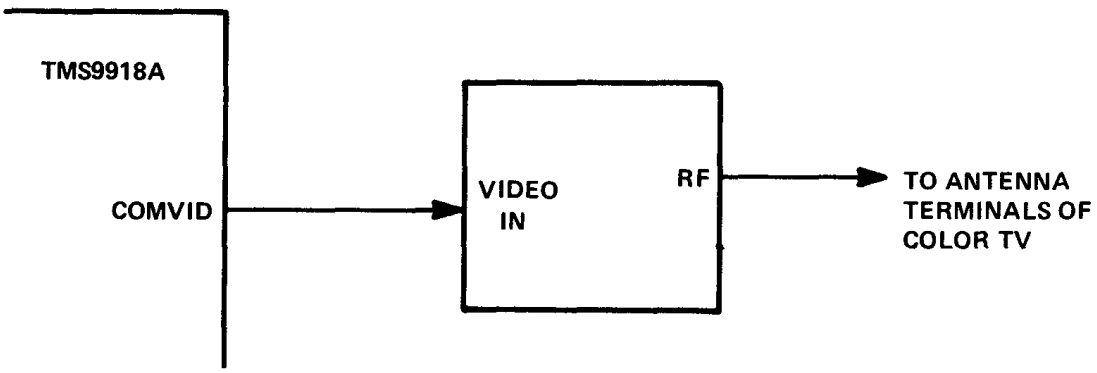


FIGURE 4-4 — RF MODULATOR CONNECTION

4.2.3 Oscillator and Timing

The TMS9918A/9928A/9929A internal timing generation is controlled by a self-contained oscillator and timing circuits. A 10.738635 (± 0.005%) MHz fundamental-frequency parallel-mode crystal is used to drive the basic oscillator frequency.

C1 and C2 are load capacitors for the parallel-resonant crystal. C1 and C2 values may be varied slightly to obtain more accuracy in timing and color generation and also to compensate for stray capacitance on the PC board. Typical values for C1 and C2 range between 15 pF and 39 pF. A trimmer capacitor with a value of 5 pF to 50 pF may also be used instead of C1 and adjusted to provide proper colors to the video monitor.

The VDP may also be operated with an external oscillator source. The VDP connections for this external source are shown in Figure 4-5.

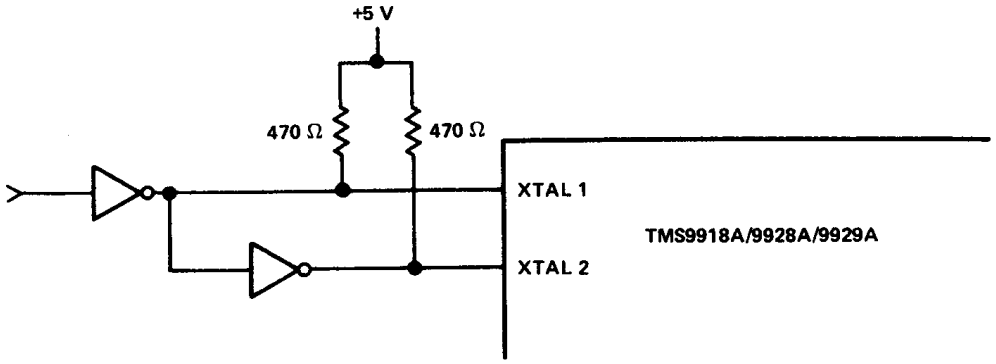


FIGURE 4-5 — EXTERNAL FREQUENCY SOURCE

There may be a slight color shift or a complete color loss in applications of RF modulators if there are mismatches in voltages levels or impedances between the VDP and the RF modulator. See Figure 3-4 for the TMS9928A/9929A interface.

4.2.4 VRAM Connections

The VRAM used in Figure 4-2 are 4116-type dynamic RAMs that meet the specifications in Section 5.

Addressing of the VRAM is done through the address bus and the memory control lines, AD1-AD7 and RAS, CAS, and WR, respectively.

Data written to the VRAM is also sent over the address bus. AD0 is a MSB, and AD7 is the LSB. Data written from the VRAM is brought into the VDP via the read data bus, RD0-RD7. The TMS9918A automatically refreshes the VRAM with no interaction necessary from the host CPU.

Note that address 0 (AD0) and data 0 (D0) are the MSBs for the TMS9918A and all other TMS9900 family members. The VRAM pin designations (A0 and D0) referenced in the data manual are shown as being the LSBs to be consistent with 4116-type dynamic RAM data sheets.

4.3 VDP INITIALIZATION

After powerup and proper reset timing, the VRAM allocation backdrop color and type of dynamic RAM need to be loaded into the VDP registers.

The values to be loaded can be calculated by using the examples and tables shown in Appendix A. The following flowchart (Figure 4-6) shows a procedure for loading all eight VDP registers. Setting 4.4 contains a typical TMS9900 software program designed to work on the demo board, shown in Figure 4-3.

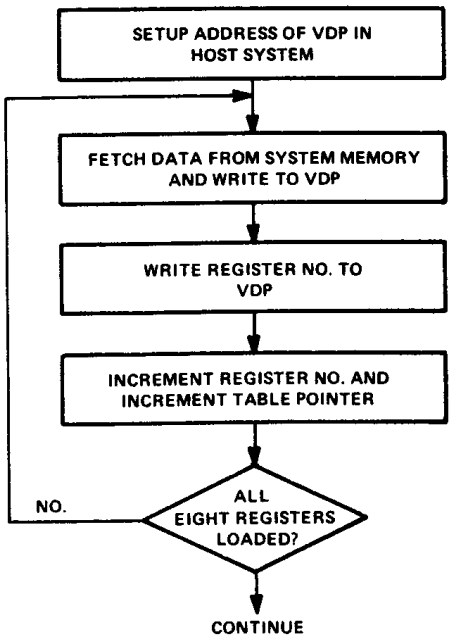


FIGURE 4-6 — VDP REGISTER INITIALIZATION PROCEDURE

4.4 TYPICAL SOFTWARE PROGRAM

4.4.1 General

This program initializes the TMS9918A and loads the Pattern Generator with the upper case character set. It then loads the color table, clears the screen and prints a sign-on message. After initialization, a user program address can be inserted at location 00A4.

```
DEMO9918   SDSMAC 3.4.0 81.117   15:45:22 MONDAY, SEP 27, 1982.
                                                          PAGE 0002

0001                    IDT 'DEMO9918'
0002 0000               AORG >0000
0003       9000  VRAMW  EQU  >9000 ADDRESS TO WRITE DATA TO VRAM
0004       9002  VDPW   EQU  >9002 ADDRESS TO WRITE DATA TO VDP
0005       9004  VRAMR  EQU  >9004 ADDRESS TO READ DATA FROM VRAM
0006       9006  VDPR   EQU  >9006 ADDRESS TO READ VDP STATUS REGISTER
0007             *
0008             ***************************************************************
0009             *          INITIALIZE THE 9918 WITH THE FOLLOWING:
0010             *
0011             *          REG 0 = 00  EXT VID OFF, GRAPH 2 OFF
0012             *
0013             *          REG 1 = 02  4116, INT DIS, VID ON, GRAPH 1
0014             *                      SIZE 1, MAG OFF
0015             *
0016             *          REG 2 = 01  NAME TABLE SUB BLOCK      @>400
0017             *
0018             *          REG 3 = 08  COLOR TABLE SUB BLOCK     @>200
0019             *
0020             *          REG 4 = 01  PATTERN GEN SUB BLOCK     @>800
0021             *
0022             *          REG 5 = 06  SPRITE NAME TAB SUB BLK   @>300
0023             *
0024             *          REG 6 = 00  SPRITE PATT GEN SUB BLK   @>000
0025             *
0026             *          REG 7 = 07  BACKDROP COLOR IS CYAN
0027             *
0028             ***************************************************************
0029             *
0030             *
0031             *                              NOTE
0032             *          THIS SOFTWARE ASSUMES THAT THE DATA BUS OF THE
0033             *          TMS9918A IS CONNECTED TO THE LEAST SIGNIFICANT
0034             *          BYTE OF THE TMS9900, WITH D7 AS THE MOST
0035             *          SIGNIFICANT BIT AND D15 AS THE LEAST SIGNIFICANT
0036             *          BIT
0037             *
0038             *
0039 0000 0201  INIT   LI   R1,VDPW          VDP WRITE ADDRESS
     0002 9002
0040 0004 0202         LI   R2,SUTA          "SET UP TABLE" ADDRESS
     0006 00B8
0041 0008 0203         LI   R3,>80           ADDRESS OF FIRST VDP REGISTER
     000A 0080
0042 000C C472  LP01   MOV  *R2+,*R1         GET DATA FROM MEM, SEND TO 9918
0043 000E C443         MOV  R3,*R1           SEND REG# TO 9918
0044 0010 0583         INC  R3               INCREMENT REGISTER COUNT
0045 0012 0283         CI   R3,>88           ALL REGS LOADED?
     0014 0088
0046 0016 16FA         JNE  LP01             NO,GO AGAIN
0047             *
0048             *
0049             *          LOAD PROGRAM LOADS THE TEXT PATTERNS FROM
0050             *          A TABLE IN MEMORY TO THE PATTERN GENERATOR
0051             *          SUB-BLOCK IN VRAM.
0052             *          ASCII >20 TO >5F ARE INCLUDED  IN THIS TABLE.
0053             *
0054             *
0055 0018 0201  LPG1   LI   R1,VRAMW         ADDRESS TO WRITE DATA TO VRAM
     001A 9000
```

```
0056 001C 0202          LI   R2,VDPW          ADDRESS TO WRITE TO VDP
     001E 9002
0057 0020 0203          LI   R3,PATT          MEM ADDR OF PATTERNS
     0022 00C0
0058 0024 0204          LI   R4,512           64 CHAR X 8 BYTES - 512 BYTES
     0026 0200
0059 0028 0205          LI   R5,>4900         ADDRESS TO LOAD PATS IN VRAM
     002A 4900
0060 002C C485          MOV  R5,*R2           SEND LSB OF VRAM ADDRESS TO VDP
0061 002E 06C5          SWPB R5               REVERSE BYTES
0062 0030 C445          MOV  R5,*R1           SEND DATA TO VRAM
0063 0032 D173  LPG2    MOVB *R3+,R5          GET BYTE FROM MEM
0064 0034 06C5          SWPB R5               REVERSE BYTES
0065 0036 C445          MOV  R5,*R1           SEND DATA TO VRAM
0066 0038 0604          DEC  R4               ALL DONE YET?
0067 003A 16FB          JNE  LPG2             NO, GO AGAIN
0068                *
0069                ********************************************************
0070                *                    LOAD COLOR TABLE
0071                *
0072                *           THIS ROUTINE LOADS THE COLOR TABLE FOR THE
0073                *           TEXT PATTERNS JUST ENTERED.
0074                *
0075                *
0076 003C 0201          LI   R1,VRAMW         ADDRESS TO WRITE DATA TO VRAM
     003E 9000
0077 0040 0202          LI   R2,VDPW          ADDRESS TO WRITE TO VDP
     0042 9002
0078 0044 0203          LI   R3,>4204         START ADDRESS OF TEXT COLOR TABL
     0046 4204
0079 0048 0204          LI   R4,>5F           CHARACTERS WILL BE BLUE ON WHITE
     004A 005F
0080 004C C483          MOV  R3,*R2           SEND LSB OF VRAM ADDRESS TO VDP
0081 004E 0205          LI   R5,8             LOAD COUNT VALUE, 64CHAR/8 = 8
     0050 0008
0082 0052 C444  LCTL    MOV  R4,*R1           SEND COLOR INFO TO VRAM
0083 0054 0605          DEC  R5               TABLE LAODED YET?
0084 0056 16FD          JNE  LCTL             NO, GO AGAIN
0085                *
0086                ************************************************************
0087                *                                  CLEAR SCREEN
0088                *
0089                *           THIS ROUTINE CLEARS THE SCREEN BY WRITING A SPACE
0090                *           CHARACTER (ASCII >20) TO ALL LOCATIONS IN THE
0091                *           NAME TABLE.
0092                *
0093                *
0094 0058 0201          LI   R1,VRAMW         ADDRESS TO WRITE DATA TO VRAM
     005A 9000
0095 005C 0202          LI   R2,VDPW          ADDRESS TO WRITE TO VDP
     005E 9002
0096 0060 0203          LI   R3,>4400         START ADDRESS IN NAME TABLE
     0062 4400
0097 0064 C483          MOV  R3,*R2           SEND MSB OF VRAM ADDRESS TO VDP
0098 0066 0202          LI   R2,768           #OF POSITIONS ON SCREEN
     0068 0300
0099 006A 0203          LI   R3,>20           ASCII SPACE CHAR
     006C 0020
0100 006E C443  CSL1    MOV  R3,*R1           SEND SPACE TO SCREEN
0101 0070 0602          DEC  R2               ARE ALL LOCATIONS CLEAR?
```

```
0102 0072 16FD          JNE  CSL1               NO, GO AGAIN
0103                    *
0104                    ************************************************************
0105                    *                    PRINT SIGN ON MESSAGE
0106                    *                 AND BRANCH TO USERS PROGRAM
0107                    *
0108                    *
0109 0074 0201          LI   R1,VRAMW           ADDRESS TO WRITE DATA TO VRAM
     0076 9000
0110 0078 0202          LI   R2,VDPW            ADDRESS TO WRITE TO VDP
     007A 9002
0111 007C 0203          LI   R3,>4400           POSITION OF MESSAGE ON SCREEN
     007E 4400
0112 0080 C483          MOV  R3,*R2             SEND MSB OF VRAM ADDRESS TO VDP
0113 0082 06C3          SWPB R3                 REVERSE BYTES
0114 0084 C483          MOV  R3,*R2             SEND MSB OF VRAM ADDRESS TO VDP
0115 0086 0203          LI   R3,MSG0            ADDRESS OF SIGN ON MESSAGE
     0088 009E
0116 008A 04C4  PRNT    CLR  R4                 CLEAR RECEPTION REGISTER
0117 008C D113          MOVB *R3,R4             GET A BYTE OF TEXT
0118 008E 0284          CI   R4,>FF00           IS IT THE EOM CHARACTER?
     0090 FF00
0119 0092 1303          JEQ  DONE               YES, GOTO NEXT PROGRAM SEGMENT
0120 0094 06C4          SWPB R4                 REVERSE  BYTES
0121 0096 C444          MOV  R4,*R1             SEND CHAR TO VRAM
0122 0098 10F8          JMP  PRNT               GET NEXT CHARACTER
0123 009A 0460  DONE    B    @DONE              INSERT BRANCH TO USERS PROGRAM
     009C 009A
0124                    *                       AT THIS POINT
0125 009E   54  MSG0    TEXT 'TEXAS INSTRUMENTS TMS9918'
     009F   45
     00A0   58
     00A1   41
     00A2   53
     00A3   20
     00A4   49
     00A5   4E
     00A6   53
     00A7   54
     00A8   52
     00A9   55
     00AA   4D
     00AB   45
     00AC   4E
     00AD   54
     00AE   53
     00AF   20
     00B0   54
     00B1   4D
     00B2   53
     00B3   39
     00B4   39
     00B5   31
     00B6   38
0126 00B7   FF          BYTE >FF
0127 00B8               EVEN
0128                    *
0129                    ************************************************************
0130                    *           THIS TABLE CONTAINS THE VALUES FOR
0131                    *         INITIALIZING THE REGISTERS IN THE 9918A
```

```
0132                    *
0133 00B8   00  SUTA    BYTE >00
0134 00B9   02          BYTE >02
0135 00BA   01          BYTE >01
0136 00BB   08          BYTE >08
0137 00BC   01          BYTE >01
0138 00BD   06          BYTE >06
0139 00BE   00          BYTE >00
0140 00BF   07          BYTE >07
0141                    *
0142                    ***************************************************************
0143                    *                   9918A TEXT PATTERNS
0144                    *
0145                    *           THESE PATTERNS FROM A 5X7 CHARACTER IN THE
0146                    *           8X8 PATTERN BLOCK THAT IS UPPER AND LEFT
0147                    *           JUSTIFIED
0148                    *
0149 00C0 0000  PATT    DATA >0000          CHARACTER SPACE ASCII 20
0150 00C2 0000          DATA >0000
0151 00C4 0000          DATA >0000
0152 00C6 0000          DATA >0000
0153 00C8 2020          DATA >2020          CHARACTER !     ASCII 21
0154 00CA 2020          DATA >2020
0155 00CC 2000          DATA >2000
0156 00CE 2000          DATA >2000
0157 00D0 5050          DATA >5050          CHARACTER "     ASCII 22
0158 00D2 5000          DATA >5000
0159 00D4 0000          DATA >0000
0160 00D6 0000          DATA >0000
0161 00D8 5050          DATA >5050          CHARACTER #     ASCII 23
0162 00DA F850          DATA >F850
0163 00DC F850          DATA >F850
0164 00DE 5000          DATA >5000
0165 00E0 2078          DATA >2078          CHARACTER $     ASCII 24
0166 00E2 A070          DATA >A070
0167 00E4 28F0          DATA >28F0
0168 00E6 2000          DATA >2000
0169 00E8 C0C8          DATA >C0C8          CHARACTER %     ASCII 25
0170 00EA 1020          DATA >1020
0171 00EC 4098          DATA >4098
0172 00EE 1800          DATA >1800
0173 00F0 40A0          DATA >40A0          CHARACTER &     ASCII 26
0174 00F2 A040          DATA >A040
0175 00F4 A890          DATA >A890
0176 00F6 6800          DATA >6800
0177 00F8 2020          DATA >2020          CHARACTER '     ASCII 27
0178 00FA 2000          DATA >2000
0179 00FC 0000          DATA >0000
0180 00FE 0000          DATA >0000
0181 0100 2040          DATA >2040          CHARACTER (     ASCII 28
0182 0102 8080          DATA >8080
0183 0104 8040          DATA >8040
0184 0106 2000          DATA >2000
0185 0108 2010          DATA >2010          CHARACTER )     ASCII 29
0186 010A 0808          DATA >0808
0187 010C 0810          DATA >0810
0188 010E 2000          DATA >2000
0189 0110 20A8          DATA >20A8          CHARACTER *     ASCII 2A
0190 0112 7020          DATA >7020
0191 0114 70A8          DATA >70A8
```

```
0192 0116 2000          DATA >2000
0193 0118 0020          DATA >0020          CHARACTER +     ASCII 2B
0194 011A 20F8          DATA >20F8
0195 011C 2020          DATA >2020
0196 011E 0000          DATA >0000
0197 0120 0000          DATA >0000          CHARACTER ,     ASCII 2C
0198 0122 0000          DATA >0000
0199 0124 2020          DATA >2020
0200 0126 4000          DATA >4000
0201 0128 0000          DATA >0000          CHARACTER -     ASCII 2D
0202 012A 00F8          DATA >00F8
0203 012C 0000          DATA >0000
0204 012E 0000          DATA >0000
0205 0130 0000          DATA >0000          CHARACTER .     ASCII 2E
0206 0132 0000          DATA >0000
0207 0134 0000          DATA >0000
0208 0136 2000          DATA >2000
0209 0138 0008          DATA >0008          CHARACTER /     ASCII 2F
0210 013A 1020          DATA >1020
0211 013C 4080          DATA >4080
0212 013E 0000          DATA >0000
0213 0140 7088          DATA >7088          CHARACTER 0     ASCII 30
0214 0142 98A8          DATA >98A8
0215 0144 C888          DATA >C888
0216 0146 7000          DATA >7000
0217 0148 2060          DATA >2060          CHARACTER 1     ASCII 31
0218 014A 2020          DATA >2020
0219 014C 2020          DATA >2020
0220 014E 7000          DATA >7000
0221 0150 7088          DATA >7088          CHARACTER 2     ASCII 32
0222 0152 0830          DATA >0830
0223 0154 4080          DATA >4080
0224 0156 F800          DATA >F800
0225 0158 F808          DATA >F808          CHARACTER 3     ASCII 33
0226 015A 1030          DATA >1030
0227 015C 0888          DATA >0888
0228 015E 7000          DATA >7000
0229 0160 1030          DATA >1030          CHARACTER 4     ASCII 34
0230 0162 5090          DATA >5090
0231 0164 F810          DATA >F810
0232 0166 1000          DATA >1000
0233 0168 F880          DATA >F880          CHARACTER 5     ASCII 35
0234 016A F008          DATA >F008
0235 016C 0888          DATA >0888
0236 016E 7000          DATA >7000
0237 0170 3840          DATA >3840          CHARACTER 6     ASCII 36
0238 0172 80F0          DATA >80F0
0239 0174 8888          DATA >8888
0240 0176 7000          DATA >7000
0241 0178 F808          DATA >F808          CHARACTER 7     ASCII 37
0242 017A 1020          DATA >1020
0243 017C 4040          DATA >4040
0244 017E 4000          DATA >4000
0245 0180 7088          DATA >7088          CHARACTER 8     ASCII 38
0246 0182 8870          DATA >8870
0247 0184 8888          DATA >8888
0248 0186 7000          DATA >7000
0249 0188 7088          DATA >7088          CHARACTER 9     ASCII 39
0250 018A 8878          DATA >8878
0251 018C 0810          DATA >0810
```

```
0252 018E E000          DATA >E000
0253 0190 0000          DATA >0000          CHARACTER :     ASCII 3A
0254 0192 2000          DATA >2000
0255 0194 2000          DATA >2000
0256 0196 0000          DATA >0000
0257 0198 0000          DATA >0000          CHARACTER ;     ASCII 3B
0258 019A 2000          DATA >2000
0259 019C 2020          DATA >2020
0260 019E 4000          DATA >4000
0261 01A0 1020          DATA >1020          CHARACTER <     ASCII 3C
0262 01A2 4080          DATA >4080
0263 01A4 4020          DATA >4020
0264 01A6 1000          DATA >1000
0265 01A8 0000          DATA >0000          CHARACTER =     ASCII 3D
0266 01AA F800          DATA >F800
0267 01AC F800          DATA >F800
0268 01AE 0000          DATA >0000
0269 01B0 4020          DATA >4020          CHARACTER >     ASCII 3E
0270 01B2 1008          DATA >1008
0271 01B4 1020          DATA >1020
0272 01B6 4000          DATA >4000
0273 01B8 7088          DATA >7088          CHARACTER ?     ASCII 3F
0274 01BA 1020          DATA >1020
0275 01BC 2000          DATA >2000
0276 01BE 2000          DATA >2000
0277 01C0 7088          DATA >7088          CHARACTER @     ASCII 40
0278 01C2 A8B8          DATA >A8B8
0279 01C4 B080          DATA >B080
0280 01C6 7800          DATA >7800
0281 01C8 2050          DATA >2050          CHARACTER A     ASCII 41
0282 01CA 8888          DATA >8888
0283 01CC F888          DATA >F888
0284 01CE 8800          DATA >8800
0285 01D0 F088          DATA >F088          CHARACTER B     ASCII 42
0286 01D2 88F0          DATA >88F0
0287 01D4 8888          DATA >8888
0288 01D6 F000          DATA >F000
0289 01D8 7088          DATA >7088          CHARACTER C     ASCII 43
0290 01DA 8080          DATA >8080
0291 01DC 8088          DATA >8088
0292 01DE 7000          DATA >7000
0293 01E0 F088          DATA >F088          CHARACTER D     ASCII 44
0294 01E2 8888          DATA >8888
0295 01E4 8888          DATA >8888
0296 01E6 F000          DATA >F000
0297 01E8 F880          DATA >F880          CHARACTER E     ASCII 45
0298 01EA 80F0          DATA >80F0
0299 01EC 8080          DATA >8080
0300 01EE F800          DATA >F800
0301 01F0 F880          DATA >F880          CHARACTER F     ASCII 46
0302 01F2 80F0          DATA >80F0
0303 01F4 8080          DATA >8080
0304 01F6 8000          DATA >8000
0305 01F8 7880          DATA >7880          CHARACTER G     ASCII 47
0306 01FA 8080          DATA >8080
0307 01FC 9888          DATA >9888
0308 01FE 7800          DATA >7800
0309 0200 8888          DATA >8888          CHARACTER H     ASCII 48
0310 0202 88F8          DATA >88F8
0311 0204 8888          DATA >8888
```

```
0312 0206 8800          DATA >8800
0313 0208 7020          DATA >7020          CHARACTER I    ASCII 49
0314 020A 2020          DATA >2020
0315 020C 2020          DATA >2020
0316 020E 7000          DATA >7000
0317 0210 0808          DATA >0808          CHARACTER J    ASCII 4A
0318 0212 0808          DATA >0808
0319 0214 0888          DATA >0888
0320 0216 7000          DATA >7000
0321 0218 8890          DATA >8890          CHARACTER K    ASCII 4B
0322 021A A0C0          DATA >A0C0
0323 021C A090          DATA >A090
0324 021E 8800          DATA >8800
0325 0220 8080          DATA >8080          CHARACTER L    ASCII 4C
0326 0222 8080          DATA >8080
0327 0224 8080          DATA >8080
0328 0226 F800          DATA >F800
0329 0228 88D8          DATA >88D8          CHARACTER M    ASCII 4D
0330 022A A8A8          DATA >A8A8
0331 022C 8888          DATA >8888
0332 022E 8800          DATA >8800
0333 0230 8888          DATA >8888          CHARACTER N    ASCII 4E
0334 0232 C8A8          DATA >C8A8
0335 0234 9888          DATA >9888
0336 0236 8800          DATA >8800
0337 0238 7088          DATA >7088          CHARACTER O    ASCII 4F
0338 023A 8888          DATA >8888
0339 023C 8888          DATA >8888
0340 023E 7000          DATA >7000
0341 0240 F088          DATA >F088          CHARACTER P    ASCII 50
0342 0242 88F0          DATA >88F0
0343 0244 8080          DATA >8080
0344 0246 8000          DATA >8000
0345 0248 7088          DATA >7088          CHARACTER Q    ASCII 51
0346 024A 8888          DATA >8888
0347 024C A890          DATA >A890
0348 024E 6800          DATA >6800
0349 0250 F088          DATA >F088          CHARACTER R    ASCII 52
0350 0252 88F0          DATA >88F0
0351 0254 A090          DATA >A090
0352 0256 8800          DATA >8800
0353 0258 7088          DATA >7088          CHARACTER S    ASCII 53
0354 025A 8070          DATA >8070
0355 025C 0888          DATA >0888
0356 025E 7000          DATA >7000
0357 0260 F820          DATA >F820          CHARACTER T    ASCII 54
0358 0262 2020          DATA >2020
0359 0264 2020          DATA >2020
0360 0266 2000          DATA >2000
0361 0268 8888          DATA >8888          CHARACTER U    ASCII 55
0362 026A 8888          DATA >8888
0363 026C 8888          DATA >8888
0364 026E 7000          DATA >7000
0365 0270 8888          DATA >8888          CHARACTER V    ASCII 56
0366 0272 8888          DATA >8888
0367 0274 8850          DATA >8850
0368 0276 2000          DATA >2000
0369 0278 8888          DATA >8888          CHARACTER W    ASCII 57
0370 027A 88A8          DATA >88A8
0371 027C A8D8          DATA >A8D8
```

```
0372 027E 8800          DATA >8800
0373 0280 8888          DATA >8888          CHARACTER X     ASCII 58
0374 0282 5020          DATA >5020
0375 0284 5088          DATA >5088
0376 0286 8800          DATA >8800
0377 0288 8888          DATA >8888          CHARACTER Y     ASCII 59
0378 028A 5020          DATA >5020
0379 028C 2020          DATA >2020
0380 028E 2000          DATA >2000
0381 0290 F808          DATA >F808          CHARACTER Z     ASCII 5A
0382 0292 1020          DATA >1020
0383 0294 4080          DATA >4080
0384 0296 F800          DATA >F800
0385 0298 F8C0          DATA >F8C0          CHARACTER [     ASCII 5B
0386 029A C0C0          DATA >C0C0
0387 029C C0C0          DATA >C0C0
0388 029E F800          DATA >F800
0389 02A0 0080          DATA >0080          CHARACTER       ASCII 5C
0390 02A2 4020          DATA >4020
0391 02A4 1008          DATA >1008
0392 02A6 0000          DATA >0000
0393 02A8 F818          DATA >F818          CHARACTER ]     ASCII 5D
0394 02AA 1818          DATA >1818
0395 02AC 1818          DATA >1818
0396 02AE F800          DATA >F800
0397 02B0 0000          DATA >0000          CHARACTER       ASCII 5E
0398 02B2 2050          DATA >2050
0399 02B4 8800          DATA >8800
0400 02B6 0000          DATA >0000
0401 02B8 0000          DATA >0000          CHARACTER _     ASCII 5F
0402 02BA 0000          DATA >0000
0403 02BC 0000          DATA >0000
0404 02BE F800          DATA >F800
0405 02C0 4020          DATA >4020          CHARACTER       ASCII 60
0406 02C2 1000          DATA >1000
0407 02C4 0000          DATA >0000
0408 02C6 0000          DATA >0000
0409 02C8 0000          DATA >0000          CHARACTER a     ASCII 61
0410 02CA 7088          DATA >7088
0411 02CC F888          DATA >F888
0412 02CE 8800          DATA >8800
0413 02D0 0000          DATA >0000          CHARACTER b     ASCII 62
0414 02D2 F048          DATA >F048
0415 02D4 7048          DATA >7048
0416 02D6 F000          DATA >F000
0417 02D8 0000          DATA >0000          CHARACTER c     ASCII 63
0418 02DA 7880          DATA >7880
0419 02DC 8080          DATA >8080
0420 02DE 7800          DATA >7800
0421 02E0 0000          DATA >0000          CHARACTER d     ASCII 64
0422 02E2 F048          DATA >F048
0423 02E4 4848          DATA >4848
0424 02E6 F000          DATA >F000
0425 02E8 0000          DATA >0000          CHARACTER e     ASCII 65
0426 02EA F080          DATA >F080
0427 02EC E080          DATA >E080
0428 02EE F000          DATA >F000
0429 02F0 0000          DATA >0000          CHARACTER f     ASCII 66
0430 02F2 F080          DATA >F080
0431 02F4 E080          DATA >E080
```

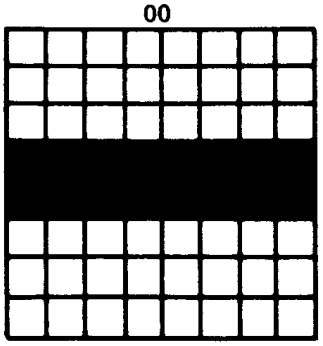
```
0432 02F6 8000          DATA >8000
0433 02F8 0000          DATA >0000          CHARACTER g    ASCII 67
0434 02FA 7880          DATA >7880
0435 02FC B888          DATA >B888
0436 02FE 7000          DATA >7000
0437 0300 0000          DATA >0000          CHARACTER h    ASCII 68
0438 0302 8888          DATA >8888
0439 0304 F888          DATA >F888
0440 0306 8800          DATA >8800
0441 0308 0000          DATA >0000          CHARACTER i    ASCII 69
0442 030A F820          DATA >F820
0443 030C 2020          DATA >2020
0444 030E F800          DATA >F800
0445 0310 0000          DATA >0000          CHARACTER j    ASCII 6A
0446 0312 7020          DATA >7020
0447 0314 20A0          DATA >20A0
0448 0316 E000          DATA >E000
0449 0318 0000          DATA >0000          CHARACTER k    ASCII 6B
0450 031A 90A0          DATA >90A0
0451 031C A0C0          DATA >A0C0
0452 031E 9000          DATA >9000
0453 0320 0000          DATA >0000          CHARACTER l    ASCII 6C
0454 0322 8080          DATA >8080
0455 0324 8080          DATA >8080
0456 0326 F800          DATA >F800
0457 0328 0000          DATA >0000          CHARACTER m    ASCII 6D
0458 032A 88D8          DATA >88D8
0459 032C A888          DATA >A888
0460 032E 8800          DATA >8800
0461 0330 0000          DATA >0000          CHARACTER n    ASCII 6E
0462 0332 88C8          DATA >88C8
0463 0334 A898          DATA >A898
0464 0336 8800          DATA >8800
0465 0338 0000          DATA >0000          CHARACTER o    ASCII 6F
0466 033A F888          DATA >F888
0467 033C 8888          DATA >8888
0468 033E F800          DATA >F800
0469 0340 0000          DATA >0000          CHARACTER p    ASCII 70
0470 0342 F088          DATA >F088
0471 0344 F080          DATA >F080
0472 0346 8000          DATA >8000
0473 0348 0000          DATA >0000          CHARACTER q    ASCII 71
0474 034A F888          DATA >F888
0475 034C A890          DATA >A890
0476 034E E000          DATA >E000
0477 0350 0000          DATA >0000          CHARACTER r    ASCII 72
0478 0352 F888          DATA >F888
0479 0354 F8A0          DATA >F8A0
0480 0356 9000          DATA >9000
0481 0358 0000          DATA >0000          CHARACTER s    ASCII 73
0482 035A 7880          DATA >7880
0483 035C 7008          DATA >7008
0484 035E F000          DATA >F000
0485 0360 0000          DATA >0000          CHARACTER t    ASCII 74
0486 0362 F820          DATA >F820
0487 0364 2020          DATA >2020
0488 0366 2000          DATA >2000
0489 0368 0000          DATA >0000          CHARACTER u    ASCII 75
0490 036A 8888          DATA >8888
0491 036C 8888          DATA >8888
```

```
 0492 036E 7000          DATA >7000
 0493 0370 0000          DATA >0000          CHARACTER v     ASCII 76
 0494 0372 8888          DATA >8888
 0495 0374 90A0          DATA >90A0
 0496 0376 4000          DATA >4000
 0497 0378 0000          DATA >0000          CHARACTER w     ASCII 77
 0498 037A 8888          DATA >8888
 0499 037C A8D8          DATA >A8D8
 0500 037E 8800          DATA >8800
 0501 0380 0000          DATA >0000          CHARACTER x     ASCII 78
 0502 0382 8860          DATA >8860
 0503 0384 2060          DATA >2060
 0504 0386 8800          DATA >8800
 0505 0388 0000          DATA >0000          CHARACTER y     ASCII 79
 0506 038A 8850          DATA >8850
 0507 038C 2020          DATA >2020
 0508 038E 2000          DATA >2000
 0509 0390 0000          DATA >0000          CHARACTER z     ASCII 7A
 0510 0392 F810          DATA >F810
 0511 0394 2040          DATA >2040
 0512 0396 F800          DATA >F800
 0513 0398 3840          DATA >3840          CHARACTER       ASCII 7B
 0514 039A 20C0          DATA >20C0
 0515 039C 2040          DATA >2040
 0516 039E 3800          DATA >3800
 0517 03A0 4020          DATA >4020          CHARACTER       ASCII 7C
 0518 03A2 1008          DATA >1008
 0519 03A4 1020          DATA >1020
 0520 03A6 4000          DATA >4000
 0521 03A8 E010          DATA >E010          CHARACTER       ASCII 7D
 0522 03AA 2018          DATA >2018
 0523 03AC 2010          DATA >2010
 0524 03AE E000          DATA >E000
 0525 03B0 40A8          DATA >40A8          CHARACTER       ASCII 7E
 0526 03B2 1000          DATA >1000
 0527 03B4 0000          DATA >0000
 0528 03B6 0000          DATA >0000
 0529 03B8 A850          DATA >A850          CHARACTER       ASCII 7F
 0530 03BA A850          DATA >A850
 0531 03BC A850          DATA >A850
 0532 03BE A800          DATA >A800
 0533                    END
NO ERRORS,     NO WARNINGS
```

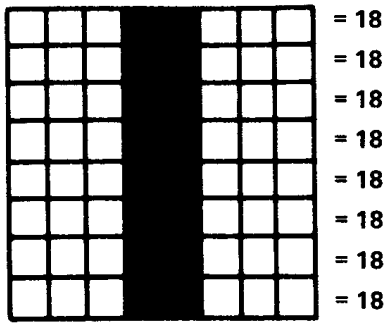
4.5 TMS9900 SOFTWARE SUBROUTINES

NOTE: Before using any of the line drawing subroutines, the "Load Line Drawing Patterns" subroutine must be executed.

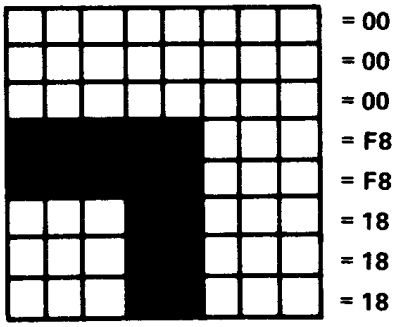
PATTERN 00



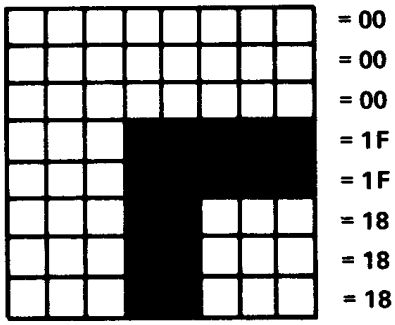
PATTERN 01



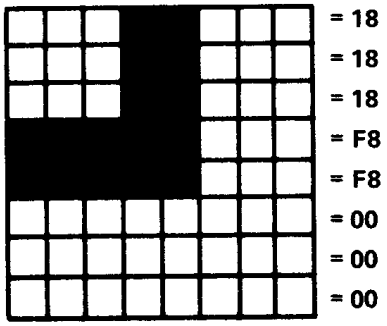
PATTERN 02



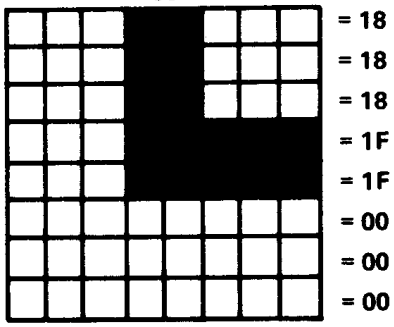
PATTERN 03



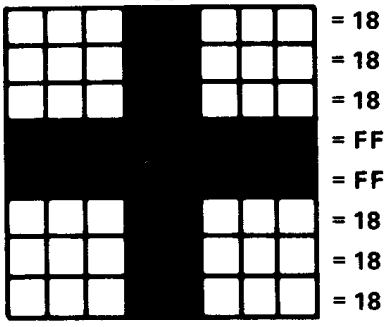
PATTERN 04



PATTERN 05



PATTERN 06



```
0001                    IDT 'SEGMENTS'
0002                ************************************************************
0003                *                                                          *
0004                *                    TMS9918A SUBROUTINES                  *
0005                *                                                          *
0006                ************************************************************
0007        9000    VRAMW   EQU  >9000 ADDRESS TO WRITE DATA TO VRAM
0008        9002    VDPW    EQU  >9002 ADDRESS TO WRITE DATA TO VDP
0009        9004    VRAMR   EQU  >9004 ADDRESS TO READ DATA FROM VRAM
0010        9006    VDPR    EQU  >9006 ADDRESS TO READ STATUS FROM VDP
0011                ************************************************************
0012                *                                                          *
0013                *                  LOAD LINE DRAWING PATTERNS              *
0014                *                                                          *
0015                *          REGISTERS USED:                                 *
0016                *                                                          *
0017                *          REG 1 = RESERVED                                *
0018                *          REG 2 = RESERVED                                *
0019                *          REG 3 = ADDRESS OF PATTERN GENERATOR            *
0020                *          REG 4 = PATTERN LOCATION IN MEMORY              *
0021                *          REG 5 = COLORS OF DRAWING PATTERNS              *
0022                *                  (USER DEFINED)                          *
0023                ************************************************************
0024                *
0025                *
0026 0000 0201          LI   R1,VRAMW          ADDRESS TO WRITE DATA TO VRAM
     0002 9000
0027 0004 0202          LI   R2,VDPW           ADDRESS TO WRITE TO VDP
     0006 9002
0028 0008 0203          LI   R3,>4800          ADDRESS IN PATT GEN FOR PATT 0
     000A 4800
0029 000C 0204          LI   R4,PATD           DRAWING PATTERNS LOCATED IN MEM
     000E 002E'
0030 0010 C483          MOV  R3,*R2            SEND MSB OF VRAM ADDRESS TO VDP
0031 0012 06C3          SWPB R3                REVERSE BYTES
0032 0014 C483          MOV  R3,*R2            SEND LSB OF VRAM ADDRESS TO VDP
0033 0016 0203          LI   R3,56             7 PATTERNS  X 8 BYTES EACH
     0018 0038
0034 001A D474  LLD1    MOVB *R4+,*R1          SEND BYTE TO VRAM
0035 001C 0603          DEC  R3                DECREMENT BYTE COUNT
0036 001E 16FD          JNE  LLD1              IF NOT DONE, GET NEXT BYTE
0037 0020 0203          LI   R3,>4200          ADDRESS OF COLOR TABLE
     0022 4200
0038 0024 C483          MOV  R3,*R2            SEND LSB OF VRAM ADDRESS TO VDP
0039 0026 06C3          SWPB R3                REVERSE BYTES
0040 0028 C483          MOV  R3,*R2            SEND MSB OF VRAM ADDRESS TO VDP
0041 002A C445          MOV  R5,*R1            SEND COLOR BYTE TO VRAM
0042 002C 045B          B    *R11              RETURN TO CALLING PROGRAM
0043                *
0044                *
0045                *     PATTERNS FOR LINE DRAWING
0046                *
0047                *
0048 002E 0000  PATD    DATA >0000         PATTERN  00
0049 0030 00FF          DATA >00FF
0050 0032 FF00          DATA >FF00
0051 0034 0000          DATA >0000
0052 0036 1818          DATA >1818         PATTERN  01
0053 0038 1818          DATA >1818
0054 003A 1818          DATA >1818
```

```
0055 003C 1818          DATA >1818
0056 003E 0000          DATA >0000        PATTERN  02
0057 0040 00F8          DATA >00F8
0058 0042 F818          DATA >F818
0059 0044 1818          DATA >1818
0060 0046 0000          DATA >0000        PATTERN  03
0061 0048 001F          DATA >001F
0062 004A 1F18          DATA >1F18
0063 004C 1818          DATA >1818
0064 004E 1818          DATA >1818        PATTERN  04
0065 0050 18F8          DATA >18F8
0066 0052 F800          DATA >F800
0067 0054 0000          DATA >0000
0068 0056 1818          DATA >1818        PATTERN  05
0069 0058 181F          DATA >181F
0070 005A 1F00          DATA >1F00
0071 005C 0000          DATA >0000
0072 005E 1818          DATA >1818        PATTERN  06
0073 0060 18FF          DATA >18FF
0074 0062 FF18          DATA >FF18
0075 0064 1818          DATA >1818
0076              *
0077              *
0078              ************************************************************
0079              *                                                          *
0080              *                LOAD SPRITES SUBROUTINE                   *
0081              *                                                          *
0082              *      REGISTERS USED:                                     *
0083              *                                                          *
0084              *      R1 = RESERVED                                       *
0085              *      R2 = RESERVED                                       *
0086              *      R3 = ADDRESS OF SPRITE TABLE IN VRAM                *
0087              *      R4 = MEMORY ADDRESS OF SPRITE TABLE (USER DEF)      *
0088              *      R5 = NUMBER OF BYTES TO TRANSFER (USER DEF)         *
0089              *                                                          *
0090              ************************************************************
0091              *
0092 0066 0201          LI   R1,VRAMW          ADDRESS TO WRITE DATA TO VRAM
     0068 8C00
0093 006A 0202          LI   R2,VDPW           ADDRESS TO WRITE TO VDP
     006C 8C02
0094 006E 0203          LI   R3,>4000          ADDRESS OF SPRITE TABLE IN VRAM
     0070 4000
0095 0072 C483          MOV  R3,*R2            SEND LSB OF VRAM ADDRESS TO VDP
0096 0074 06C3          SWPB R3                REVERSE BYTES
0097 0076 C483          MOV  R3,*R2            SEND MSB OF VRAM ADDRESS TO VDP
0098 0078 D0F4  LDPL    MOVB *R4+,R3           GET BYTE OF DATA FROM MEM
0099 007A 06C3          SWPB R3                REVERSE BYTES
0100 007C C443          MOV  R3,*R1            SEND DATA TO VRAM
0101 007E 0605          DEC  R5                ARE WE DONE YET?
0102 0080 16FB          JNE  LDPL              NO, GO AGAIN
0103 0082 045B          B    *R11              YES, RETURN TO CALLING PROGRAM
```

```
0105                    ************************************************************
0106                    *                                                          *
0107                    *                CLEAR SCREEN SUBROUTINE                   *
0108                    *                                                          *
0109                    *     REGISTERS USED:                                      *
0110                    *                                                          *
0111                    *     R1 = RESERVED                                        *
0112                    *     R2 = RESERVED                                        *
0113                    *     R3 = START ADDRESS ON SCREEN                         *
0114                    *                                                          *
0115                    ************************************************************
0116                    *
0117                    *
0118 0084 0201  CLSC    LI    R1,VRAMW          ADDRESS TO WRITE DATA TO VRAM
     0086 9000
0119 0088 0202          LI    R2,VDPW           ADDRESS TO WRITE TO VDP
     008A 9002
0120 008C 0203          LI    R3,>4400          START LOCATION OF THE NAME TABLE
     008E 4400
0121 0090 C483          MOV   R3,*R2            SEND LSB OF VRAM ADDRESS TO VDP
0122 0092 06C3          SWPB  R3                REVERSE BYTES
0123 0094 C483          MOV   R3,*R2            SEND MSB OF VRAM ADDRESS TO VDP
0124 0096 0202          LI    R2,768            #OF POSITIONS ON SCREEN
     0098 0300
0125 009A 0203          LI    R3,>20            ASCII SPACE CHAR
     009C 0020
0126 009E C443  CSL1    MOV   R3,*R1            SEND SPACE CHAR TO VRAM
0127 00A0 0602          DEC   R2                ARE ALL LOCATIONS CLEAR?
0128 00A2 16FD          JNE   CSL1              NO, GO AGAIN
0129 00A4 045B          B     *R11              YES, RETURN TO CALLING PROGRAM
0130                    *
0131                    *
0132                    ************************************************************
0133                    *                                                          *
0134                    *                 PRINT MESSAGE SUBROUTINE                 *
0135                    *                AND BRANCH TO USERS PROGRAM               *
0136                    *                                                          *
0137                    *          REGISTERS USED:                                 *
0138                    *                                                          *
0139                    *          R1 = RESERVED                                   *
0140                    *          R2 = RESERVED                                   *
0141                    *          R3 = STARTING ADDRESS OF MESSAGE IN NAME        *
0142                    *               TABLE (USER DEFINED)                       *
0143                    *          R4 = MEMORY ADDRESS  OF MESSAGE (USER DEFINED)  *
0144                    *                                                          *
0145                    *          NOTE:  END MESSAGE STRING WITH A BYTE 00        *
0146                    *                                                          *
0147                    ************************************************************
0148                    *
0149                    *
0150 00A6 0201  PRNT    LI    R1,VRAMW          ADDRESS TO WRITE DATA TO VRAM
     00A8 9000
0151 00AA 0202          LI    R2,VDPW           ADDRESS TO WRITE TO VDP
     00AC 9002
0152 00AE C483          MOV   R3,*R2            SEND LSB OF VRAM ADDRESS TO VDP
0153 00B0 06C3          SWPB  R3                REVERSE BYTES
0154 00B2 C483          MOV   R3,*R2            SEND MSB OF VRAM ADDRESS TO VDP
0155 00B4 D0B4  PRL1    MOVB  *R4+,R2           GET BYTE OF TEXT FROM MEM
0156 00B6 1303          JEQ   PRL2              IF ZERO, THEN END OF MESS
0157 00B8 06C2          SWPB  R2                INDEX BYTE INTO POSITION
```

```
0158 00BA C442          MOV  R2,*R1          SEND CHAR TO VRAM
0159 00BC 10FB          JMP  PRL1            GET NEXT CHAR
0160 00BE 045B  PRL2    B    *R11            RETURN TO CALLING PROGRAM
0161                    *
0162                    *
0163                    ************************************************************
0164                    *                                                          *
0165                    *            ERASE TO END OF SCREEN SUBROUTINE             *
0166                    *                                                          *
0167                    *      REGISTERS USED:                                     *
0168                    *                                                          *
0169                    *      R1 = RESERVED                                       *
0170                    *      R2 = RESERVED                                       *
0171                    *      R3 = ADDRESS IN NAME TABLE TO START ERASURE         *
0172                    *           (USER DEFINED), R3 MUST BE EQUAL  TO           *
0173                    *           OR GREATER THAN >4400, AND MUST BE LESS        *
0174                    *           THAN OR EQUAL TO >46FF                         *
0175                    *                                                          *
0176                    ************************************************************
0177                    *
0178 00C0 0201  EEOS    LI   R1,VRAMW        ADDRESS TO WRITE DATA TO VRAM
     00C2 9000
0179 00C4 0202          LI   R2,VDPW         ADDRESS TO WRITE TO VDP
     00C6 9002
0180 00C8 C483          MOV  R3,*R2          SEND LSB OF VRAM ADDRESS TO VDP
0181 00CA 06C3          SWPB R3              REVERSE BYTES
0182 00CC C483          MOV  R3,*R2          SEND MSB OF VRAM ADDRESS TO VDP
0183 00CE 0202          LI   R2,>20          LOAD R2 WITH 'SPACE' CHAR
     00D0 0020
0184 00D2 C442  EES1    MOV  R2,*R1          SEND 'SPACE' TO SCREEN
0185 00D4 0583          INC  R3              INCREMENT CHAR COUNT
0186 00D6 0283          CI   R3,>4700        ARE WE AT THE END OF SCREEN
     00D8 4700
0187 00DA 1AFB          JL   EES1            IF NOT GO AGAIN
0188 00DC 045B          B    *R11            YES, RETURN TO CALLING PROGRAM
0189                    *
0190                    *
0191                    ************************************************************
0192                    *                                                          *
0193                    *                    ERASE LINE SUB                        *
0194                    *                                                          *
0195                    *      REGISTERS USED:                                     *
0196                    *                                                          *
0197                    *      R1 = RESERVED                                       *
0198                    *      R2 = RESERVED                                       *
0199                    *      R3 = STARTING ADDRESS IN LINE IN NAME               *
0200                    *           TABLE TO BE ERASED (USER DEFINED)              *
0201                    ************************************************************
0202                    *
0203                    *
0204 00DE 0201  ERLN    LI   R1,VRAMW        ADDRESS TO WRITE DATA TO VRAM
     00E0 9000
0205 00E2 0202          LI   R2,VDPW         ADDRESS TO WRITE TO VDP
     00E4 9002
0206 00E6 C483          MOV  R3,*R2          SEND LSB OF VRAM ADDRESS TO VDP
0207 00E8 06C3          SWPB R3              REVERSE BYTES
0208 00EA C483          MOV  R3,*R2          SEND MSB OF VRAM ADDRESS TO VDP
0209 00EC 0202          LI   R2,>20          LOAD R2 WITH 'SPACE' CHAR
     00EE 0020
0210 00F0 0203          LI   R3,32           LOAD R3 WITH # OF POSTIONS
```

```
     00F2 0020
0211 00F4 C442  ERL1   MOV  R2,*R1          SEND 'SPACE' CAHR TO NAME TABLE
0212 00F6 0603         DEC  R3              DECREMENT CHAR COUNT
0213 00F8 16FD         JNE  ERL1            IF NOT DONE, GO AGAIN
0214 00FA 045B         B    *R11            DONE, RETURN TO CALLING PROG
0215                *
0216                *
0217                ************************************************************
0218                *                                                          *
0219                *                    DRAW A HORIZ LINE                     *
0220                *                                                          *
0221                *      REGISTERS USED:                                     *
0222                *                                                          *
0223                *      R3 = ADDRESS OF UPPER LEFT CORNER                   *
0224                *      R4 = # OF HORIZ POSITIONS                           *
0225                *      R5 = # OF VERT POSITIONS                            *
0226                *      R9 = PATTERN # OFFSET                               *
0227                *                                                          *
0228                ************************************************************
0229                *
0230                *
0231 00FC 0201  DBOX   LI   R1,>9000        ADDRESS OF DATA TO 9918
     00FE 9000
0232 0100 0202         LI   R2,>9002        ADDRESS OF ADDRESSES TO 9918
     0102 9002
0233 0104 C483         MOV  R3,*R2          SEND LSB OF ADDRESS TO 9918
0234 0106 06C3         SWPB R3              REVERSE BYTES
0235 0108 C483         MOV  R3,*R2          SEND MSB OF ADDRESS TO 9918
0236 010A 06C3         SWPB R3              REVERSE BYTES
0237 010C C189         MOV  R9,R6           GET OFFSET
0238 010E 0226         AI   R6,>01          POINT TO UPPER LEFT CORNER PATTE
     0110 0001
0239 0112 C446         MOV  R6,*R1          SEND IT TO THE 9918
0240 0114 C1C4         MOV  R4,R7           STORE HORIZ COUNT IN TEMP REG
0241 0116 0647         DECT R7              DETERMINE (LENGTH - CORNERS)
0242 0118 1304         JEQ  DBL2            NO LENGTH OTHER THAN CORNERS
0243 011A C189         MOV  R9,R6           PATTERN 00 + OFFSET
0244 011C C446  DBL1   MOV  R6,*R1          SEND LINE SEGMENT TO 9918
0245 011E 0607         DEC  R7              DEC LINE COUNT
0246 0120 16FD         JNE  DBL1            NOT DONE YET, GO AGAIN
0247 0122 C189  DBL2   MOV  R9,R6           GET OFFSET
0248 0124 0226         AI   R6,>05          POINT TO UPPER RIGHT CORNER
     0126 0005
0249 0128 C446         MOV  R6,*R1          SEND IT TO 9918
0250 012A C203         MOV  R3,R8           SEND ADDRESS OF UPPER LEFT TO TE
0251 012C C1C5         MOV  R5,R7           SEND VERT COUNT TO TEMP
0252 012E 0647         DECT R7              DETERMINE (HEIGTH - CORNERS)
0253 0130 130C         JEQ  DBL4            NO HEIGTH OTHER THAN CORNERS
0254 0132 C189         MOV  R9,R6           GET OFFSET
0255 0134 0226         AI   R6,>03          POINT TO VERT LINE PATTERN
     0136 0003
0256 0138 0228  DBL3   AI   R8,>20          INC VERT POSITION BY 1 CHAR
     013A 0020
0257 013C C488         MOV  R8,*R2          SEND ADDRESS TO 9918
0258 013E 06C8         SWPB R8              REVERSE BYTES
0259 0140 C488         MOV  R8,*R2          SEND ADDRESS TO 9918
0260 0142 06C8         SWPB R8              REVERSE BYTES
0261 0144 C046         MOV  R6,R1           SEND VERT LINE SEGMENT TO 9918
0262 0146 0607         DEC  R7              DECREMENT VERT COUNT
0263 0148 16F7         JNE  DBL3            NOT DONE YET, GO AGAIN
```

```
0264 014A 0228  DBL4   AI   R8,>20          INC VERT POSITION BY 1 CHAR
     014C 0020
0265 014E C488         MOV  R8,*R2          SEND ADDRESS TO 9918A
0266 0150 06C8         SWPB R8              REVERSE BYTES
0267 0152 C088         MOV  R8,R2           SEND ADDRESS TO 9918
0268 0154 06C8         SWPB R8              REVERSE BYTES
0269 0156 C189         MOV  R9,R6           GET OFFSET
0270 0158 0226         AI   R6,>02          POINT TO LOWER LEFT CORNER PATT
     015A 0002
0271 015C C446         MOV  R6,*R1          SEND IT TO 9918
0272 015E C1C4         MOV  R4,R7           SEND HORIZ COUNT TO TEMP
0273 0160 0647         DECT R7              DETERMINE (HORIZ - CORNERS)
0274 0162 1304         JEQ  DBL6            NO HORIZ OTHER THAN CORNERS
0275 0164 C189         MOV  R9,R6           GET OFFSET
0276 0166 C446  DBL5   MOV  R6,*R1          SEND HORIZ PATTERN TO 9918
0277 0168 0607         DEC  R7              DECREMENT HORIZ COUNT
0278 016A 16FD         JNE  DBL5            IF NOT DONE, GO AGAIN
0279 016C C189  DBL6   MOV  R9,R6           GET OFFSET
0280 016E 0226         AI   R6,>04          POINT TO LOWER RIGHT PATTERN
     0170 0004
0281 0172 C446         MOV  R6,*R1          SEND PATTERN TO 9918
0282 0174 C1C5         MOV  R5,R7           STORE VERT COUNT IN TEMP
0283 0176 0647         DECT R7              DECREMENT VERT COUNT
0284 0178 1312         JEQ  DBL8            IF NO VERT SEGMENTS, DONE
0285 017A C1C4         MOV  R4,R7           STORE HORIZ COUNT IN TEMP
0286 017C 0607         DEC  R7              DECREMENT HORIZ COUNT
0287 017E C203         MOV  R3,R8           STORE ADDRESS OF UPPER LEFT COR
0288 0180 A207         A    R7,R8           FIND UPPER RIGHT CORNER LOCATION
0289 0182 C189         MOV  R9,R6           GET OFFSET
0290 0184 0226         AI   R6,>03          POINT TO VERTICAL LINE PATTERN
     0186 0003
0291 0188 C1C5         MOV  R5,R7           STORE VERT COUNT IN TEMP
0292 018A 0647         DECT R7              DETERMINE (HEIGHT - CORNERS)
0293 018C 0228  DBL7   AI   R8,>20          INCREMENT VERT POSITION BY 1 CHA
     018E 0020
0294 0190 C488         MOV  R8,*R2          SEND ADDRESS TO 9918
0295 0192 06C8         SWPB R8              REVERSE BYTES
0296 0194 C488         MOV  R8,*R2          SEND ADDRESS TO 9918
0297 0196 06C8         SWPB R8              REVERSE BYTES
0298 0198 C446         MOV  R6,*R1          SEND PATTERN TO 9918
0299 019A 0607         DEC  R7              DECREMENT VERT COUNT
0300 019C 16F7         JNE  DBL7            IF NOT DONE, GO AGAIN
0301 019E 045B  DBL8   B    *R11            SUB DONE RETURN TO CALLING PROG
0302                   *
0303                   *
0304                   ************************************************************
0305                   *                                                          *
0306                   *              LOAD TEXT COLORS SUBROUTINE                 *
0307                   *                                                          *
0308                   *       REGISTERS USED:                                    *
0309                   *                                                          *
0310                   *       R4 = COLORS FOR TEXT CHARACTERS                    *
0311                   *                                                          *
0312                   ************************************************************
0313                   *
0314 01A0 0201  LDTC   LI   R1,>9000
     01A2 9000
0315 01A4 0202         LI   R2,>9002
     01A6 9002
0316 01A8 0203         LI   R3,>4204        ADDRESS FOR TEXT COLORS IN 9918
```

```
      01AA 4204
 0317 01AC C483          MOV  R3,*R2           SEND ADDRESS TO 9918
 0318 01AE 06C3          SWPB R3               REVERSE BYTES
 0319 01B0 C483          MOV  R3,*R2           8 COLOR CHAR X 8 TEXT/CHAR = 64
 0320 01B2 C444  LCLI    MOV  R4,*R1           SEND WORD TO 9918
 0321 01B4 0602          DEC  R2               DECREMENT COUNT
 0322 01B6 16FD          JNE  LCLI             IF NOT DONE, GO AGAIN
 0323 01B8 045B          B    *R11             DONE, RETURN TO CALLING PROG
 0324              *
 0325              *
 0326              ************************************************************
 0327              *                                                          *
 0328              *                 DRAW A VERTICAL LINE SUB                 *
 0329              *                                                          *
 0330              *      REGISTERS USED:                                     *
 0331              *                                                          *
 0332              *      R3 = ADDRESS ON SCREEN                              *
 0333              *      R4 = # OF POSITIONS                                 *
 0334              *      R9 = PATTERN OFFSET                                 *
 0335              *                                                          *
 0336              ************************************************************
 0337              *
 0338 01BA C483  DVLN    MOV  R3,*R2           SEND ADDRESS TO 9918
 0339 01BC 06C3          SWPB R3               REVERSE BYTES
 0340 01BE C483          MOV  R3,*R2           SEND ADDRESS TO 9918
 0341 01C0 06C3          SWPB R3               REVERSE BYTES
 0342 01C2 C445          MOV  R5,*R1           SEND PATTERN TO 9918
 0343                    END
NO ERRORS,     NO WARNINGS
```

5. TMS9918A/9928A/9929A ELECTRICAL SPECIFICATIONS

5.1 ABSOLUTE MAXIMUM RATINGS OVER OPERATING FREE-AIR TEMPERATURE RANGE (unless otherwise noted)*

Supply voltage, V_{CC} ..... −0.3 to 20 V
All input voltages ..... −0.3 to 20 V
Output voltage ..... −2 to 7 V
Continuous power dissipation ..... 1.3 W
Operating free-air temperature range ..... 0°C to 70°C
Storage temperature range ..... −55°C to +150°C

*Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the Recommended Operating Conditions section of this specification is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

5.2 RECOMMENDED OPERATING CONDITIONS*

PARAMETER		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.75		5.25	V
Supply voltage, V_{SS}			0		V
Input Voltage, V_I, $\overline{\text{RESET}}$/SYNC pin	SYNC active	10		12	V
	RESET active			0.6	V
	SYNC and RESET inactive	3		6	V
High-level input, V_{IH}	XTAL1, XTAL2	2.75			V
	All other inputs	2.2			V
Input voltage, V_I, EXT VDP pin (TMS9918A only)	SYNC level		2.6		V
	White level		3.7		V
	Black level		3		V
Low-level input voltage, V_{IL}				0.8	V
Operating free-air temperature, T_A		0		70	°C

* All voltage values are with respect to V_{SS}.

5.3 ELECTRICAL CHARACTERISTICS OVER FULL RANGES OF RECOMMENDED OPERATING CONDITIONS (unless otherwise noted)

TMS9918A/9928A/9929A

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	$\overline{RAS}$, $\overline{CAS}$, R/$\overline{W}$	I_{OH} = 400 μA	2.7	3.4		V
		All other outputs		2.4	3.2		
V_{OL}	Low-level output voltage	CPU data	I_{OL} = 1.2 mA		0.3	0.6	V
		DRAM interface	I_{OL} = 800 μA			0.6	
I_{OZH}	Off-state output current high-level voltage applied, D0-D7 outputs		V_O = 5.25 V		1	100	μA
I_{OZL}	Off-state output current high-level voltage applied, D0-D7 outputs		V_O = 0.4 V		1	-100	μA
I_{IH}	High-level input current		V_I = 5.25 V, all other pins at 0 V			10	μA
I_{IL}	Low-level input current		V_I = 0 V, All other pins at 0 V			-10	μA

TMS9918A Only (Figure 5-1)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
V_{white}	Video voltage level of white, COMVID	R_L = 470 Ω	2.8	3.0	3.2 V	
V_{black}	Video voltage level of black (blank), COMVID		2.1	2.3	2.5 V	
V_{sync}	Video voltage level of sync, COMVID		1.85	2.0	2.1 V	

† All typical values are at V_{CC} = 5.25 V, T_A = 25°C.

5.3 ELECTRICAL CHARACTERISTICS OVER FULL RANGES OF RECOMMENDED OPERATING CONDITIONS (unless otherwise noted) (Continued)

TMS9928A/9929A Only (Figure 5-1)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
V_{white}	Video voltage level of white, Y, R-Y, B-Y outputs	R_L = 470 Ω	2.5	3	3.6	V
V_{black}	Video voltage level of black (blank), Y, R-Y, B-Y outputs		1.6	2.3	2.5	V
V_{sync}	Video voltage level of sync, Y output		1.2	1.8	2	V

TMS9929A Only

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{PS}	Color burst video voltage level with respect to V no color	R-Y output		0.25		V
V_{neg}	Color burst video voltage level with respect to V no color	B-Y output		-0.25		V

TMS9918A/9928A/9929A (Figure 5-2)

PARAMETER			TEST CONDITIONS	MIN	NOM	MAX	UNIT
	Video voltage difference, white-black, Y, R-Y, B-Y outputs			0.7	1.0		V
I_{CC}	Average supply current from V_{CC}		T_A = 25°C		200	250	mA
C_i	Input capacitance	D0-D7	unmeasured f = 11 MHz, pins at 0 V			20	pF
		All other inputs				10	
						10	
C_o	Output capacitance		unmeasured f = 11 MHz, pins at 0 V			20	pF

† All typical values are at V_{CC} = 5.25 V, T_A = 25°C.

5.4 TIMING REQUIREMENTS OVER FULL RANGES OF RECOMMENDED OPERATING CONDITIONS (TMS9918A/9928A/9929A)

CPU - VDP Interface (Figures 5-3 and 5-4)

PARAMETER		MIN	NOM	MAX	UNIT
$t_{su(A-RL)}$	Address setup time before CSR low		0		ns
$t_{su(A-WL)}$	Address setup time before CSW low		30		ns
$t_{h(WL-A)}$	Address hold time after CSW low		30		ns
$t_{su(D-WH)}$	Data setup time before CSW high		100		ns
$t_{h(WH-D)}$	Data hold time after CSW high		30		ns
$t_{w(WL)}$	Pulse width, CSW low		200		ns
$t_{w(CS-H1)}$	Pulse width, chip select high (requesting memory access)		8		μs
$t_{w(CS-H2)}$	Pulse width, chip select high (not requesting memory access)		2		μs

VDP-VRAM Interface (Figure 5-5 and 5-6)

PARAMETER		MIN	NOM	MAX	UNIT
t_c	Memory read or write cycle time	372			ns
$t_{su(D-CH)}$	Input data setup time before CAS high	60			ns
$t_{h(CH-D)}$	Input data hold time after CAS high	0			ns

External Clock Source (Figure 5-7)

PARAMETER		MIN	TYP	MAX	UNIT
f_{ext}	External source frequency	10.738098	10.738635	10.739172	MHz
t_r/t_f	External source rise/fall time		10	15	ns
t_{wH}	External source high-level pulse width	42	47	52	ns
t_{wL}	External source low-level pulse width	42	47	52	ns
t_{pD}	External source phase delay from XTAL1 falling edge to XTAL2 falling edge	42	47	52	ns

5.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (TMS9918A/9928A/9929A)

CPU-VDP Interface

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$T_{A(CSR)}$	Data access time from CSR low	C_L = 300 pF		100	150	ns
t_{PVX}	Data disable time after CSR high			65	100	ns
$t_{PVX,A}$	Data invalid time from address changes			0		ns
f_{CPUCLK}	CPU clock output clock frequency (f_{ext} ÷ 3)		3.4	3.58	3.76	MHz
$f_{GROMCLK}$	GROM clock output clock frequency (f_{ext} ÷ 24)		425.12	447.5	469.88	kHz

VDP-VRAM Interface (Figures 5-5 and 5-6)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_W	Pulse width, $\overline{CAS}$ high	C_L = 50 pF	80	100	120	ns
$t_{W(CL)}$	Pulse width, $\overline{CAS}$ low		220	230	250	ns
$t_{W(RH)}$	Pulse width, $\overline{RAS}$ high		100	125	150	ns
$t_{W(RL)}$	Pulse width, $\overline{RAS}$ low		190	210	230	ns
$t_{W(W)}$	Pulse width, write pulse		170	190	210	ns
t_{CA-CL}	Delay time, column address to $\overline{CAS}$ low		-10	-2		ns
t_{RA-RL}	Delay time, row address to $\overline{RAS}$ low		25	45	65	ns
t_{d-WL}	Delay time, data to R/$\overline{W}$ low		0	6	20	ns
t_{WH-CL}	Delay time, R/$\overline{W}$ high to $\overline{CAS}$ low		25	50	75	ns
t_{W-CH}	Delay time, R/$\overline{W}$ low to $\overline{CAS}$ high		120	140	160	ns
t_{W-RH}	Delay time, R/$\overline{W}$ low to $\overline{RAS}$ high		60	75	90	ns

5.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (TMS9918A/9928A/9929A) (Continued)

TMS9918A Composite video output (Figures 5-8 and 5-9)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{CL-CA}	Column address valid after $\overline{CAS}$ low	C_L = 50 pF	45	65	85	ns
t_{RL-RA}	Row address valid after $\overline{RAS}$ low		20	25	30	ns
t_{RL-CA}	Column address valid after $\overline{RAS}$ low		95	110	130	ns
t_{CL-D}	Data valid after $\overline{CAS}$ low		240	260	280	ns
t_{RL-D}	Data valid after $\overline{RAS}$ low		95	110	125	ns
t_{WL-D}	Data valid after R/$\overline{W}$ low		135	165	195	ns
t_{CH-WL}	Read command valid after $\overline{CAS}$ high		0			ns
t_{CL-W}	Write command valid after $\overline{CAS}$ low		270	290	310	ns
t_{CH-RL}	Delay time, $\overline{CAS}$ high to $\overline{RAS}$ low		45	65		ns
t_{CL-RH}	Delay time, $\overline{CAS}$ low to $\overline{RAS}$ high		150	170	190	ns
t_{RL-CL}	Delay time, $\overline{RAS}$ low to $\overline{CAS}$ low		30	40	50	ns

5.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (TMS9918A/9928A/9929A) (Continued)

TMS9918A Composite video output (Figures 5-8 and 5-9)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{fl}	Fall time, V_{black} to V_{sync}	R_L = 470 Ω C_L = 150 pF		10		ns
$t_{w(HS)}$	Pulse width, horizontal sync			4.84		μs
t_{rl}	Rise time, V_{sync} to V_{black}			20		ns
t_{HS-CD}	Delay time, sync to color burst			372		ns
$t_{w(CB)}$	Width, color burst			261		μs
t_{CB-LB}	Delay time, color burst to left border			1.49		μs
t_{r2}	Rise time, V_{black} to V_{white}			60		ns
$t_{w(LB)}$	Left border video width			2.42		μs
t_{f2}	Fall time, V_{white} to V_{black}			110		ns
$t_{w(AD)}$	Width of active display area			47.68		μs
$t_{w(RB)}$	Right border video width			2.79		μs
t_{RB-HS}	Delay time, right border to horizontal sync			1.49		μs μs
t_{VFB}	Vertical front blanking			191.1		μs
t_{VS}	Vertical sync			191.1		μs
V_{VBB}	Vertical back blanking			828		μs
t_{ABA}	Active plus border area time			18.8		ms

NOTE: Fall times depend on external pull-down resistor

5.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (TMS9918A/9928A/9929A) (Continued)

TMS9928A/9929A Y, R-Y, B-Y outputs (Figures 5-10 through 5-13)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{f3}	Fall time, V_{black} to V_{sync}	R_L = 470 Ω C_L = 15 pF		100		ns
$t_{w(HSI)}$	Pulse width, horizontal sync			4.84		µs
t_{r3}	Rise time, V_{sync} to V_{black}			150		ns
$t_{w(BP)}$	Width, back porch			4.47		µs
$t_{w(LBI)}$	Width, left border			2.8		µs
$t_{w(P)}$	Pulse width, pixel			186.24		ns
$t_{w(horz)}$	Width, horizontal line			63.695		µs
$t_{w(ADI)}$	Width, active display area			47.67		µs
t_{r4}	Rise time, V_{black} to V_{white}			75		ns
t_{r4}	Fall time, V_{white} to V_{black}			50		ns
$t_{w(RBI)}$	Width, right border			2.42		µs
$t_{w(FP)}$	Width, front porch			1.49		µs
t_{r5}	Rise time, V no color to V pos CB			150		ns
$t_{w(CB1)}$	Pulse width, pos color burst			2.6		µs
t_{f5}	Fall time, V pos CB to V no color			100		ns
$t_{w(CB-LBI)}$	Delay time, pos CB to left border			1.49		µs
t_{f6}	Fall time, V no color to V neg CB			100		ns
t_{r6}	Rise time, V neg CB to V no color			150		ns
$t_{w(VSI)}$	Pulse width, vertical sync			465		ns
t_{VFBI}	Vertical front blanking			191.09		µs
t_{VSI}	Vertical sync			191.09		µs
t_{VBBI}	Vertical back blanking			828.04		µs
t_{ABAI}	Active area plus border area total			18.70		mS
	Vertical time			19.91		mS

NOTE: Fall times depend on external pull-down resistor.

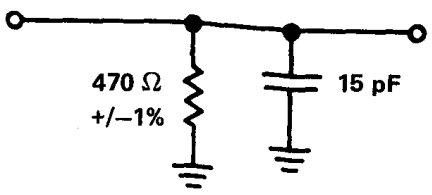


FIGURE 5-1 – LOAD CIRCUIT FOR COMVID (ALL DEVICES) AND R-Y, Y, B-Y SWITCHING CHARACTERISTICS (TMS9928A/9929A)

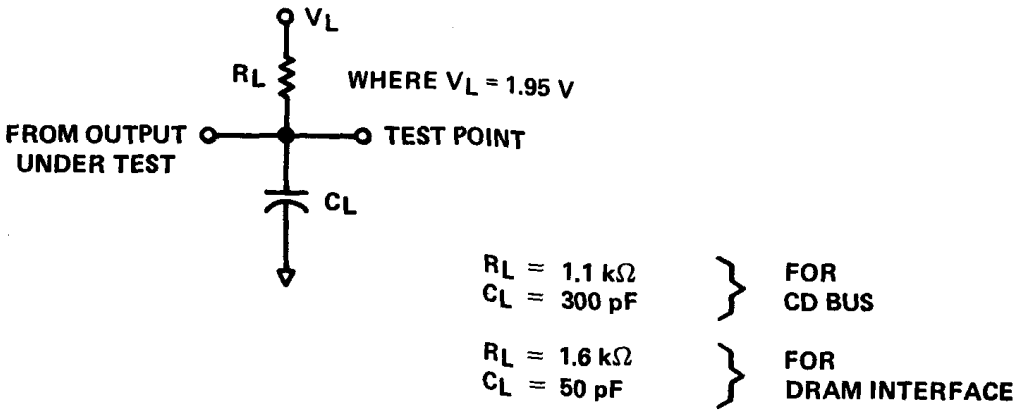


FIGURE 5-2 – LOAD CIRCUITS FOR ALL OUTPUTS EXCEPT COMVID, R-Y, Y, B-Y

WRITE CYCLE

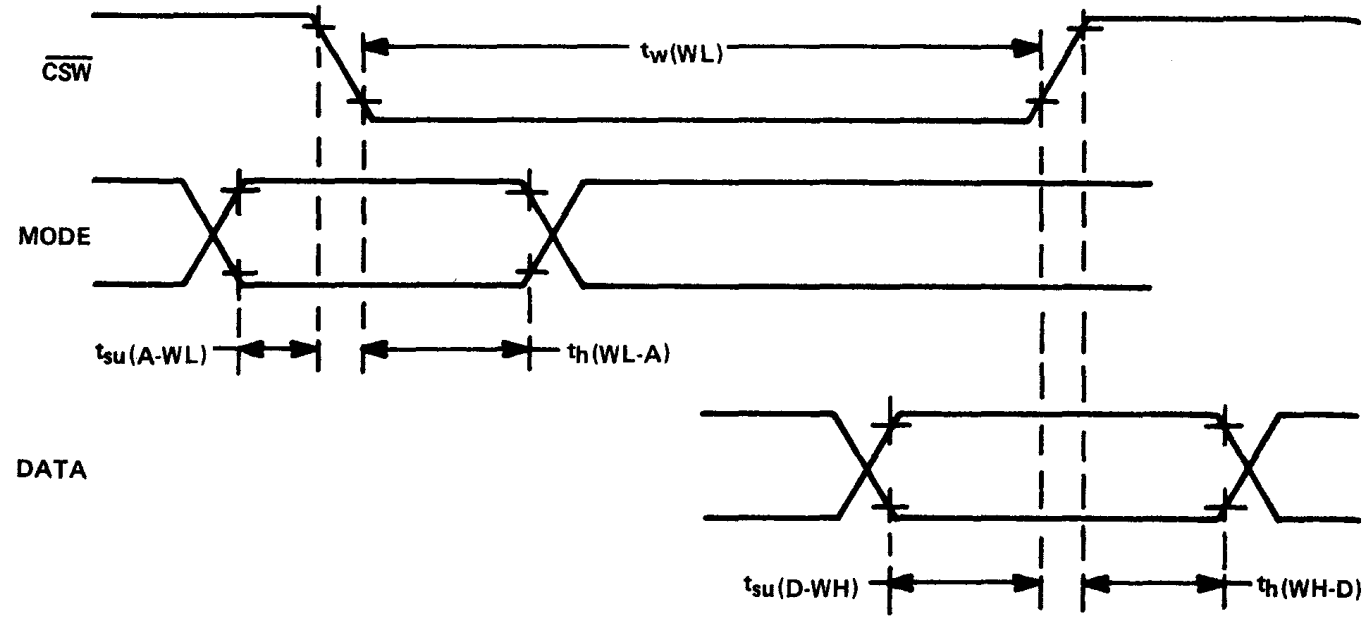
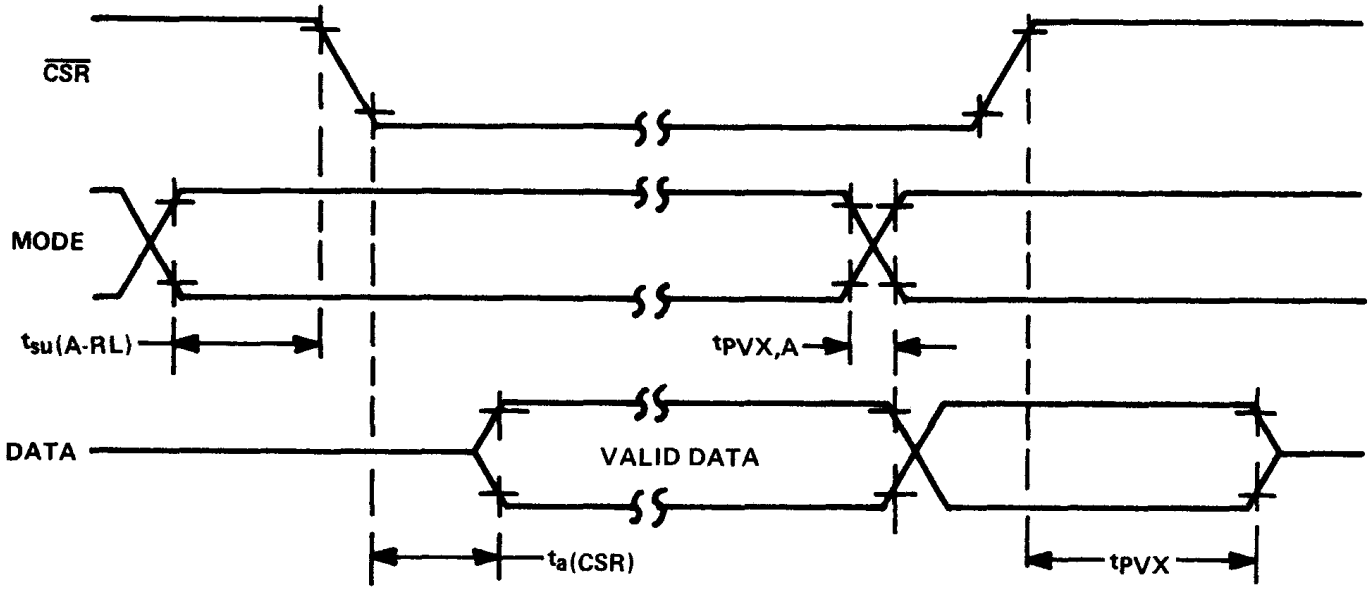


FIGURE 5-3 — CPU-VDP WRITE CYCLE FOR TMS9918A/9928A/9929A

READ CYCLE



NOTE: All measurements are made at 10% and 90% points.

FIGURE 5-4 — CPU-VDP READ CYCLE FOR TMS9918A/9928A/9929A

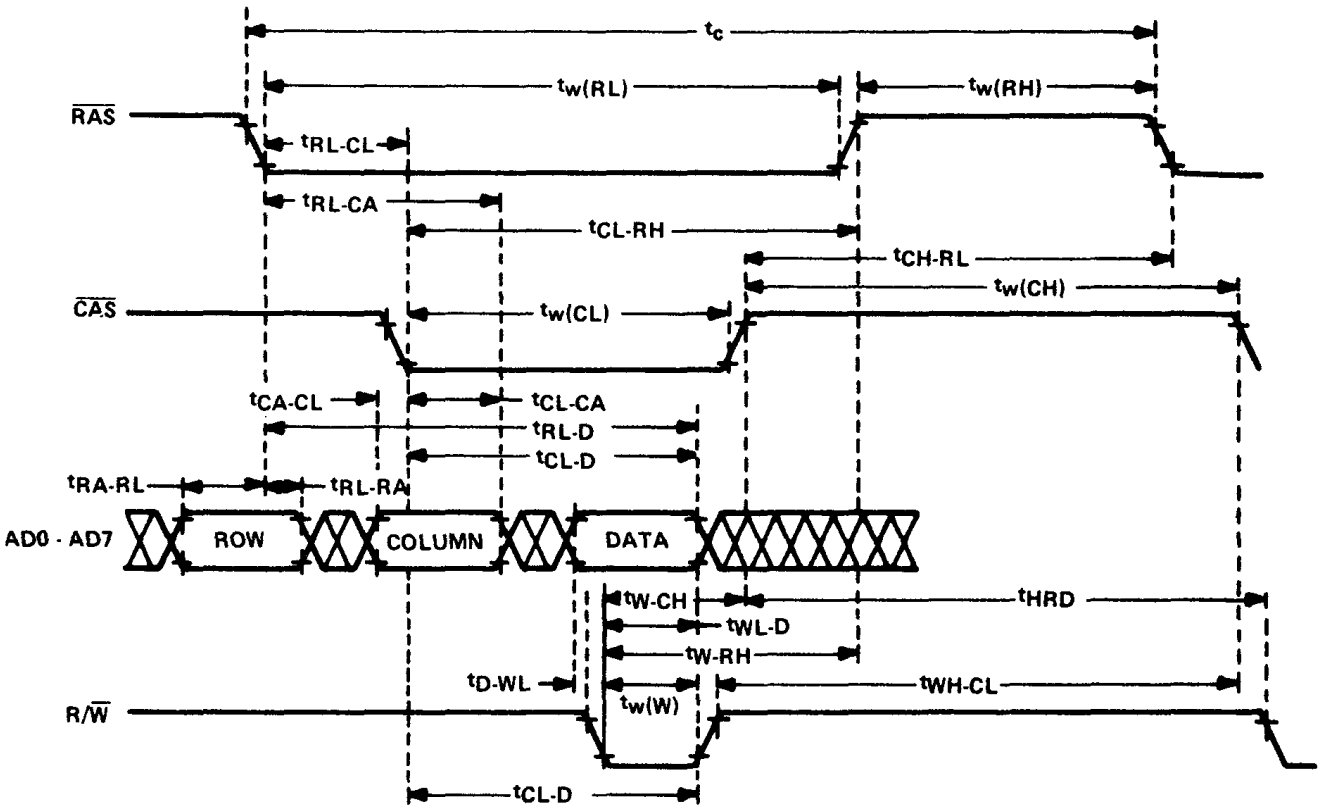
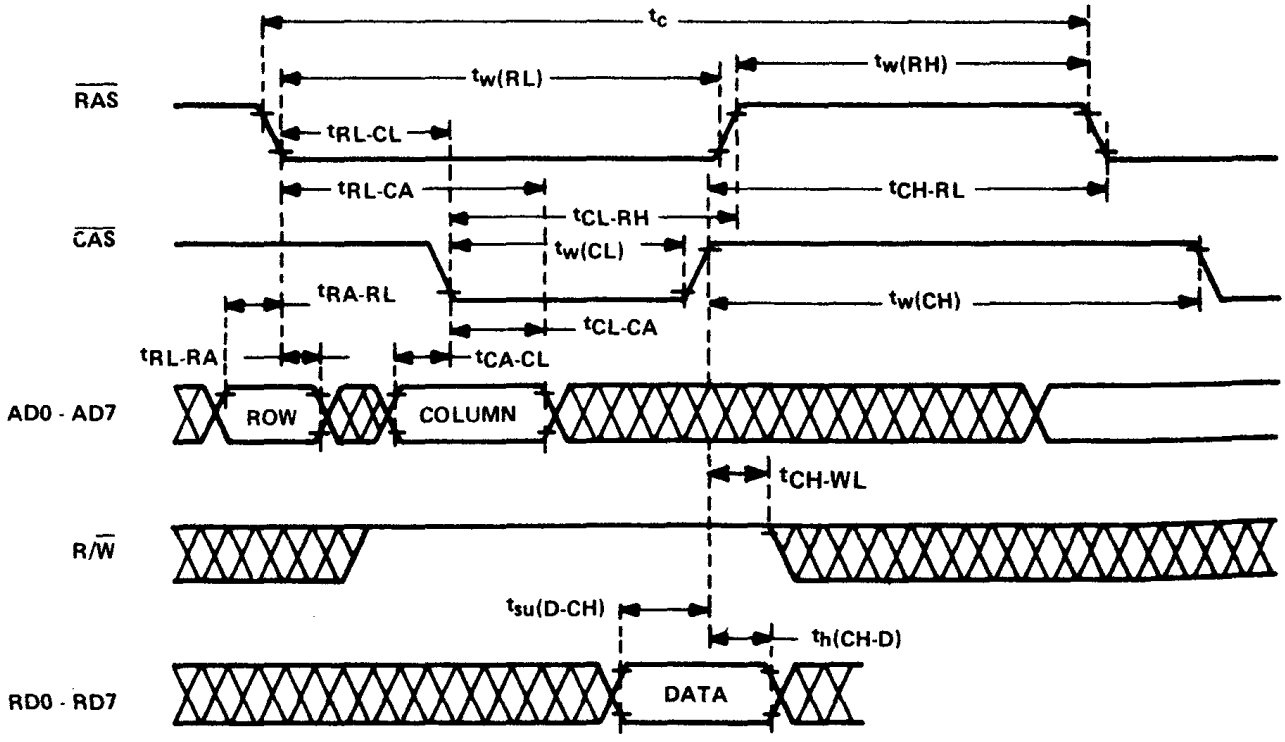
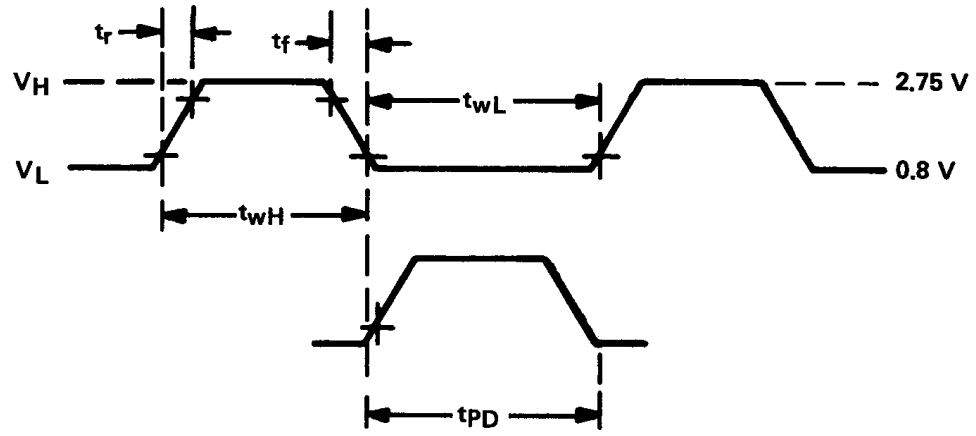


FIGURE 5-5 – VRAM WRITE CYCLE



NOTE: All measurements are made at 10% and 90% points.

FIGURE 5-6 – VRAM READ CYCLE



NOTE: All measurements are made at 10% and 90% points.

FIGURE 5-7 – EXTERNAL CLOCK TIMING WAVEFORM

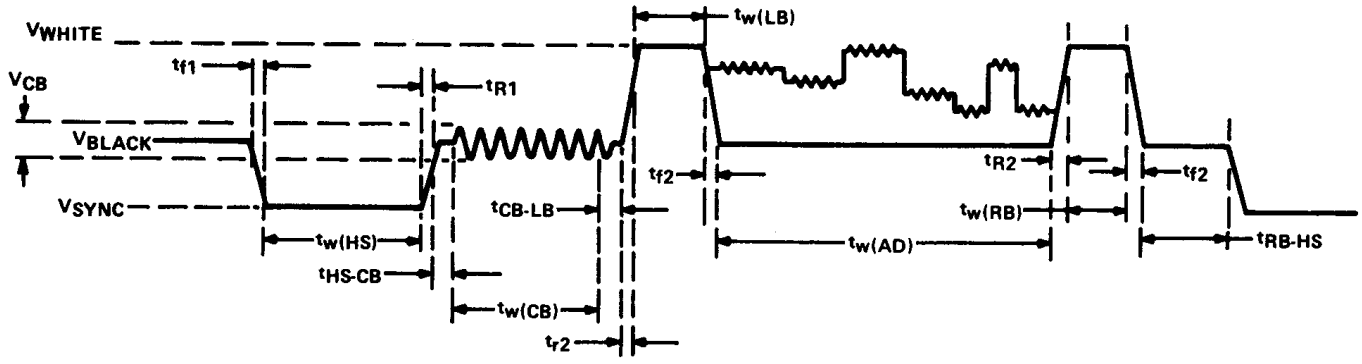
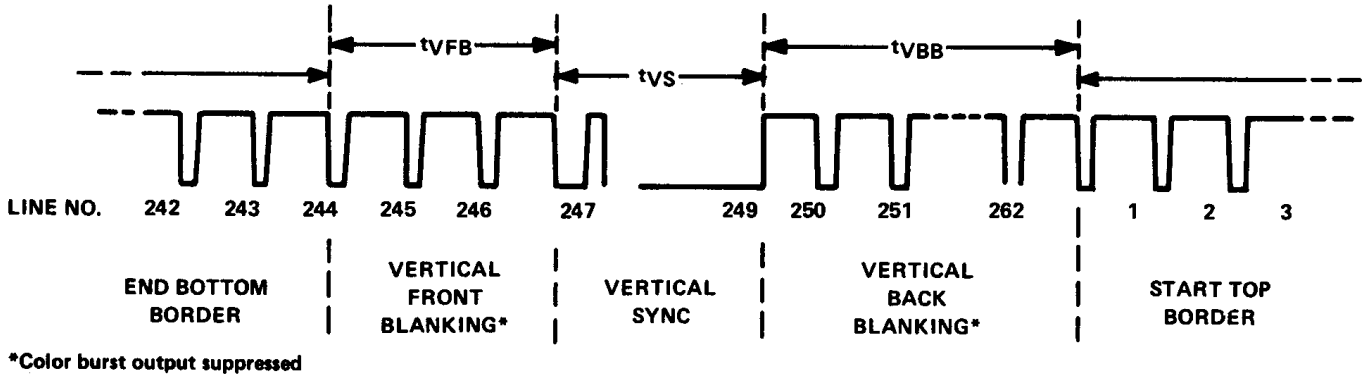


FIGURE 5-8 – TMS9918A COMVID HORIZONTAL TIMING



*Color burst output suppressed

FIGURE 5-9 – TMS9918A VERTICAL TIMING

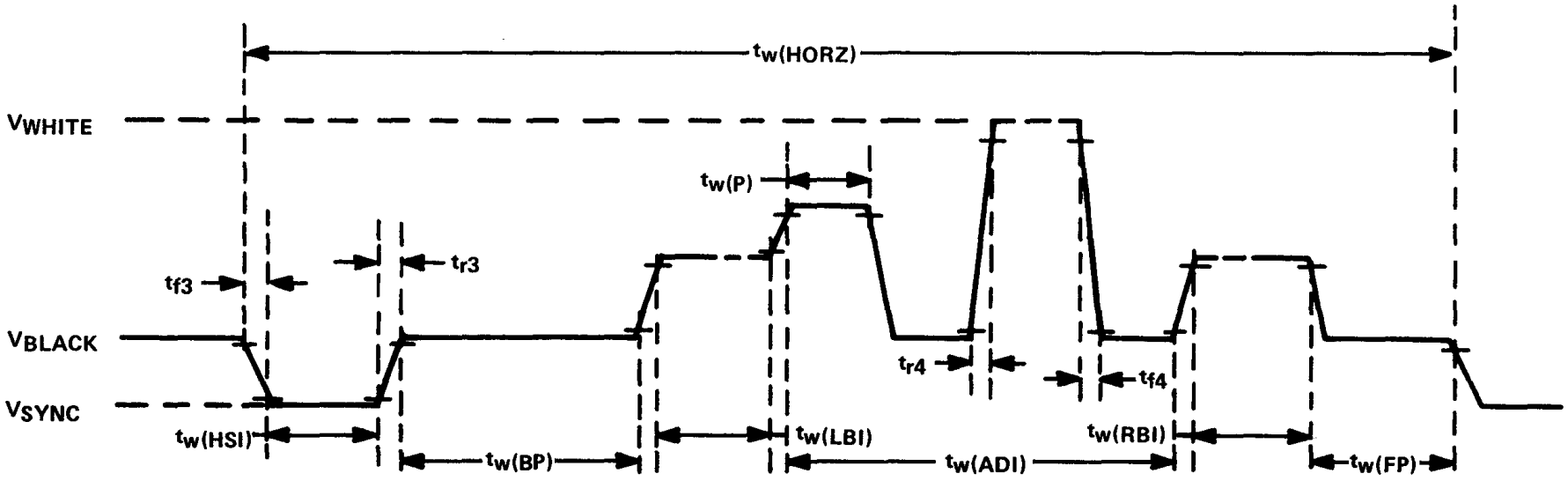
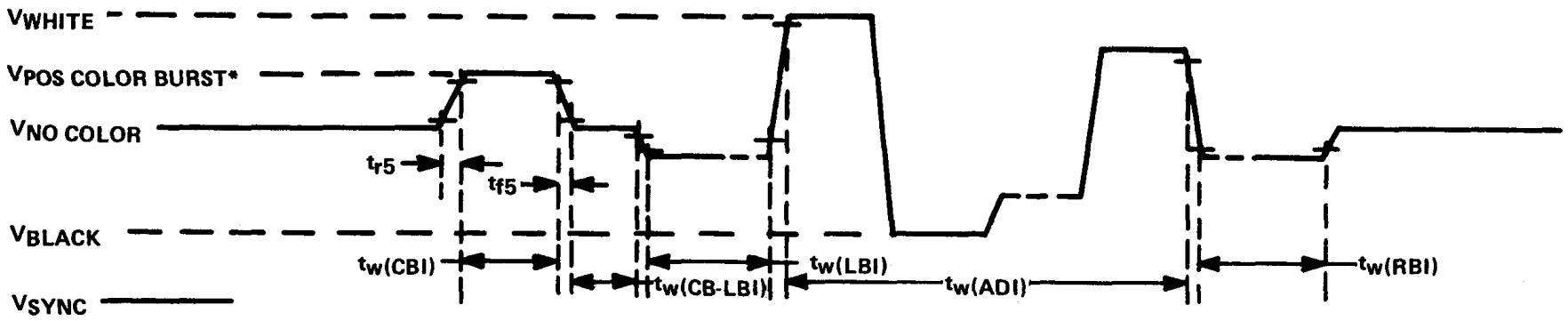
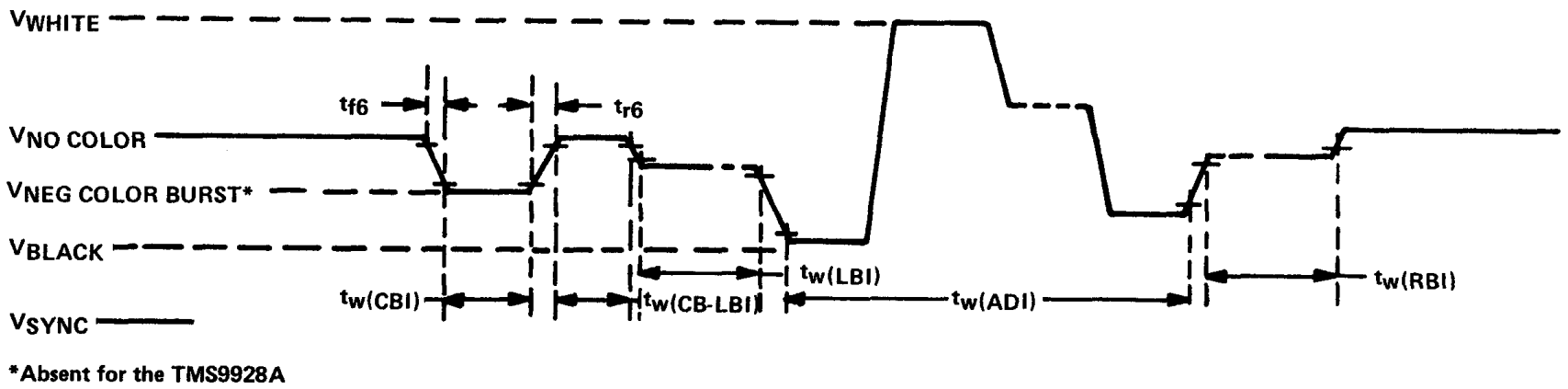


FIGURE 5-10 — TMS9928A/9929A Y HORIZONTAL TIMING



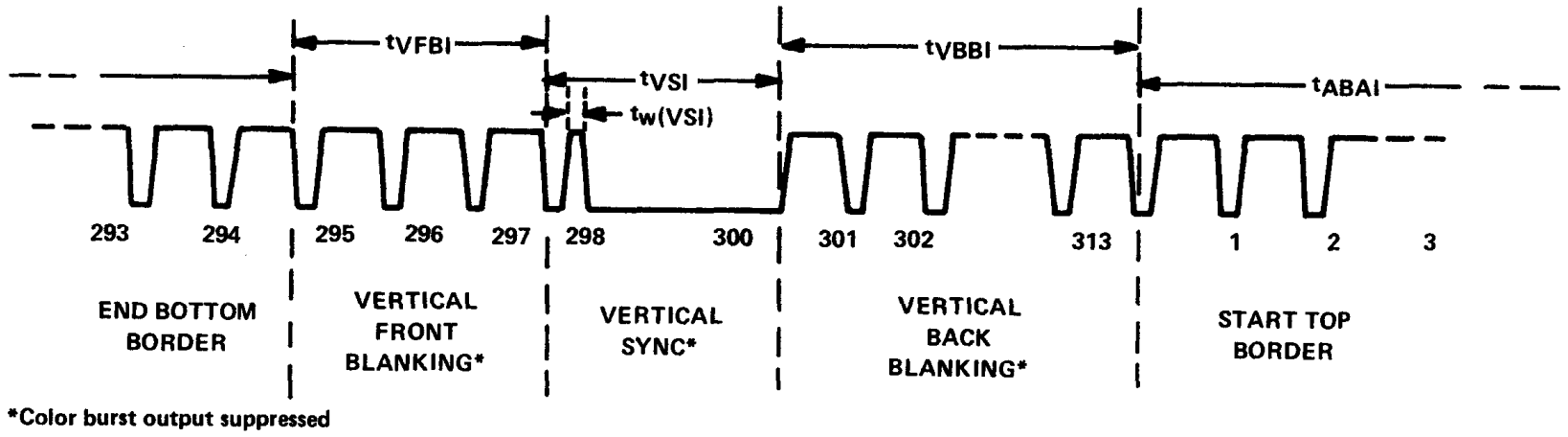
*Absent for the TMS9928A

FIGURE 5-11 — TMS9928A/9929A R-Y HORIZONTAL TIMING



*Absent for the TMS9928A

FIGURE 5-12 — TMS9928A/9929A B-Y HORIZONTAL TIMING

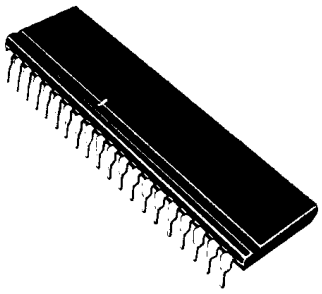


*Color burst output suppressed

FIGURE 5-13 — TMS9929A VERTICAL TIMING

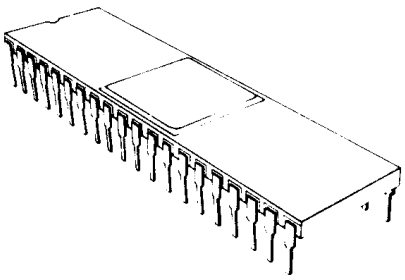
6. MECHANICAL DATA

6.1 TMS9918 40-PIN PLASTIC DUAL-IN-LINE PACKAGE



NOTES: a. All linear dimensions are in inches and parenthetically in millimeters. Inch dimensions goven.
b. Each pin centerline is located within 0.010 (0.25) of its true longitudinal position.

6.2 TMS9918 40-PIN CERAMIC DUAL-IN-LINE PACKAGE



NOTES: a. All linear dimensions are in inches and parenthetically in millimeters. Inch dimensions goven.
b. Each pin centerline is located within 0.010 (0.25) of its true longitudinal position.

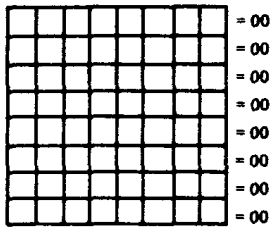
APPENDIX A

ASCII CHARACTER SET

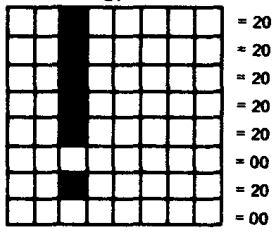
Software programs apply to all three VDPs (TMS9918A/9928A/9929A).

This appendix contains the diagrams and software listing of an upper and lower case ASCII character set. The character matrix is 5 X 7 in the 8 X 8 pixel block. These characters are left-justified so they can be used in the text (6 X 8 pixels) mode.

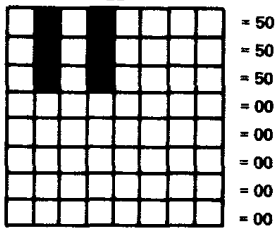
PATTERN 20



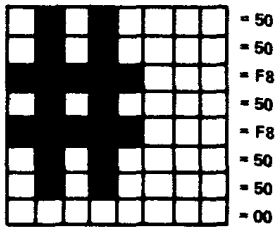
PATTERN 21



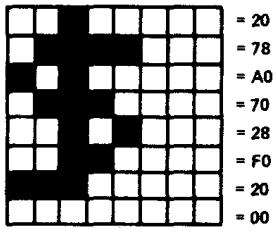
PATTERN 22



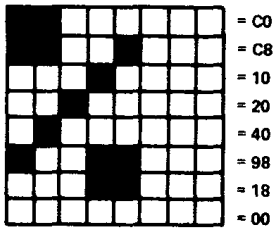
PATTERN 23



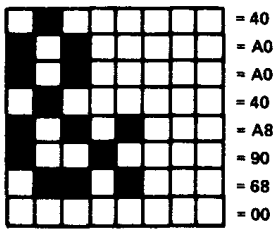
PATTERN 24



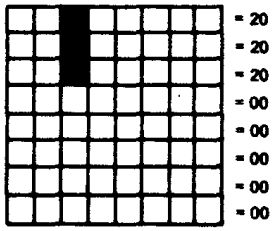
PATTERN 25



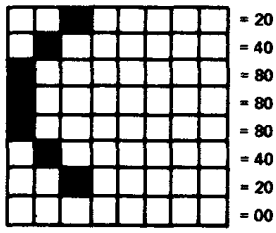
PATTERN 26



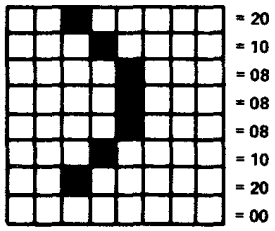
PATTERN 27



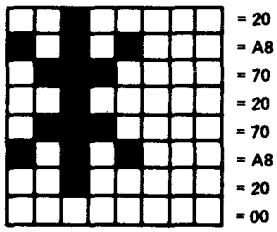
PATTERN 28



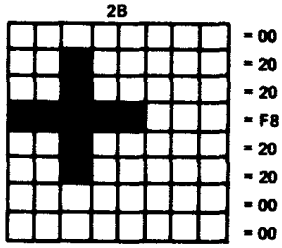
PATTERN 29



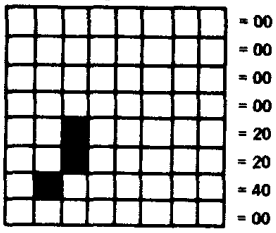
PATTERN 2A



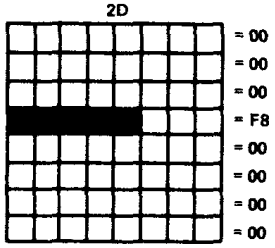
PATTERN 2B



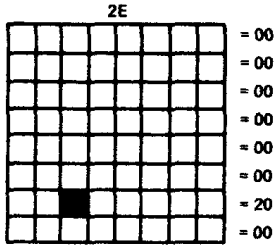
PATTERN 2C



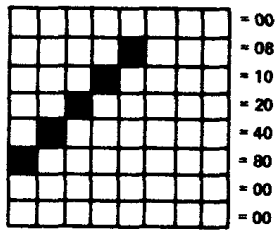
PATTERN 2D



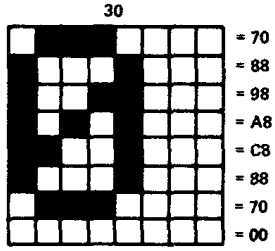
PATTERN 2E



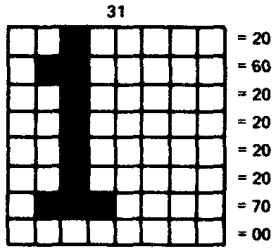
PATTERN 2F



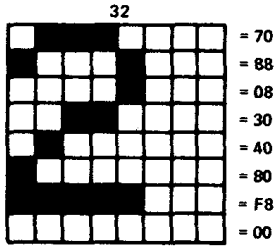
PATTERN 30



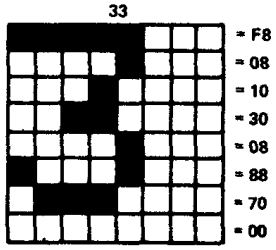
PATTERN 31



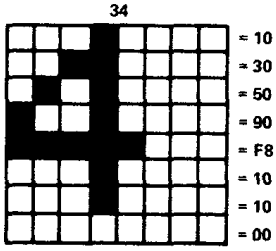
PATTERN 32



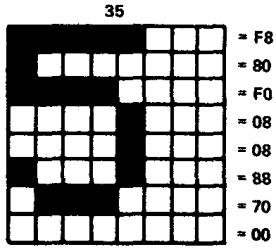
PATTERN 33



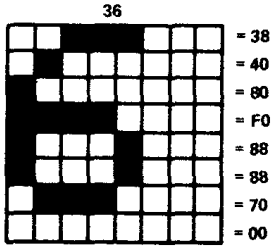
PATTERN 34



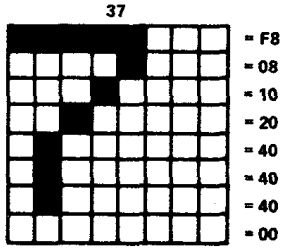
PATTERN 35



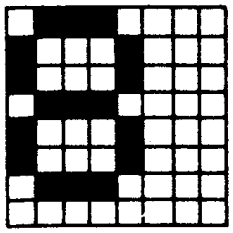
PATTERN 36



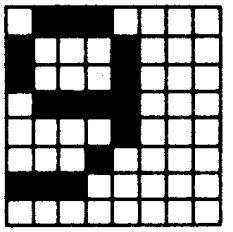
PATTERN 37



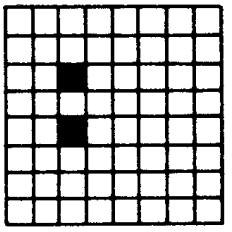
PATTERN 38



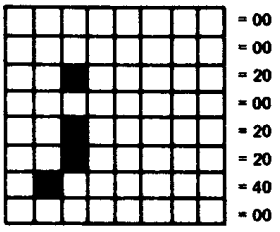
PATTERN 39



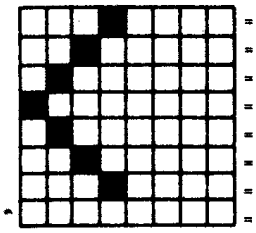
PATTERN 3A



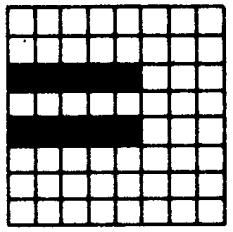
PATTERN 3B



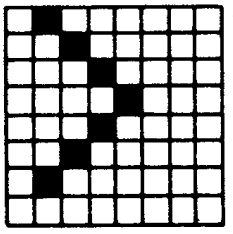
PATTERN 3C



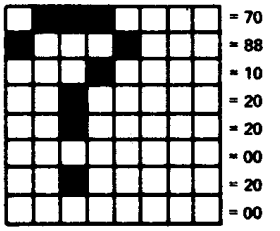
PATTERN 3D



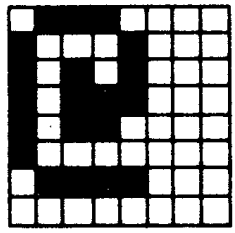
PATTERN 3E



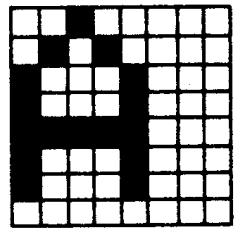
PATTERN 3F



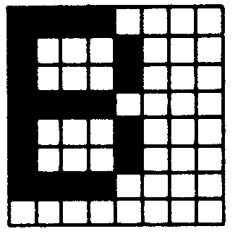
PATTERN 40



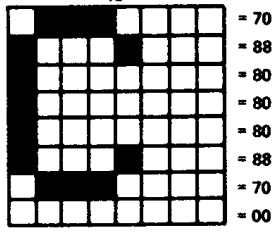
PATTERN 41



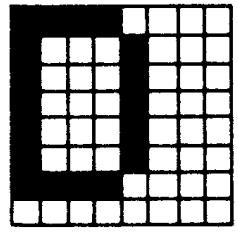
PATTERN 42



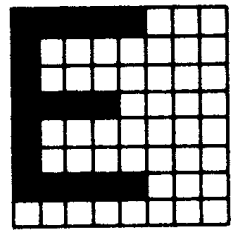
PATTERN 43



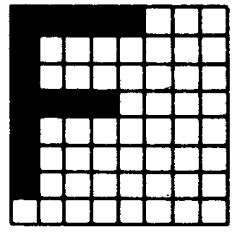
PATTERN 44



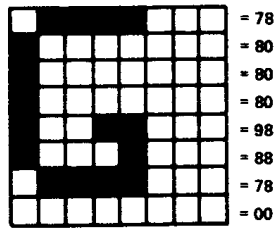
PATTERN 45



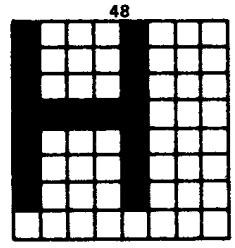
PATTERN 46



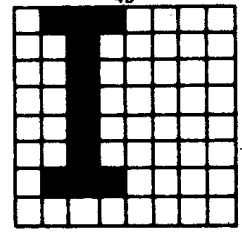
PATTERN 47



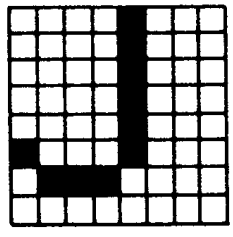
PATTERN 48



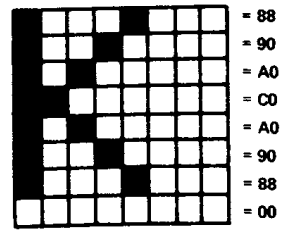
PATTERN 49



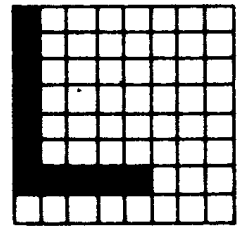
PATTERN 4A



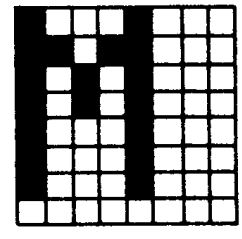
PATTERN 4B



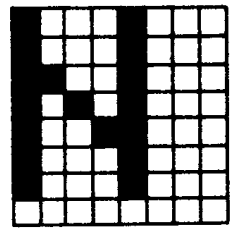
PATTERN 4C



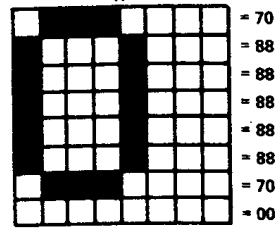
PATTERN 4D



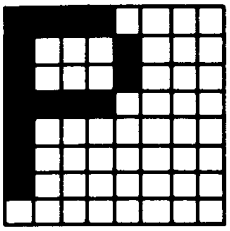
PATTERN 4E



PATTERN 4F

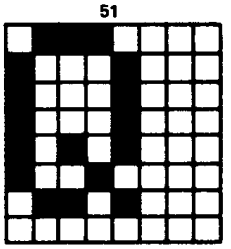


PATTERN 50



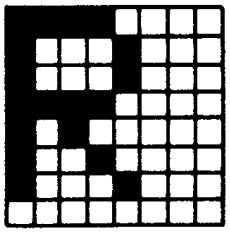
= F0
= 88
= 88
= F0
= 80
= 80
= 80
= 00

PATTERN 51



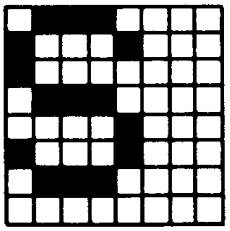
= 70
= 88
= 88
= 88
= A8
= 90
= 68
= 00

PATTERN 52



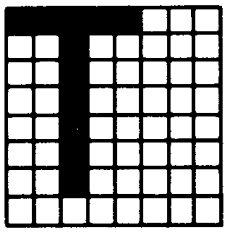
= F0
= 88
= 88
= F0
= A0
= 90
= 88
= 00

PATTERN 53



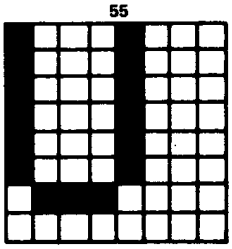
= 70
= 88
= 80
= 70
= 08
= 88
= 70
= 00

PATTERN 54



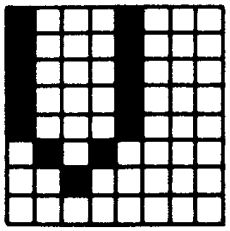
= F8
= 20
= 20
= 20
= 20
= 20
= 20
= 00

PATTERN 55



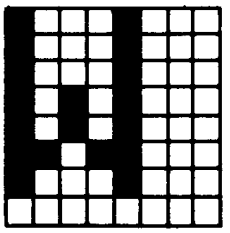
= 88
= 88
= 88
= 88
= 88
= 88
= 70
= 00

PATTERN 56



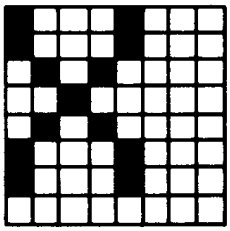
= 88
= 88
= 88
= 88
= 88
= 50
= 20
= 00

PATTERN 57



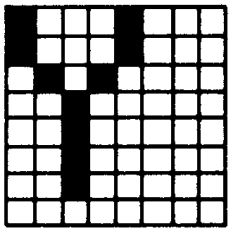
= 88
= 88
= 88
= A8
= A8
= D8
= 88
= 00

PATTERN 58



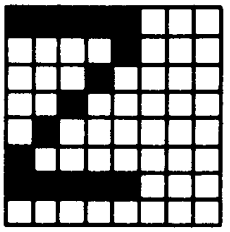
= 88
= 88
= 50
= 20
= 50
= 88
= 88
= 00

PATTERN 59



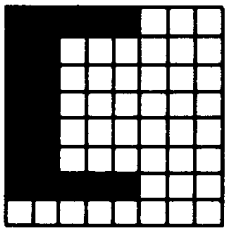
= 88
= 88
= 50
= 20
= 20
= 20
= 20
= 00

PATTERN 5A



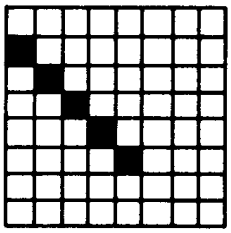
= F8
= 08
= 10
= 20
= 40
= 80
= F8
= 00

PATTERN 5B



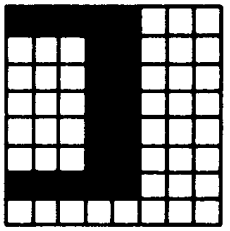
= F8
= C0
= C0
= C0
= C0
= C0
= F8
= 00

PATTERN 5C



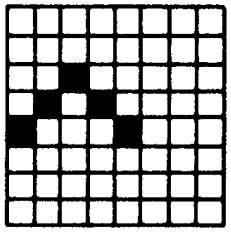
= 00
= 80
= 40
= 20
= 10
= 08
= 00
= 00

PATTERN 5D



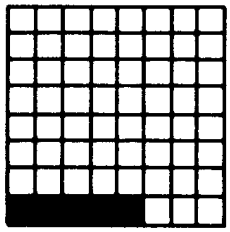
= F8
= 18
= 18
= 18
= 18
= 18
= F8
= 00

PATTERN 5E



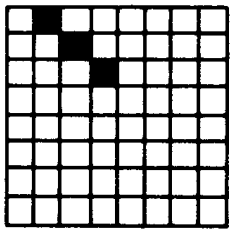
= 00
= 00
= 20
= 50
= 88
= 00
= 00
= 00

PATTERN 5F



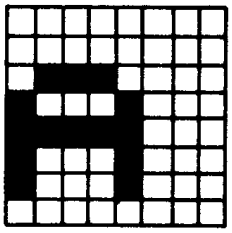
= 00
= 00
= 00
= 00
= 00
= 00
= 00
= F8

PATTERN 60



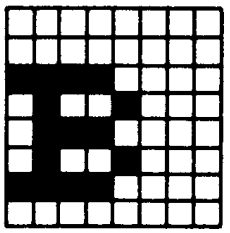
= 40
= 20
= 10
= 00
= 00
= 00
= 00
= 00

PATTERN 61



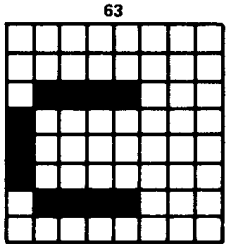
= 00
= 00
= 70
= 88
= F8
= 88
= 88
= 00

PATTERN 62



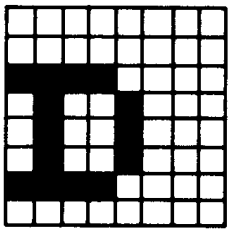
= 00
= 00
= F0
= 48
= 70
= 48
= F0
= 00

PATTERN 63

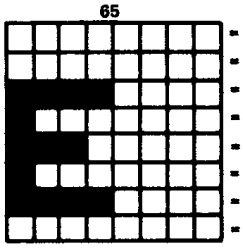


= 00
= 00
= 78
= 80
= 80
= 80
= 78
= 00

PATTERN 64

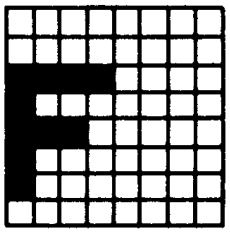


PATTERN 65

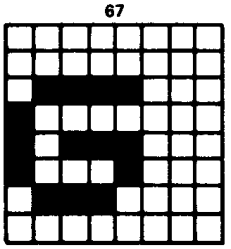


= 00
= 00
= F0
= 80
= E0
= 80
= F0
= 00

PATTERN 66

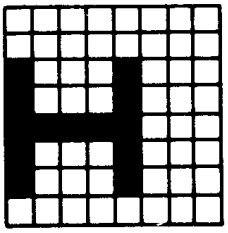


PATTERN 67

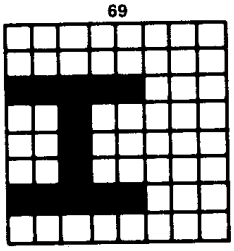


= 00
= 00
= 78
= 80
= B8
= 88
= 70
= 00

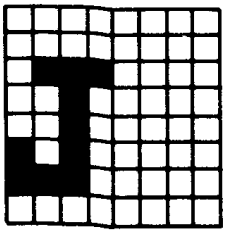
PATTERN 68



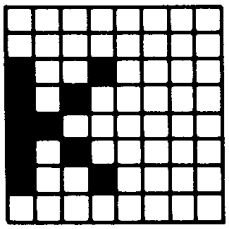
PATTERN 69



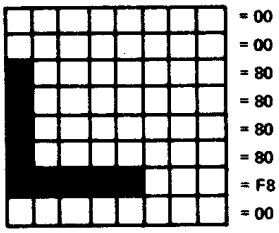
PATTERN 6A



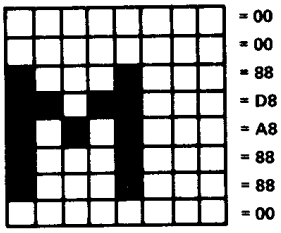
PATTERN 6B



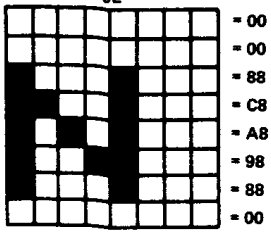
PATTERN 6C



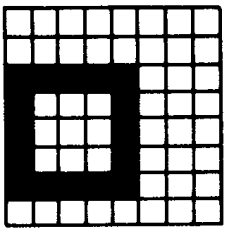
PATTERN 6D



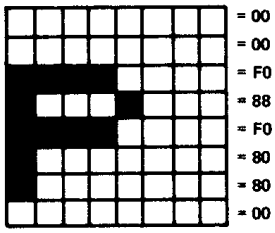
PATTERN 6E



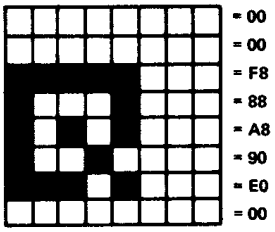
PATTERN 6F



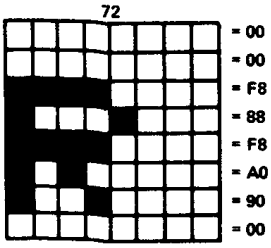
PATTERN 70



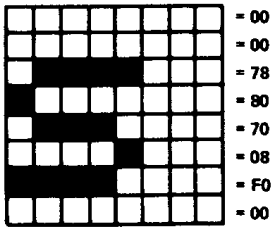
PATTERN 71



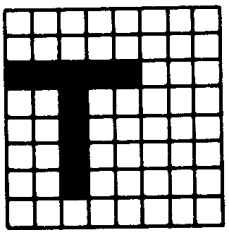
PATTERN 72



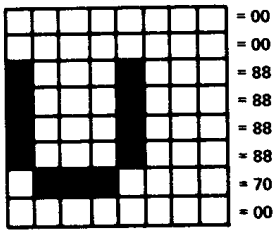
PATTERN 73



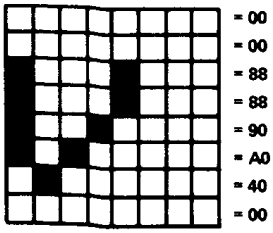
PATTERN 74



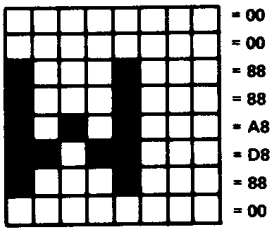
PATTERN 75



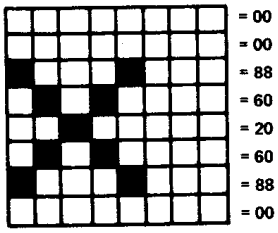
PATTERN 76



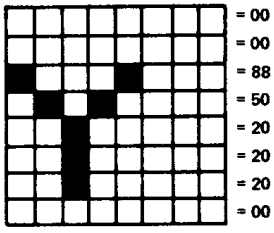
PATTERN 77



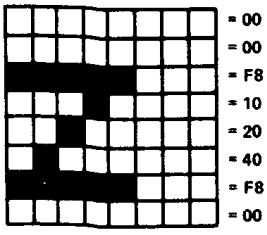
PATTERN 78



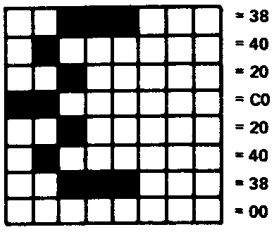
PATTERN 79



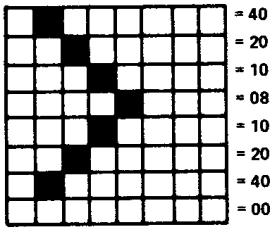
PATTERN 7A



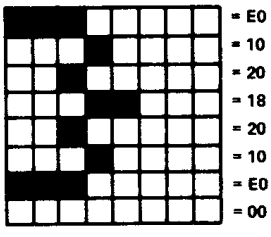
PATTERN 7B



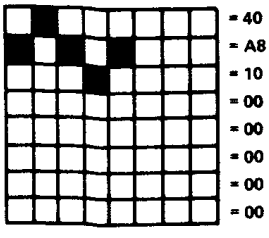
PATTERN 7C



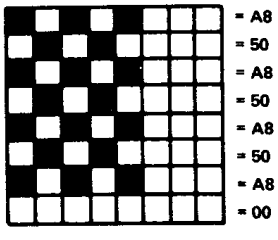
PATTERN 7D



PATTERN 7E



PATTERN 7F



APPENDIX B

CHOOSING VRAM MEMORY

When choosing the VRAM memory, the user must take into consideration the propagation delay times of the system in addition to the access time of the memory and data setup time of the VDP.

After the VDP outputs a low level signal on $\overline{RAS}$, there is a delay time ($t_{d(RAS)}$) for this low level to reach the VRAM memory; there is a similar delay ($t_{d(CAS)}$) for a signal output on the $\overline{CAS}$ pin to reach the VRAM memory. Finally, there is a delay ($t_{d(data)}$) for data output by the memory to reach the VDP. These delays (shown in Figure B1) depend on the length of the wires between VDP and memory, and on the capacitive load being driven.

Valid data appearing on RD0-RD7 is strobed into the VDP when $\overline{CAS}$ is brought high. Therefore, the memory chosen must have fast enough access times, $t_{a(R)}$ and $t_{a(C)}$, so that valid data is present on RD0-RD7 when a positive transition occurs on $\overline{CAS}$.

For 16-K memories from Texas Instruments (TMS4116-XX), the times, t_{RL-CL} and $t_{a(C)}$, can vary, but their sum is equal to $t_{a(R)}$ ($t_{RL-CL} + t_{a(C)} = t_{a(R)}$). Thus, when $t_{d(RAS)} \geq t_{d(CAS)}$, the limiting access time is $t_{a(R)}$.

After the memory receives a negative transition on the $\overline{RAS}$ input, the memory access time, $t_{a(R)}$, must be fast enough so that valid data is present on RD0-RD7 when $\overline{CAS}$ goes high (see Figure B2). The equation for this is:

$$t_{RL-CL} + t_{wCL} \geq t_{d(R)} + t_{d(data)} + t_{su(D-CH)}$$

Under worst case conditions, this equation can be used to find out how much time is allowed for system delays using different memories.

TABLE B-1 — WORST CASE TIMING FOR VDP.

MEMORIES	SYSTEM DELAYS
t_{wCL}	230 ns MIN
t_{RL-CL}	40 ns MIN
$t_{su(D-CH)}$	60 ns MAX

If the values from Table B1 are placed in the equation, we find ($t_{RL-CL} + t_{wCL}$) VDP MIN $\geq$ [($t_{d(RAS)} + t_{d(data)}$) SYS + $t_{a(R)}$ MEM + t_{suD-CH} VDP MAX]

210 ns - $t_{a(R)}$ MEM MAX $\geq$ [$t_{d(RAS)} + t_{d(data)}$] SYS MAX

TABLE B-2 — DRAM SYSTEM DELAYS

PART NO.	$t_{a(R)}$	SYSTEM DELAYS
4116-15	150 ns	60 ns MAX
4116-20	200 ns	10 ns MAX
4116-25	250 ns	-40 ns MAX

From the data given here, the VDP will work with both -15 and -20 TMS4116 dynamic RAMs provided the system delays are small enough. The VDP does not meet the $t_{a(R)}$ specifications for the -25 TMS4116 and is unable to use the -25 under worst case conditions. The VDP has been verified to work with both -15 and -20 TMS4116s in a system application. Note that in addition to the equation derived above, that all memory timing requirements must be met as specified in a memory data book.

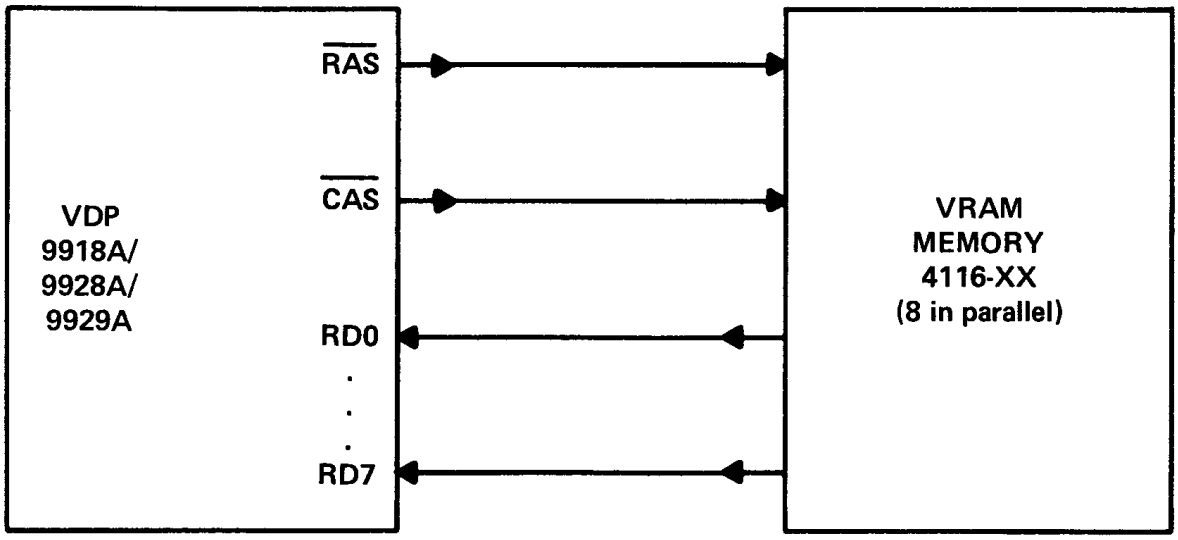


FIGURE B-1 – MEMORY CONFIGURATION SHOWING DELAY TIMES

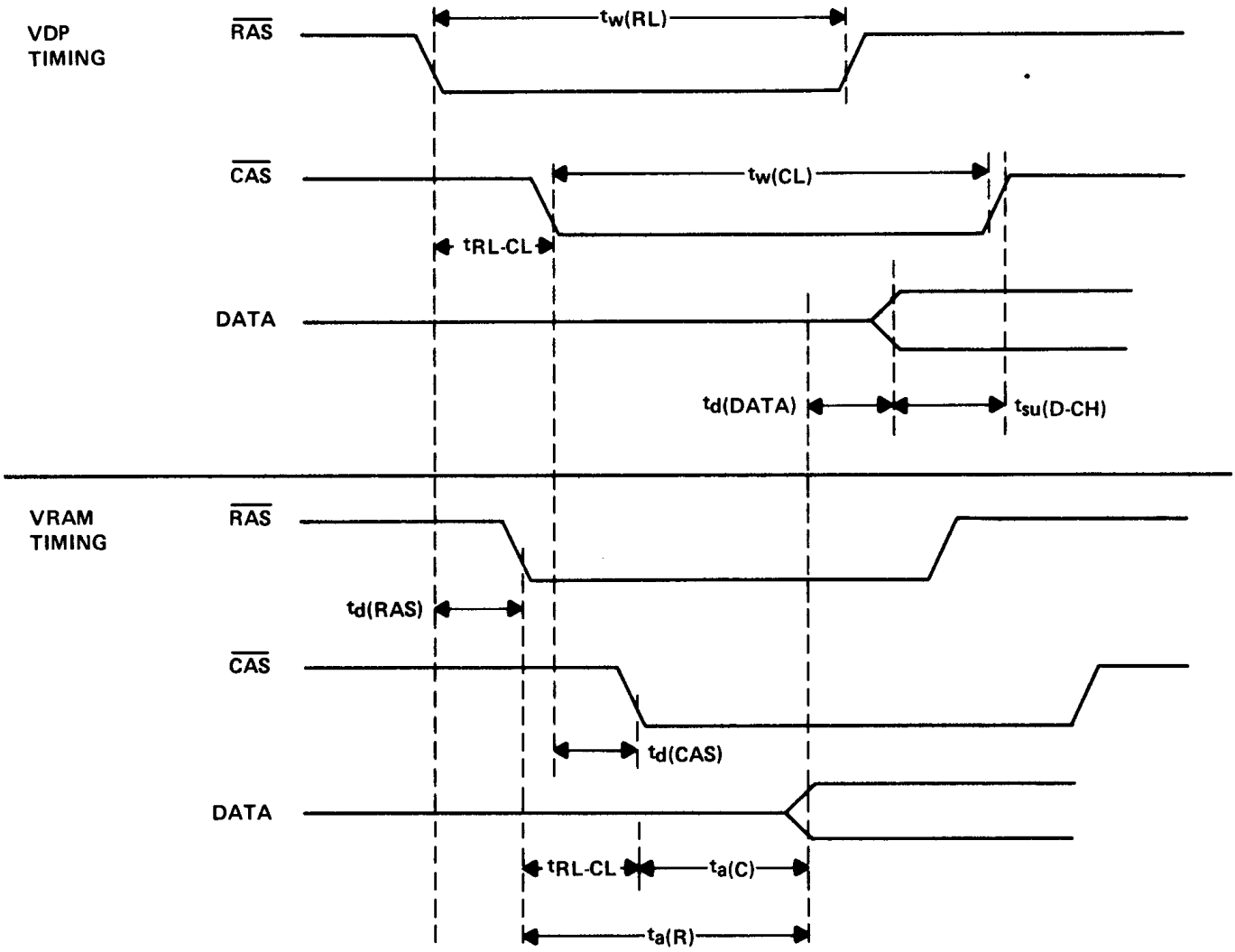
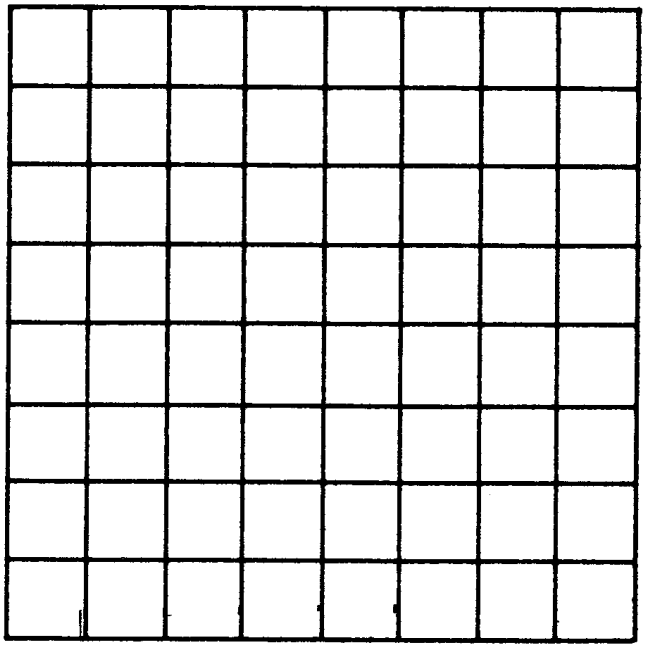


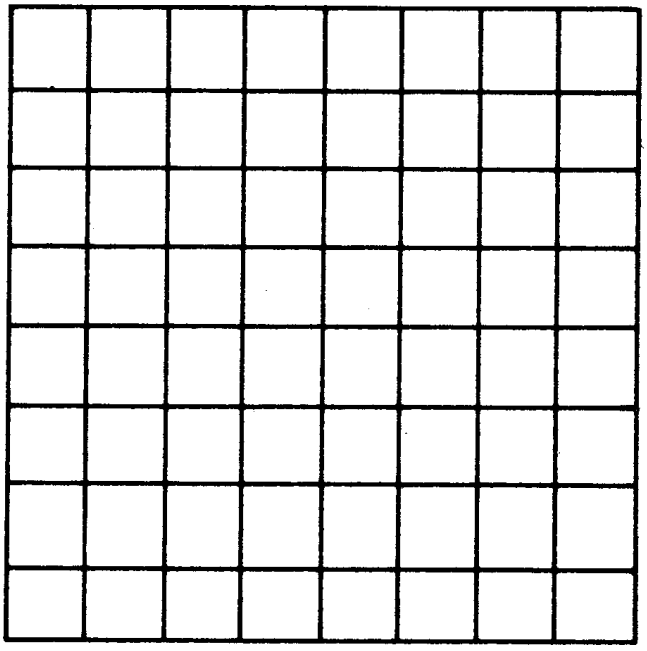
FIGURE B-2 – RELATIVE TIMING OF VRAM TO VDP

APPENDIX C

PATTERN AND SCREEN WORKSHEETS

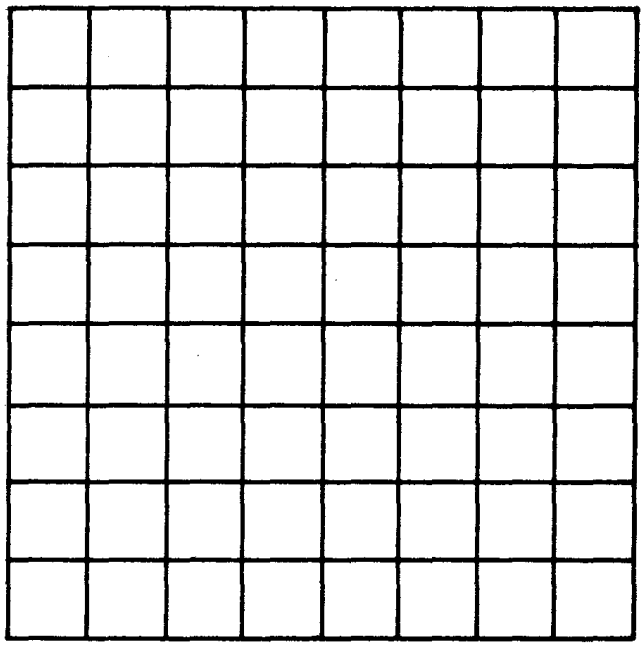
PATTERN
NAME





PATTERN
NAME

_____



_____

_____

_____

_____

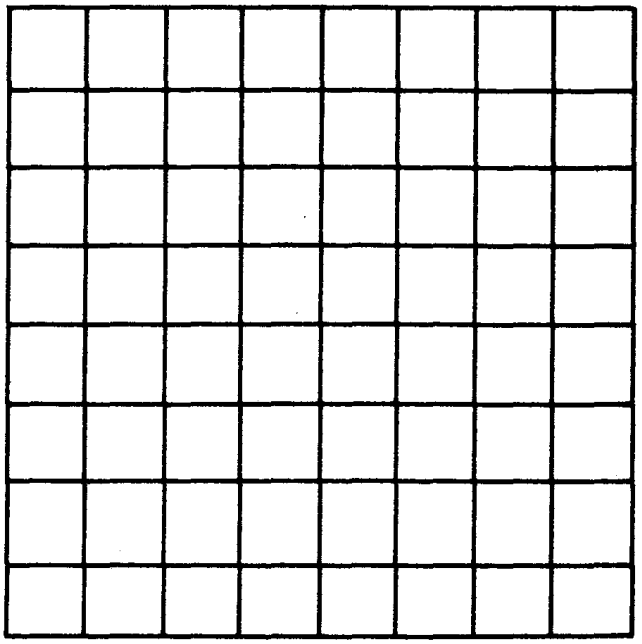
_____

_____

_____

_____

_____



_____

_____

_____

_____

_____

_____

_____

_____

TEXAS INSTRUMENTS

November 1982
MP010A

Post Office Box 1443 • Houston, Texas 77001
Semiconductor Group

Printed in U.S.A.