MEGATEK 7000

GRAPHICS DISPLAY SYSTEM

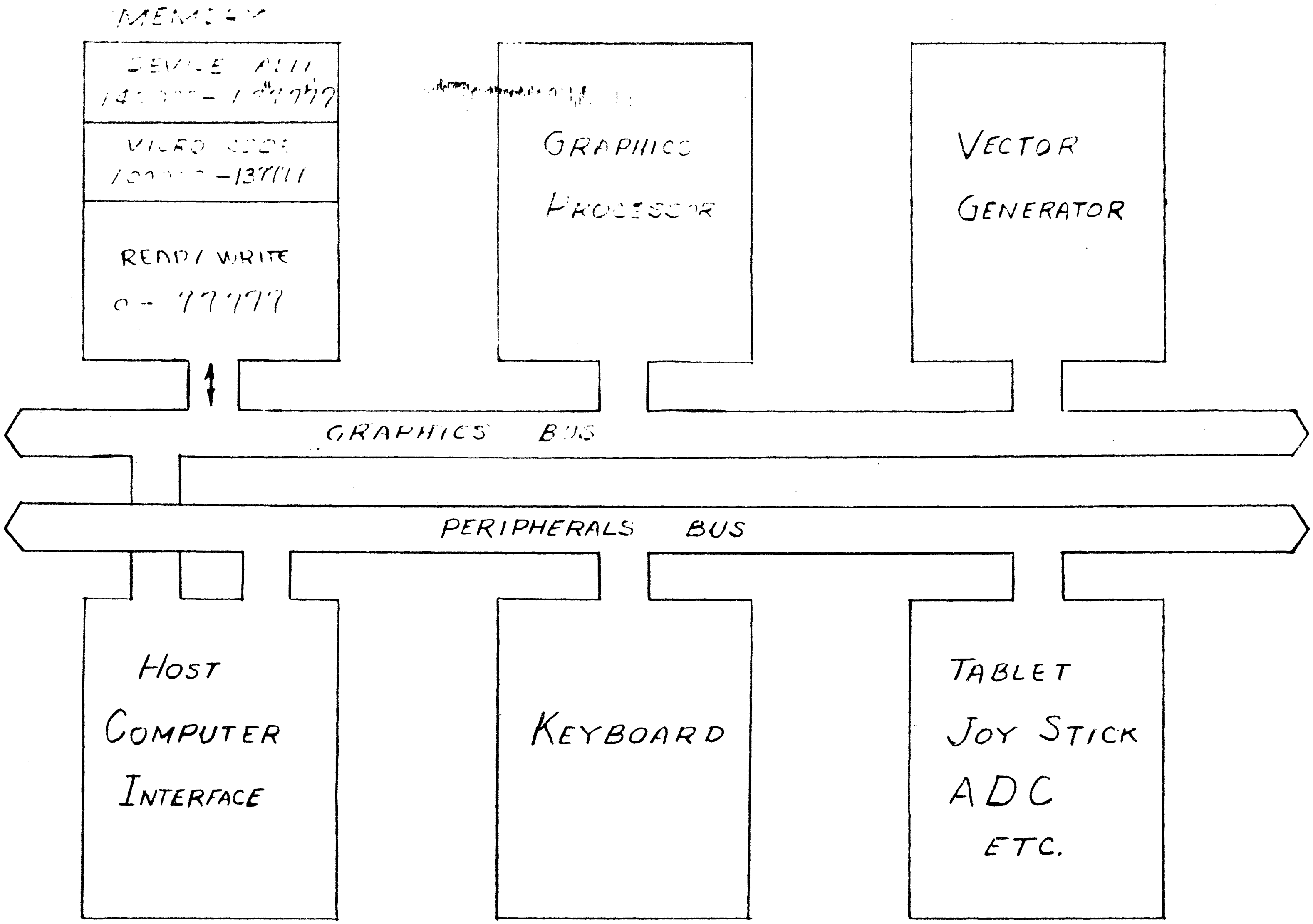
INTERFACE DESCRIPTION

TABLE OF CONTENTS

1. INTRODUCTION

The MEGRAPHIC 7000 Graphics Display System is a stand-alone system containing its own refresh memory and graphics display processor. The 7000 is a bus-oriented system and is therefore easily interfaced to a wide variety of host processors should the user desire. This document presents a detailed description of the operation of the bus and requirements for a user-designed bus interface.

Figure 1-1 shows the organization of the MEGRAPHIC 7000 Graphics Display System. The graphics bus ties together the graphics display processor, the vector generator and memory. The memory carries both the micro-code for the graphics display processor and the display list. Note that the graphics bus is a memory-mapped system and that addresses OCTAL: 140000 through OCTAL: 177777 are reserved [illegible]



MEGRAPHIC 7000 SYSTEM ORGANIZATION

FIGURE 1-1

2. PRIORITY DETERMINATION

The establishment of priorities on the graphics and peripherals buses is performed in a different manner. The first section will deal with priority determination on the graphics bus and the second section with priority determination on the peripherals bus.

2.1 GRAPHICS BUS

Priority is established by a single line that is daisy-chained from one device to another. The highest priority device is the device which is at the head or top of the chain. The priority signal is then passed from device to device down to the device with the lowest priority located at the bottom of the chain. The priority signal, which is normally LOW, is passed into each device through the signal pin $\overline{\text{GPIN}}$ and exits each device through the signal pin $\overline{\text{GPOUT}}$.

Figure 2-1 shows a typical example of bus priority circuitry located on a bus interface card.

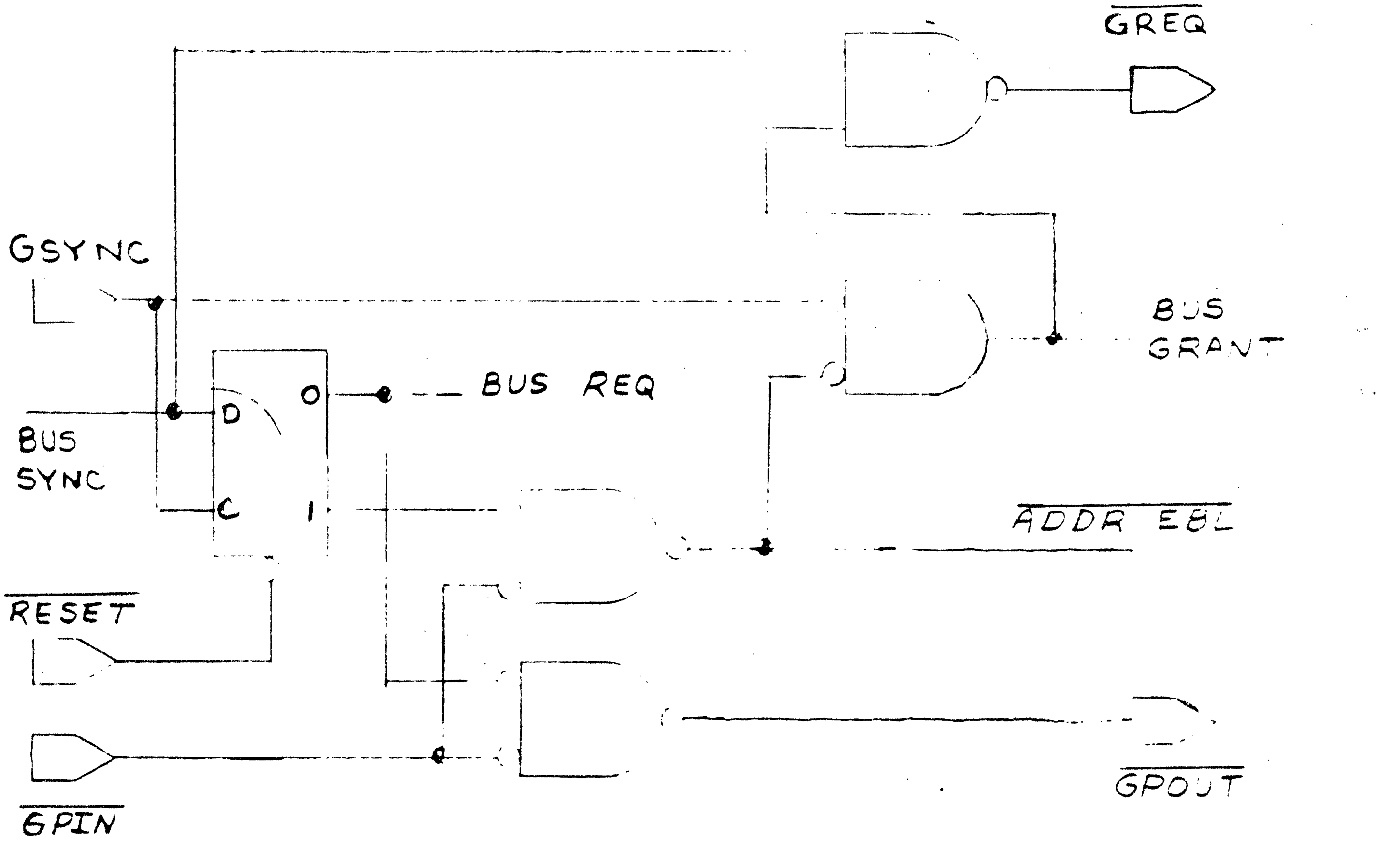
In the quiescent or stand-by state the BUS REQ flip-flop is not set indicating that the interface does not require the bus. In this state the signal $\overline{\text{GPOUT}}$ will reflect the state of the signal $\overline{\text{GPIN}}$. If $\overline{\text{GPIN}}$ is LOW then no other device of higher priority is in control of the bus. If $\overline{\text{GPIN}}$ is HIGH, some device of higher priority is in control of the bus. The interface then passes the HIGH signal down the chain by setting $\overline{\text{GPOUT}}$ HIGH. Since $\overline{\text{GPIN}}$ is HIGH, the interface is inhibited from gaining control of the bus and, by forcing $\overline{\text{GPOUT}}$ HIGH, all devices lower in the chain are similarly inhibited. Note that, when $\overline{\text{GPIN}}$ is HIGH, $\overline{\text{ADDR EBL}}$ is held HIGH which in turn holds BUS GRANT in a LOW state. In order to gain control of the bus, the signal BUS GRANT must be HIGH.

The signal GSYNC is used to synchronize requests for the bus. When the bus is idle, GSYNC looks like a clock signal. When the bus is busy, GSYNC is forced LOW by the signal $\overline{\text{GREQ}}$.

Assuming the bus is idle, $\overline{\text{GDIN}}$ will be low. A typical sequence to gain control of the bus is as follows: referring to Figure 2-2, the first action is for the interface to raise BUS SYNC. Upon the next positive transition of GSYNC, the bus request flip flop is set and BUS REQ goes HIGH. This in turn causes

1) $\overline{\text{GPOUT}}$ to go HIGH thus inhibiting all devices of lower priority,

2) $\overline{\text{ADDR EBL}}$ to go LOW thus directing the interface to place the desired address on the bus.



Typical Bus Priority Network

Figure 2-1

Upon the next negative-going transition of GSYNC the following occurs:

1) BUS GRANT goes HIGH indicating the interface now has control of the bus and may transfer data.

2) $\overline{\text{GREQ}}$ is forced LOW indicating the bus is busy which in turn causes GSYNC to be held LOW.

At the conclusion of the data transfer, the interface allows BUS SYNC to go LOW. This causes $\overline{\text{GREQ}}$ to go HIGH and allows the signal GSYNC to start clocking once more. On the first transition (positive-going), the bus request flip-flop is reset which causes

1) $\overline{\text{ADDR EBL}}$ to go HIGH.

2) $\overline{\text{GPOUT}}$ to go LOW thus re-establishing the priority chain.

The transfer sequence as described provides a well defined process for gaining access to the bus. Some conditions that may arise during normal operation, however, deserve mention.

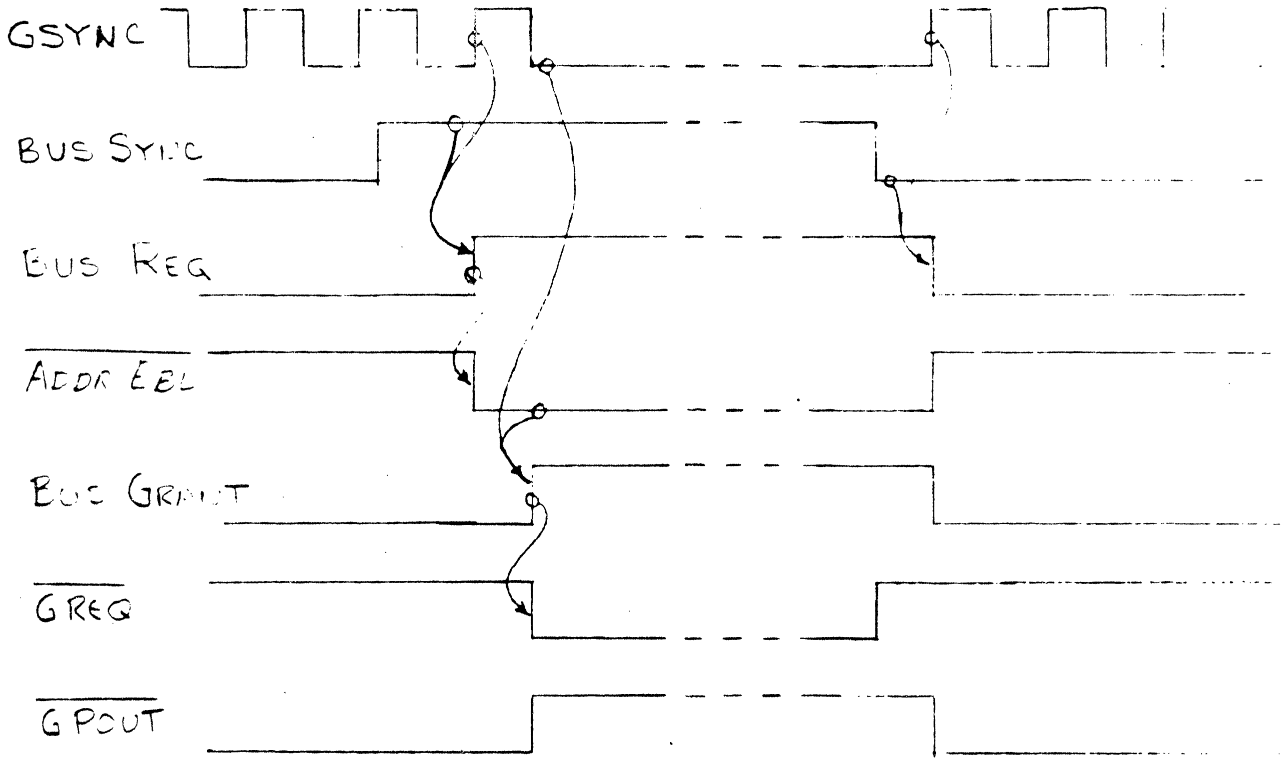
If, for example, two or more bus request flip-flops are set on a particular GSYNC transition, the device highest in the priority chain will gain control of the bus. At the conclusion of this transfer, at least one pulse of GSYNC will appear. This pulse will reset the bus request flip-flop of the device that is terminating a transfer sequence and allow any other waiting devices to set their bus request flip-flop, thus re-examining the entire priority chain.

Any device that raises its BUS SYNC signal HIGH is ignored until the next positive-going transition of GSYNC.

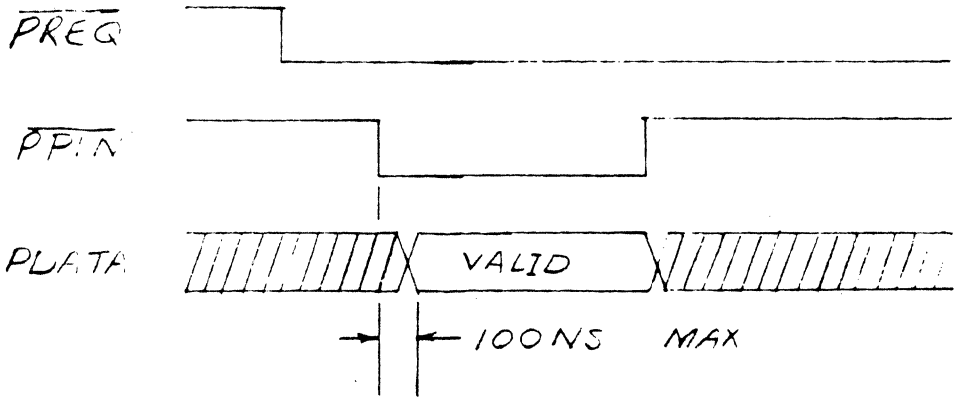
2.2 PERIPHERALS BUS

Figure 2-3 shows the timing for the peripherals bus priority determination. When some device on the bus lowers $\overline{\text{PREQ}}$, it is the signal for the bus interface to lower $\overline{\text{PPIN}}$. When $\overline{\text{PPIN}}$ is drawn LOW, the interrupting device of the highest priority will place its device select address on the data bus within 100 ns. The bus interface must then accept the device select address before allowing $\overline{\text{PPIN}}$ to go HIGH.

The device select address is placed on the lines $\overline{\text{PDATA10}}$ through $\overline{\text{PDATA15}}$. Bit 0 of the device select appears on $\overline{\text{PDATA10}}$ and bit 5 of the device select appears on $\overline{\text{PDATA15}}$. Both are, of course, in inverted form. In addition, $\overline{\text{PDATA9}}$ informs the processor whether the interrupting device is an input device or an output device.

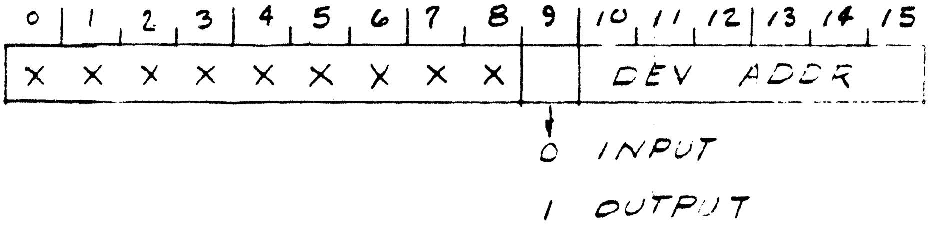


GRAPHICS BUS PRIORITY TIMING
FIGURE 2-2



PERIPHERALS BUS PRIORITY TIMING

FIGURE 2-3



PERIPHERALS BUS PRIORITY FORMAT

FIGURE 2-4

The line $\overline{\text{PREQ}}$ will return HIGH if both the interrupting device is serviced and no further requests are pending. If additional requests are pending, $\overline{\text{PREQ}}$ will remain LOW until all requests are satisfied.

3. TRANSFER SEQUENCES

Once access to the bus has been obtained data may be transferred. The two types of transfers are read and write. A read transfers data from the bus and a write transfers data to the bus. The transfer sequences are similar; however, important differences in timing do exist. The first section deals with the graphics bus and the second section with the peripherals bus.

3.1 GRAPHICS BUS

3.1.1 READ SEQUENCE

A typical read sequence is shown in Figure 3-1 for a bus interface card transferring data off of the bus.

As the interface gains access to the bus, the ADDR EBL signal will go LOW. The ADDR EBL signal is a signal generated locally by the interface as described in Section 2. When the ADDR EBL is LOW, the interface places the first address on the bus. BUS GRANT is similarly generated locally by the interface. When BUS GRANT goes HIGH, GRD/WR should go HIGH and remain there for the duration of the transfer.

After the address data have settled, the bus interface should draw GARDY LOW. When the data are presented on the bus and are valid, the bus pulls GDRDY LOW. The bus interface uses the falling edge of GDRDY to strobe data off of the bus. After the data are received, the bus interface allows GARDY to go HIGH. The bus will respond by allowing GDRDY to go HIGH. The bus interface may now repeat the process by presenting a new address and pulling GARDY LOW.

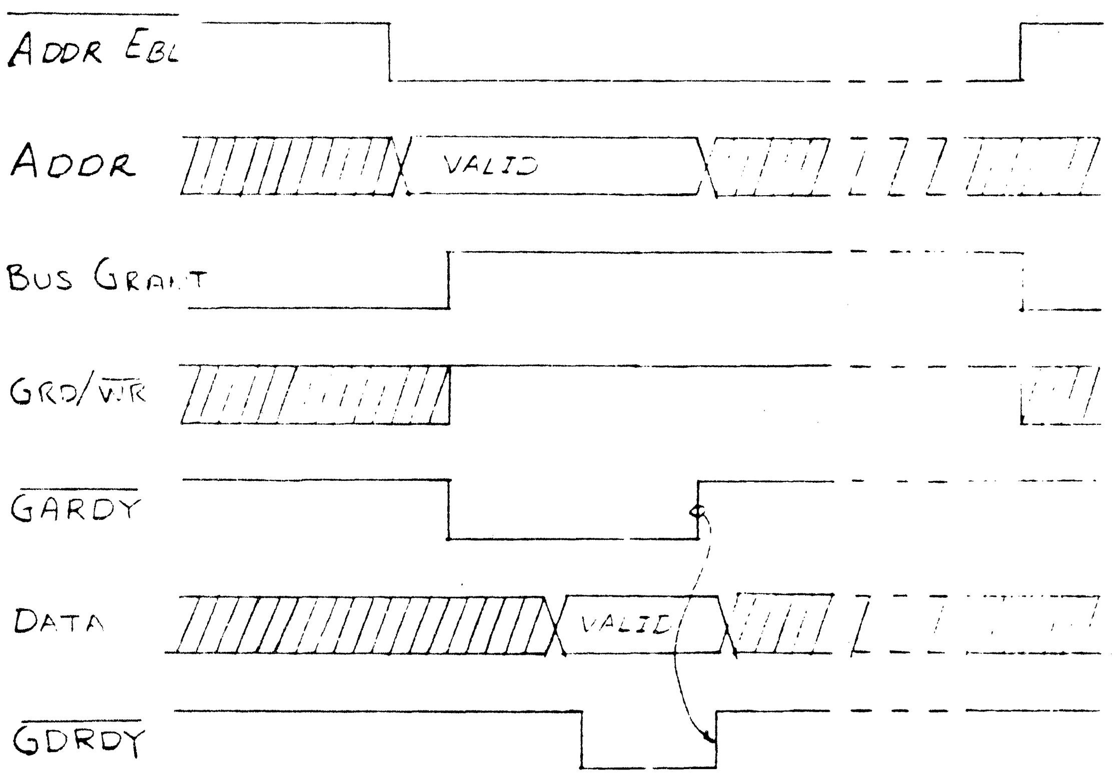
At the conclusion of the transfer, ADDR EBL is returned to its HIGH state and BUS GRANT is returned to its LOW state as presented in Sec. 2.

3.1.2 WRITE SEQUENCE

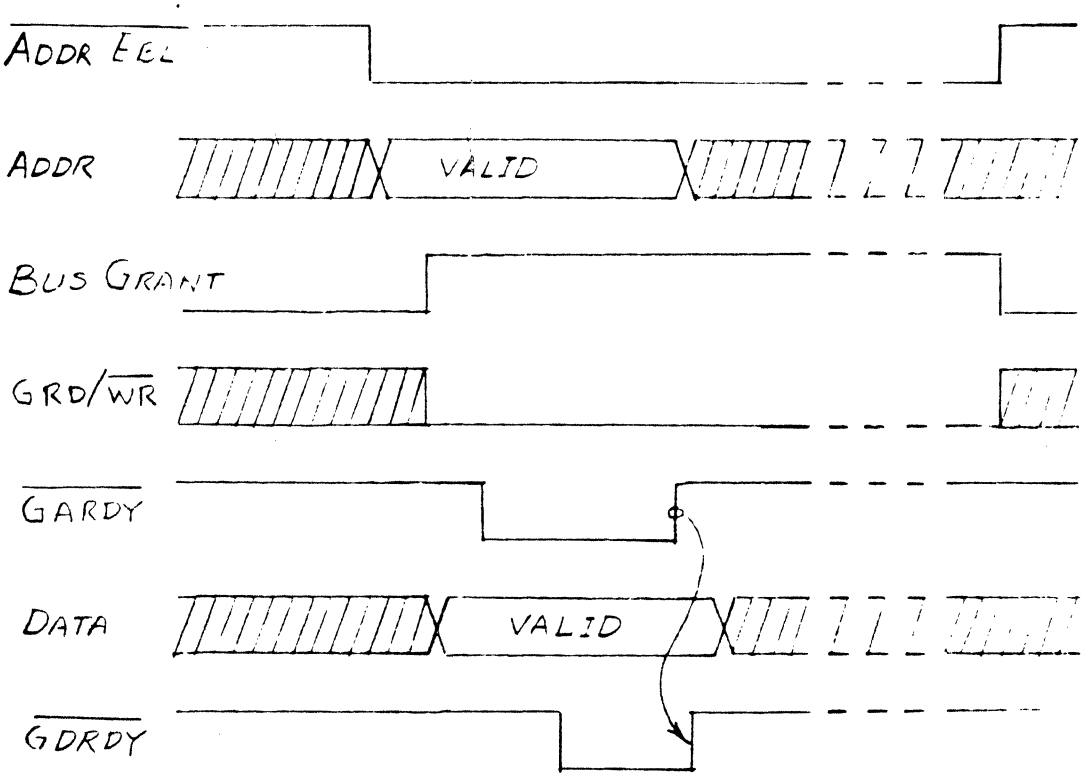
A typical write sequence is shown in Figure 3-2 for a bus interface card transferring data onto the bus.

As the interface gains access to the bus, the ADDR EBL signal will go LOW. The ADDR EBL signal is a signal generated locally by the interface as described in Section 2. When the ADDR EBL is LOW, the interface places the first address on the bus. BUS GRANT is similarly generated locally by the interface. When BUS GRANT goes HIGH, GRD/WR should go LOW and remain there for the duration of the transfer.

After both the address and data have been applied to the bus and have settled, the bus interface should draw GARDY LOW. When the



Graphics Bus Read Timing
Figure 3-1


$\overline{\text{ADDR EEL}}$
ADDR
VALID
BUS GRANT
GRD/$\overline{\text{WR}}$
$\overline{\text{GARDY}}$
DATA
VALID
$\overline{\text{GDRDY}}$

GRAPHICS BUS WRITE TIMING

FIGURE 3-2

bus has accepted the data, it will respond by bringing $\overline{\text{GDRDY}}$ LOW. The interface then responds by allowing $\overline{\text{GARDY}}$ to go HIGH. The bus interface may now repeat the process by presenting a new address and new data on the bus and pulling $\overline{\text{GARDY}}$ LOW.

At the conclusion of the transfer, $\overline{\text{ADDR EBL}}$ is returned to its HIGH state and BUS GRANT is returned to its LOW state as presented in Sec. 2.

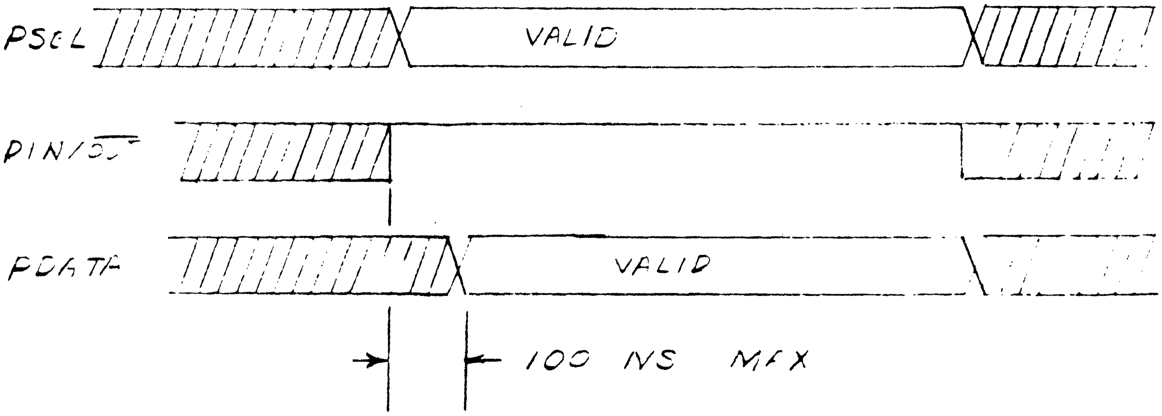
3.2 PERIPHERALS BUS

3.2.1 READ SEQUENCE

The timing for a read operation on the peripherals bus is shown in Figure 3-3. Within 100 ns. after the peripheral select lines (PSEL0 through PSEL5) and the PIN/$\overline{\text{OUT}}$ line have been set by the bus interface, the peripheral data bus will present data from the selected device. The bus interface must then accept the data before lowering PIN/$\overline{\text{OUT}}$ and before changing any of the select lines. The bus provides no clocking signal for the read operation. Note that data are transferred from the bus into the bus interface during a read.

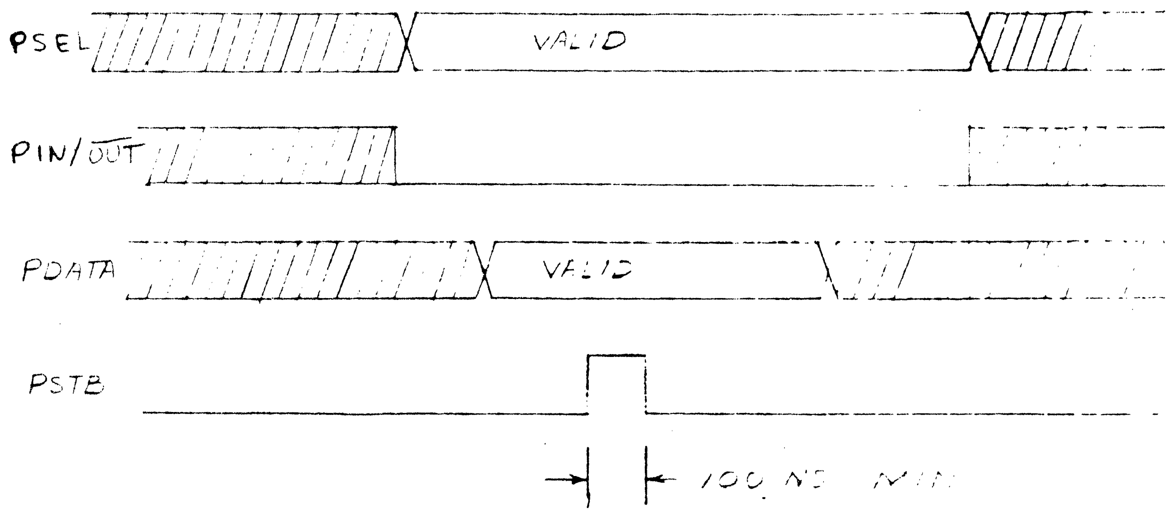
3.2.2 WRITE SEQUENCE

The timing for a write operation on the peripherals bus is shown in Figure 3-4. Operation is similar to a read except that the bus interface must supply a strobe pulse (PSTB) in order to cause the peripheral device to accept the data.



PERIPHERALS BUS READ TIMING

FIGURE 3-3



PERIPHERALS BUS WRITE TIMING

FIGURE 3-4

4. INTERFACE DESIGN

The logical design of a bus interface to the MEGRAPHIC 7000 Graphics Display System should follow the guidelines set forth in Sections 2 and 3. Because of the high speed nature of the bus certain precautions must be taken in the actual circuit design of bus interfaces.

In order to reduce the possibility of spurious transients interfering with the operation of the bus, line filters should be placed on critical lines. A typical filter is shown in Figure 4-1. Lines to which this type of filter should be applied are presented in Table 4-1.

Table 4-1

Signal Lines Requiring Filters

$\overline{\text{GARDY}}$
$\overline{\text{GDRDY}}$
$\overline{\text{GSNS}}$
$\overline{\text{RESET}}$

PSTB
PCLR
PSTR

The graphics bus data lines (GDATA0 through GDATA31) and graphics bus address lines (GADDR0 through GADDR15) are equipped with tri-state drivers. Recommended loading for these lines is one low power Schottky load.

The host computer bus interface should supply all necessary pull-up resistors for open collector outputs. These lines are presented in Table 4-2.

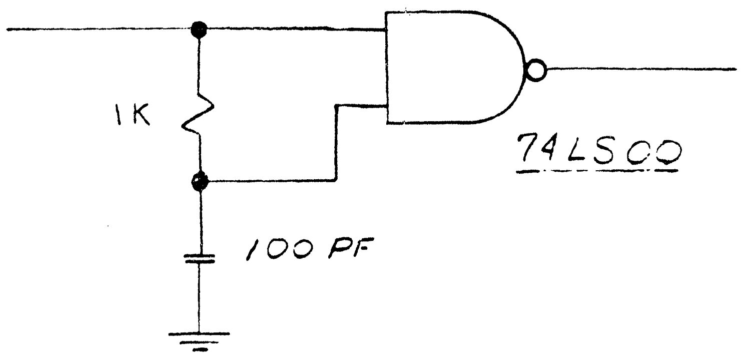
Table 4-2

Signal Lines Requiring 330/390

Pull-Up Resistors

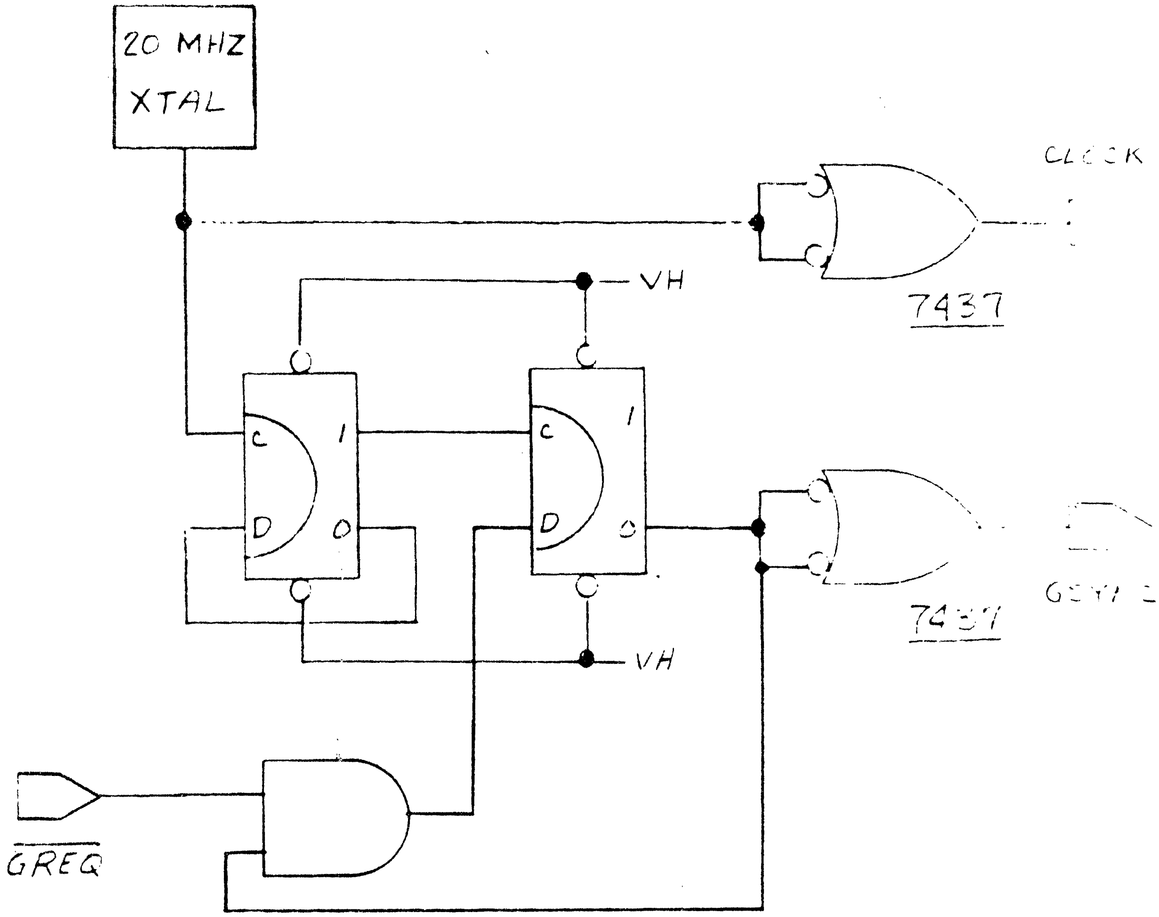
$\overline{\text{PREQ}}$
$\overline{\text{PDATA0}}$ through
$\overline{\text{PDATA15}}$

$\overline{\text{GREQ}}$
$\overline{\text{GARDY}}$
$\overline{\text{GDRDY}}$
$\overline{\text{GSNS}}$
GRD/$\overline{\text{WR}}$
$\overline{\text{RESET}}$



TYPICAL LINE FILTER
FIGURE 4-1

The host computer interface must also supply the GSYNC and 20MHz clock signals. A recommended circuit is presented in Figure 4-2. Note that the 20 MHz clock must be crystal controlled in order to insure proper operation of the graphics processor and that high current drivers (7437) are provided for the CLOCK and GSYNC lines.



Typical Clock Circuit

Figure 4-2

APPENDIX A - PIN LIST

(100 Pin Winchester Connector) *Use Filter on Line

PIN	SIGNAL	PIN	SIGNAL
1	GND	50	GADDR 7
2	GND	51	GADDR 8
3	+5	52	GADDR 9
4	+5	53	GADDR 10
5	$\overline{\text{PPOUT}}$	54	GADDR 11
6	$\overline{\text{PPIN}}$	55	GADDR 12
7	$\overline{\text{GPOUT}}$	56	GADDR 13
8	$\overline{\text{GPIN}}$	57	GADDR 14
9	$\overline{\text{PREQ}}$	58	GADDR 15
10	$\overline{\text{GREQ}}$	59	GRD/$\overline{\text{WR}}$
11	GDATA 0	60	*$\overline{\text{GARDY}}$
12	GDATA 1	61	*$\overline{\text{GDRDY}}$
13	GDATA 2	62	*$\overline{\text{GSNS}}$
14	GDATA 3	63	GSYNC
15	GDATA 4	64	60 Hz
16	GDATA 5	65	CLOCK (20 MHZ)
17	GDATA 6	66	*$\overline{\text{RESET}}$
18	GDATA 7	67	*PSTB
19	GDATA 8	68	*PCLR
20	GDATA 9	69	*PSTR
21	GDATA 10	70	PSEL 0
22	GDATA 11	71	PSEL 1
23	GDATA 12	72	PSEL 2
24	GDATA 13	73	PSEL 3
25	GDATA 14	74	PSEL 4
26	GDATA 15	75	PSEL 5
27	GDATA 16	76	PIN/$\overline{\text{OUT}}$
28	GDATA 17	77	$\overline{\text{PDATA 0}}$
29	GDATA 18	78	$\overline{\text{PDATA 1}}$
30	GDATA 19	79	$\overline{\text{PDATA 2}}$
31	GDATA 20	80	$\overline{\text{PDATA 3}}$
32	GDATA 21	81	$\overline{\text{PDATA 4}}$
33	GDATA 22	82	$\overline{\text{PDATA 5}}$
34	GDATA 23	83	$\overline{\text{PDATA 6}}$
35	GDATA 24	84	$\overline{\text{PDATA 7}}$
36	GDATA 25	85	$\overline{\text{PDATA 8}}$
37	GDATA 26	86	$\overline{\text{PDATA 9}}$
38	GDATA 27	87	$\overline{\text{PDATA 10}}$
39	GDATA 28	88	$\overline{\text{PDATA 11}}$
40	GDATA 29	89	$\overline{\text{PDATA 12}}$
41	GDATA 30	90	$\overline{\text{PDATA 13}}$
42	GDATA 31	91	$\overline{\text{PDATA 14}}$
43	GADDR 0	92	$\overline{\text{PDATA 15}}$
44	GADDR 1	93	-15
45	GADDR 2	94	-15
46	GADDR 3	95	+15
47	GADDR 4	96	+15
48	GADDR 5	97	+5
49	GADDR 6	98	+5
		99	GND
		100	GND

APPENDIX B

GLOSSARY OF SIGNAL TERMS

CLOCK

20 MHz continuous clock signal.

GADDR0 through GADDR 15

Graphics bus address lines. These lines are tri-state and switched to the high impedance state when the $\overline{\text{GARDY}}$ line is HIGH. These lines present a single low power Schottky load and data are true when HIGH. Bit 0 is the most significant bit.

$\overline{\text{GARDY}}$

Graphics bus address ready. This line should be driven by an open collector gate. $\overline{\text{GARDY}}$ is driven LOW when the graphics address bus has a valid address presented to it. Line is bidirectional and also should be filtered as described in Section 4.

GDATA0 through GDATA 31

Graphics bus data lines. These lines are tri-state and are switched to the high impedance state when the $\overline{\text{GDRDY}}$ line is HIGH. These lines present a single low power Schottky load and data are true when HIGH. Bit 0 is the most significant bit.

$\overline{\text{GDRDY}}$

Graphics bus data ready. This line should be driven by an open collector gate. $\overline{\text{GDRDY}}$ is driven LOW when the graphics data bus has valid data presented to it. Line is bidirectional and also should be filtered as described in Section 4.

GND

Ground.

$\overline{\text{GPIN}}$

Graphics bus priority in. This signal is used in a daisy-chain fashion to establish priority for bus requests. When $\overline{\text{GPIN}}$ is LOW it indicates that no other devices on the bus of higher priority are requesting the bus.

$\overline{\text{GPOUT}}$

Graphics bus priority out. This signal is used in a daisy-chain fashion to establish priority for bus requests. When $\overline{\text{GPOUT}}$ is driven HIGH by the bus interface, it inhibits all devices of lower priority from requesting the bus.

$\overline{\text{GREQ}}$

Graphics bus request line. This line should be driven by an open collector gate. When $\overline{\text{GREQ}}$ is LOW the bus timing signal GSYNC is inhibited.

GRD/$\overline{\text{WR}}$

Graphics bus read signal. This line should be driven by an open collector gate. GRD/$\overline{\text{WR}}$ is LOW when data are transferred from the bus interface to the bus (WRITE).

$\overline{\text{GSNS}}$

Graphics bus sense signal. This line is held LOW when the vector generator is unable to accept data due to a full input buffer. When such an overflow occurs, $\overline{\text{GSNS}}$ will go LOW and $\overline{\text{GDRDY}}$ will remain HIGH. This line is not normally used by a host computer bus interface.

GSYNC

Graphics bus synchronization signal. This signal is a clocking signal used to synchronize bus requests. When the graphics bus is busy, the GSYNC signal is inhibited and held LOW.

PCLR

Peripherals bus clear signal. Resets all peripherals devices when HIGH. Line should be filtered as described in Section 4.

$\overline{\text{PDATA0}}$ through $\overline{\text{PDATA 15}}$

Peripherals bus data lines. These lines are open collector data lines and should be terminated with 230 to 390 ohm resistances. Data are true when LOW and bit 0 is the most significant bit.

PIN/$\overline{\text{OUT}}$

Peripherals bus input/output signal. This line should be driven by an open collector gate. PIN/$\overline{\text{OUT}}$ is HIGH when data are to be transferred from the bus to the peripheral device. PIN/$\overline{\text{OUT}}$ is LOW when data are to be transferred from the peripheral device to the bus. PIN/$\overline{\text{OUT}}$ operates in much the same fashion as GRD/$\overline{\text{WR}}$.

$\overline{\text{PPIN}}$

Peripherals bus priority in. This signal is used in a daisy-chain fashion to establish priority for bus requests. When $\overline{\text{PPIN}}$ is LOW, it causes the requesting device to place its device select address on the data bus.

$\overline{\text{PPOUT}}$

Peripherals bus priority out. This signal is used in a daisy-chain fashion to establish priority for bus requests. When $\overline{\text{PPOUT}}$ is driven HIGH by the bus interface, it inhibits all devices of lower priority from requesting the bus.

$\overline{\text{PREQ}}$

Peripherals bus request line. This line should be driven by an open collector gate. When $\overline{\text{PREQ}}$ is LOW, it indicates some device on the peripheral bus is requesting service. Typical response is to lower $\overline{\text{PPIN}}$.

$\overline{\text{PSTB}}$

Peripherals bus data stable. This line should be terminated by a line filter as described in Section 4. $\overline{\text{PSTB}}$ is driven LOW when the peripherals data bus has valid data presented to it.

PSEL0 through
PSEL5

Peripheral bus select lines. These lines select the desired device on the peripheral bus and hence perform a function similar to the address lines on the graphics bus. Data are true when HIGH and bit 0 is the most significant bit.

PSTR

Peripheral bus start signal. This signal when raised HIGH will cause a device dependent function to be performed in the device whose device select code appears on the PSEL0 through PSEL 5 lines. This line should be filtered as described in Section 4.

APPENDIX C MEMORY MAPPED IO ADDRESSES
(Graphics Bus Only)

OCTAL	
177761	Graphics Processor 0
177762	Graphics Processor 1
177764	Graphics Processor 2
177770	Graphics Processor 3
177600	Vector Generator 0
177601	Vector Generator 1
177602	Vector Generator 2
177603	Vector Generator 3