Sun 1024 Video Board

User Manual

Sun Microsystems Inc

February 1983

Revision B

The Sun 1024 Video Board is a bitmap graphics subsystem offering a 1024 by 1024 frame buffer, "RasterOp" hardware support, and high-speed updating on a single board compatible with the IEEE 796 Bus (Intel Multibus).

This document describes the architecture, programming, and the installation of the Sun 1024 Video Board.

Multibus is a trademark of Intel Corporation. Sun is a trademarks of Sun Microsystems Inc.

Table of Contents

List of Figures

1. System Architecture

1.1. Features

- 1024 by 1024 pixel frame buffer

- 1024 by 800 pixel visible display area

- bit-map is organized by (x,y) coordinates

- arbitrary 1 by 16 rectangles directly accessible

- on-board "RasterOP" unit with 256 combination functions

- Vector drawing speed: 1 pixel per microsecond

- Character painting speed: 16 microseconds

- Screen fill: 64 milliseconds

- Single board compatible with IEEE-796 Bus/Intel Multibus

- 5V only operation.

1.2. Overview

The Sun 1024 video board brings high-resolution and high-speed bitmap graphics capabilities to IEEE-796/Intel Multibus based systems.

The Sun 1024 video board was specifically designed for interactive applications such as text processing, CAD, CAM, programming environments, and process control. In particular, the Sun video board can display composite images, combining text and graphics, multiple character sets, variable width fonts, multi-window systems, menus, and other symbols. In addition, the Sun video board is highly suited for applications traditionally served with vector or storage tube displays.

1.3. Introduction

This chapter provides an overview of the architecture of the Sun 1024 Video Board.

1.4. The Frame Buffer

The Sun video board hardware consists of a bit mapped memory or frame buffer, coupled with a Raster-function unit capable of performing bit manipulation functions on the contents of this memory. The graphical memory is 128K bytes, corresponding to a bitmap of 1024 by 1024 picture elements (pixels). Each pixel has a one bit value, i.e., is either ON or OFF. Although there are 1024 by 1024 pixels in the frame buffer, only a region 800 pixels high by 1024 pixels wide is displayed on the video screen, as shown in Figure 1-1.

The pixels not displayed (1024 by 224) are still accessible, and may be used as a cache to store bit maps that are not visible. Bitmaps from the invisible region may be moved (or copied) into the visible region by one of the raster operations described below.

The frame buffer is addressed in a cartesian coordinate system, in which <0,0> is the upper-left corner of the screen. Positive X displacement is to the right; positive Y displacement is downward.

The frame buffer is a dual ported memory. One port of the frame buffer connects (thru the function unit) to the host processor; the other port is dedicated to video-refresh, which has priority over host processor access. Nevertheless, the processor port can access the frame buffer once every 0.8 microseconds for a 16-bit operation.

The Sun video board is designed to drive a high-resolution interlaced monitor with decomposite video, horizontal, and vertical synchronization inputs. The nominal display area is 1024 by 800 pixels in landscape mode.
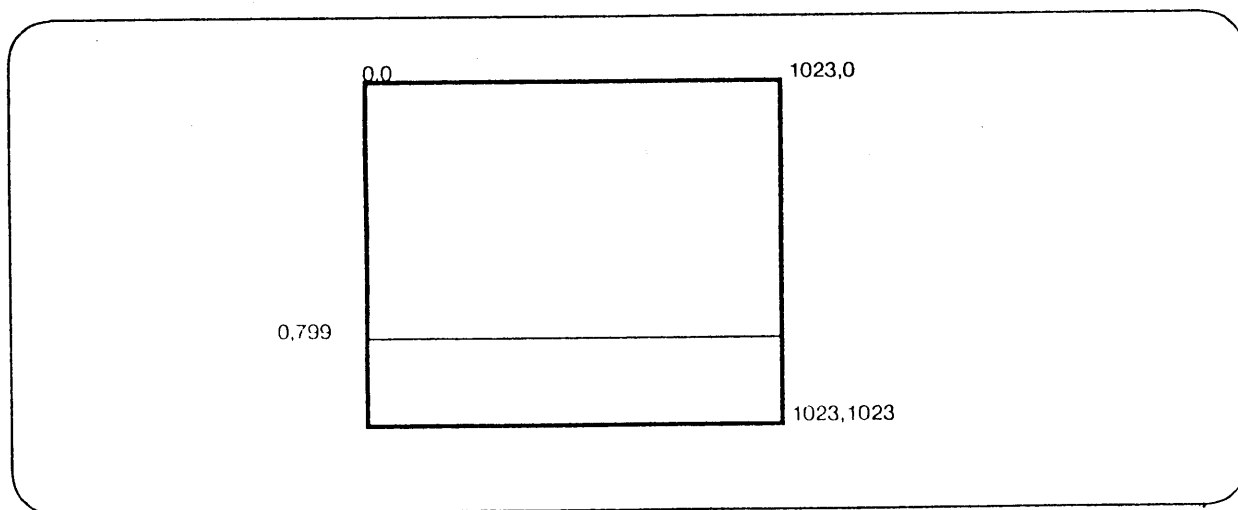

Figure 1-1: Landscape Video Monitor Display

1.5. RasterOp Architecture

The Sun Graphics system incorporates the concept of "RasterOP". RasterOp means that rectangular areas of display data ("Raster") are modified or combined according to a preselected operation ("Op"). RasterOp provides complete generality to paint characters, manipulate windows, scroll screens, and to draw vectors. An example for RasterOp is shown in Figure 1-2, in which a source character is copied to a destination anded with a mask.

During a RasterOp, the pixels accessed in the frame buffer are modified according to the function specified by the function register and as a result of data present in the data registers: destination, source, and mask. Destination is the operand being changed in the frame buffer, the source and mask operands can be loaded either from the frame buffer or from the main memory of the host processor. There are 256 possible functions mapping three boolean operands into a boolean result, any one of which is supported by the hardware.
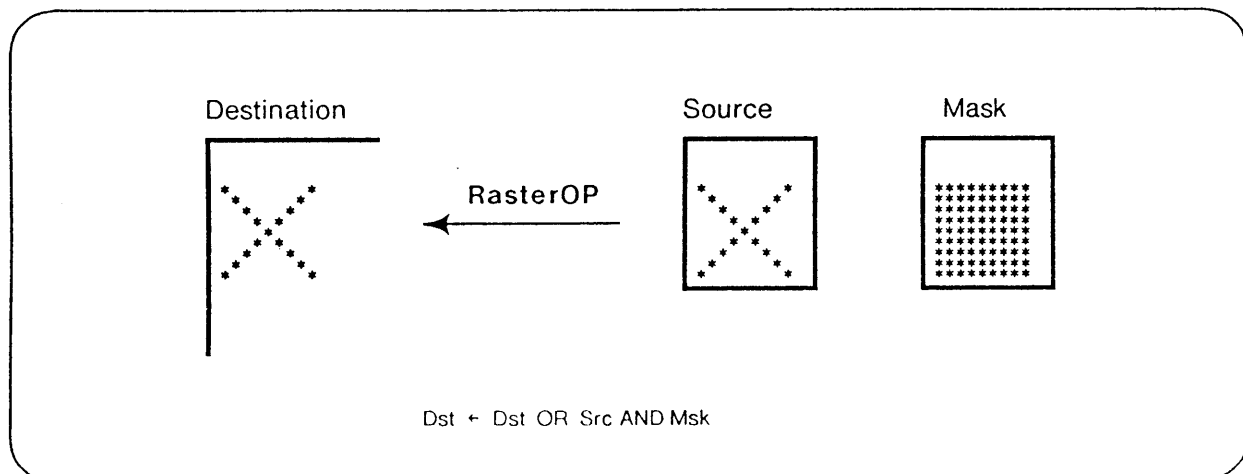

Figure 1-2: A RasterOp Operation

1.6. Hardware/Software Interface

The Sun video board divides the task of executing RasterOPs between hardware and software in such a way that maximum performance is maintained while minimizing hardware complexity.

In brief, the Sun video board is equipped with two special hardware mechanisms: cartesian coordinate access to arbitrary 1 by 16 pixel rectangles, and hardware execution of the raster operation. The Sun video board itself does not store any control information, such as algorithms for manipulating rectangles or drawing vectors. All control resides in the program of the host processor.

The Sun video board design has been optimized for such an interface by incorporating a number of registers to hold all state information that relates to a particular raster operation. This state information includes the data being manipulated, the RasterOp function to be performed, and the (x,y) addresses of the operands. Storing this state information locally on the video board extends the register set of the host processor and allows raster operations to proceed at full speed without having to reload critical state information.

1.7. Speed

A major problem with high-resolution bit-map graphics is the time required for creating and modifying the frame buffer image. The problem is rooted in the sheer number of bits being manipulated. For example, if a 1024 by 1024 pixel frame buffer were updated at a rate of 1 pixel per microsecond, it would take 1 second to fill the screen.

In comparision, the Sun 1024 Video Board, in conjunction with a high-performance processor board such as the Sun 68000 Board, can update 16 pixels per microsecond, or fill the screen in 64 milliseconds (excluding higher-level overhead). Painting a 16 by 16 pixel character takes approximately 20 microseconds and drawing vectors takes 1 microsecond per pixel, excluding overhead.

1.8. References

A. Bechtolsheim and F. Baskett, "High-Performance Raster Graphics for Microcomputer Systems", *Proceedings SIGGRAPH Conference*, Seattle, July 1980.

W. M. Newman and R. F. Sproull, *Principles of Interactive Computer Graphics*, second edition, McGraw Hill, 1979.

C. P. Thacker, E. M. McCreight, B. W. Lampson, R. F. Sproull, and D. R. Boggs, "Alto: A personal Computer", in Siewiorek, Bell, and Newell, eds., *Computer Structures: Readings and Examples*, McGraw Hill, 1979.

1.9. Specification Summary

Frame Buffer and Addressing

```
1024 by 1024 by 1, addressable in (x,y) coordinates.
Cartesian coordinate system with (0,0) in upper left corner.
Four sets of (x,y) registers are provided.
Each (x,y) registers pair points to a 1 by 16 pixel element.
(x,y) registers are updated by accessing the respective register.
```

Video Monitor and Video Interface

```
Ball HD-17, 77 Hz vertical, 30.86 kHz horizontal
Separate synchronization and video signals in TTL level.
```

Update Characteristics

```
One frame buffer update every 0.8 microseconds.
Update width is selectable from 1 to 16 bits.
Raster operation modifies destination data in frame buffer
in conjunction with source and mask data previously loaded.
Any one of the 256 possible raster operations can be selected.
Source and mask data can be loaded from frame buffer
or from the data bus.
```

796-Bus Compatibility

```
D16 M20 VOL, 16-bit data only.
```

Electrical Characteristics

```
+ 5V +- 5%.  Maximum current: 4 A.
```

Physical Characteristics

```
Width: 12.00 in. (30.48 cm)
Height: 6.75 in. (17.15 cm)
Depth:  0.50 in. (1.27 cm)
Weight: 16 oz.   (447 g)
```

Environmental Characteristics

```
Operating Temperature: 0-50 C
```

2. Programming the Sun 1024 Video Board

This section provides detailed information about the functions and the registers of the Sun 1024 video board.

2.1. Coordinate System

The frame buffer allows the access of up to sixteen horizontally adjacent pixels in one access to the frame buffer. From one to 16 pixels can be read or written in a single cycle, starting at the current <x,y> position and the data bits within a word being left-justified. For example, a 4-bit update at location <200,300> will write the most significant four data bits (D15 through D12) into locations <200,300> through <203,300>.

2.2. Registers and Function Unit

Figure 2-1 shows the major functional components of the Sun video board. There are three data registers, *destination*, *source*, and *mask*, feeding into the function unit. There are four control registers controlling update and overall system operation: *function*, *width*, *status*, and *interrupt acknowledge*. There are four sets of <x,y> address registers for frame buffer addressing.
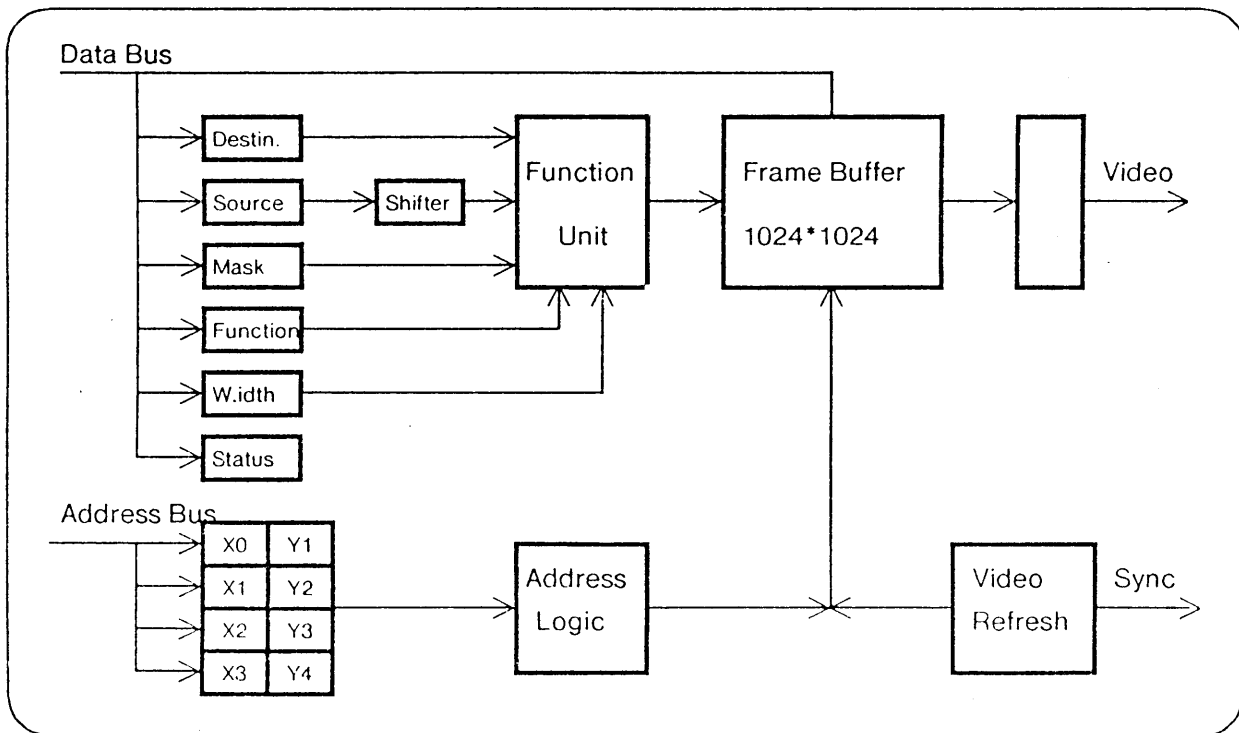

Figure 2-1: Sun Video Board Block Diagram

2.3. Data Registers

The three data registers: destination, source, and mask, hold the data participating in a frame buffer update operation. Data registers can be loaded on read or write cycles. On read cycles, they are loaded with data read from frame buffer memory, on write cycles they are loaded with the data from the processor.

2.3.1. Destination Register

The destination register holds the data that is being modified with a read-modify-write cycle on update operations in the frame buffer. The destination register can not be accessed directly from the processor.

2.3.2. Source Register

The source register holds data to be combined with the destination data and the mask (pattern) data to compose new data for the frame buffer. The source register can be loaded from the frame buffer or from the processor. The data in the source register is bit-wise aligned with the bit-address of the destination in the frame buffer.

2.3.3. Mask Register

Like the source register, the mask register holds data to be combined with the destination register and the source register to compose new data for the frame buffer. However, unlike the source register the mask register is not bit-aligned with the bit-address of the destination in the frame buffer. Instead, the mask register is aligned with 16-bit memory words (x mod 16 = 0) in the frame buffer. This way, the mask register can be used for background coloring and stipple-pattern generation where bit-alignment is undesirable.

2.4. Control Registers

The four control registers: function, width, status, and interrupt acknowledge, control the operation of the function unit and the interrupt mechanism. Control registers are loaded on write cycles with data from the processor.

2.4.1. Function Register

The function register specifies how the function unit combines destination, source, and mask data to form the data written into the frame buffer memory. The eight-bit contents of the function register selects one of the 256 possible RasterOps for three boolean operands. See section "RasterOps", 2.6, for further information.

2.4.2. Width Register

The width register controls the width of an RasterOp update operation. Frame buffer updates can be from 1 to 16 pixels wide. It is loaded from the low order 4 bits of the data bus, with 0 meaning 16, so its valid range is from 1 through 16. For RasterOps less than 16 wide, the high-order bits of the data in the source and mask registers are significant.

2.4.3. Status Register

The status register controls video enable, interrupt enable, interrupt level, and video board LED as follows, where bits are numbered from D15 (most significant) to D0 (least significant):

```
.............................
D13..D15   Interrupt Level
D11..12    Reserved
D10        LED Enable
D9         Video Enable
D8         Interrupt Enable
.............................
```

The Video Enable bit turns on video to the monitor; the screen appears blank when this bit is off. The Interrupt Enable bit enables interrupts on the level selected. When enabled, an interrupt is generated at the beginning of every vertical retrace, allowing synchronization of display updates with display refresh. The Interrupt flag stays pending until reset in software by accessing the interrupt acknowledge register (see below). The LED Enable bit turns the video board LED off when set; the LED lights when this bit is zero.

The status register is cleared (set to zeros) on INIT causing a blank screen, LED on, and disabled interrupts when the video board is powered up.

2.4.4. Interrupt Acknowledge Register

This location is a pseudoregister, not an actual register. Accessing this location clears a pending interrupt (resets the interrupt flag).

2.5. Address Registers

The host processor accesses graphical objects in the frame buffer via (x-y) register pairs. Four sets of (x-y) registers are provided that can be selected dynamically via the address bus. Only one (x) or (y) register is updated at any one time; the others do not change. The address registers are loaded from the address bus. On every access to the Sun 1024 video board, an address register is updated.

2.6. RasterOps

The Sun 1024 Video Board supports a three operand RasterOp in hardware. The RasterOp units operates on the pixels addressed by the active address register in conjunction with the width register.

During a RasterOp, the pixels accessed in the frame buffer are modified according to the function specified by the function register and as a result of data present in the data registers: destination, source, and mask (see the description of these registers for details).

There are 256 possible functions mapping three boolean operands into a boolean result. The frame buffer's eight-bit FUNCTION register selects one of these at a time by acting as a three-bits-in, one-bit-out lookup table for corresponding bits of the Destination, Source, and Mask. For example, suppose we want to set Destination equal to (Dst OR Src), ignoring the value of the mask. Consider the application of this function to a single pixel. The function may be expressed in tabular form as follows:

```
MSK      SRC      DSR      DST' = SRC OR DST
--------------------------------
0        0        0        0
0        0        1        1
0        1        0        1
0        1        1        1
1        0        0        0
1        0        1        1
1        1        0        1
1        1        1        1
--------------------------------
```

The MSK, SRC, and DST columns in the table form an index running from zero (000) through seven (111). The eight bits of the result column uniquely specify the desired boolean function, and these are precisely the eight bits which are to be loaded into the frame buffer's Function register. By convention, the least significant bit of the function appears at the top of the table, hence this function (Src OR Dst) is represented by the eight-bit value 11101110 (0xEE). Examples of other function encodings are 0x0 (clear destination bits), 0xFF (set destination bits), and 0xCC (copy source to destination).

The Sun video board allows all 256 possible RasterOp functions, although only a few are used in practice.

For example, to clear the entire screen, the constant function 0 is applied to the viewable rectangle. To flash (invert) a window, the function NOT DST is performed on that window. To write a character, the SRC function is used, while NOT SRC writes the character inverted (black on white). DST OR SRC overstrikes (paints) the character, and SRC OR MSK writes the character with a background pattern.

2.7. Video Board Operation

The Sun 1024 video board initiates operations in response to read and write commands from the Multibus.

The Sun video board uses both the data and the address lines of the Multibus to maximize the information sent to the video board in one bus cycle. The data lines always transfer data, but the address lines transfer both addressing information and command information, all in one bus cycle.

In a single cycle, the Multibus can transfer a new 16-bit data item via the data bus and a new (x) or (y) address via the address bus. At the same time, the Multibus can select which of the four sets of (x,y) registers to use, whether to load a register from the data bus or from the frame buffer, whether to load the source or the mask register, and whether to execute a frame buffer update or not. Refer to the next section for details on how this information is encoded on the Multibus.

The sequence of internal register transfers is somewhat complex and best explained in terms of an

example depicting a typical read or write cycle.

On a write cycle to the video board, five events occur sequentially.

1. One of the four sets of (x,y) cursor registers is selected.

2. An x or y coordinate encoded in the address is loaded into the (x) or (y) register of the selected pair.

3. Data from the Multibus is written into the selected register on the video board (source, mask, function, width, or control), or the data is ignored if no register is selected.

4. The contents of the addressed (x,y) frame buffer location are read into the destination register.

5. If and only if an update is requested, the data stored in the destination, source, and mask registers are combined according to the preselected function and new data is written back into the addressed frame buffer location.

On a read cycle from the video board, four things happen. The first two are the same as for the case of a write cycle.

1. One of the four sets of (x,y) cursor registers is selected.

2. An x or y coordinate encoded in the address is loaded into the (x) or (y) register of the selected pair.

3. Data is read from the addressed (x,y) location in the frame buffer into the selected register on the video board.

4. The data then stored in the source register is made available to the Multibus, correctly bit-aligned with the bus data lines.

Note that the frame buffer is never updated on a read cycle and the update bit is ignored for read cycles.

A one-deep FIFO decouples the video board from the Multibus. Multibus requests are latched on the video board and are subsequently executed independently and in parallel with other bus activity. This makes the frame buffer a zero-access-time device as long as the request rate does not exceed one request per microsecond. Since normally streams of data are being transferred, the pipelining maximizes throughput.

On write cycles, the FIFO operation is invisible to the host processor. However, on read cycles, due to the pipelining, the data read back corresponds to the previous read request. Thus, to read a stream of data, one additional word needs to be read before valid data is obtained.

2.8. Video Board Address Decoding

The video board decodes the 20 bits of the Multibus memory address lines as follows:

```
--------------------------------------------------------------------
            A0       0

            A1..10   New X or Y address

            A11      0 -> X,  1 -> Y

            A12..13 Selects one of the four sets of (x,y) registers

            A14..15 Select data register to be updated
                         0 -> No Register
                         1 -> Control Registers
                         2 -> Source Register
                         3 -> Mask Register

                    Control register are further decoded with A1..A2 as follows:
                         0 -> Function Register
                         1 -> Width Register
                         2 -> Control Registers
                         3 -> Interrupt Acknowledge


            A16      Enable raster operation on frame buffer
                         0 -> no update operation
                         1 -> execute update operation


            A17..19 Board Select
--------------------------------------------------------------------
```

A0: Byte Address. Since the Sun 1024 video board is wordorganized, this bit is always zero.

A1..A10: X-Y Address. These 10 bits constitute the new *x* or *y* address to be loaded into the address register selected by A11 through A13.

A11..A13: Address Register Select. There are four pairs of ten-bit (x,y) address registers, selected by bits A12 and A13. Bit A11 selects whether the X or Y register of the pair is updated. Every read or write access to the video board will load one of these address registers as a sideeffect.

A14..A15: Data Register Select. These two bits determine which data register (or none) is loaded as a result of the current operation. On *read* cycles, the selected register is loaded from data supplied by the frame buffer; on *write* cycles, with data from the host processor. "0" means that no register is loaded, "1" means that one of the control registers will be loaded, "2" selects the source register, and "3" selects the mask register. When a control register is specified, address bits A1 and A2 select the register: "0" is the function register, "1" is the width register, "2" is the status register, and "3" is the interrupt acknowledge register.

A16: Update Bit. The update bit enables modification of the frame buffer on write cycles. If the bit is off, registers can be loaded on write cycles without changing the frame buffer. If the Update bit is set, any registers affected is loaded first and then a RasterOp is performed. The Update Bit is valid only for write cycles and is ignored on read cycles.

A17..A19: Unit Number. The most significant three bits determine the base address of a particular video board on a 128K byte boundary. Each video board occupies 128K bytes of Multibus memory address space.

3. Preparation for Use

3.1. Introduction

This chapter provides information on installing the Sun 1024 Video Board. Included are instructions for unpacking, inspection, switch and jumper setting, and interfacing the Sun 1024 Video Board with other 796-Bus boards.

3.2. Unpacking Instructions

Inspect the shipping carton immediately upon receipt for evidence of damage. If the shipping carton is severely damaged, request that the carrier's agent be present when the carton is opened. If the carrier's agent is not present when the carton is opened and the contents are damaged, keep the content and carton for the agent's inspection.

It is suggested that salvageable shipping cartons and packing material be saved for future use in the event the product must be reshipped.

3.3. Installation Considerations

The board is designed for installation into a Intel Multibus or IEEE 796-Bus compatible backplane or cardcage. Make sure that the system provides adequate power and cooling for the Sun 1024 Video Board.

The Sun 1024 Video Board requires a 5V power supply and draws a maximum current of 4 Amps. When installing the board in an enclosed environment or under restricted airflow conditions, ensure that the internal operating temperature does not exceed 130 degrees F or 55 degrees C.

CAUTION: To prevent possible equipment damage, do not install board in a cardcage while power is on. Also, to prevent damage due to static voltages, avoid exposing the board to plastic materials.

3.4. Repair Information

To return a Sun 1024 video board for repair, obtain a return material authorization number (RMA) from the address below and send the board with the RMA number and a detailed description of the problem to the following address:

```
Sun Microsystems Inc
Att: Service Department
2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

415-960-1300
```

3.5. Switches and Jumpers

The Sun 1024 Video Board has only one user-selectable switch which selects the Multibus base address of the board. The video board occupies 128K Bytes in the Multibus memory address space. The board can be addressed on any one of the eight 128k Byte boundaries of the 1M Byte Multibus address space be means of switch selector U801 or the alternate Jumper selector J800.

```
-----------------------
BASE      SWITCH  JUMPER
ADDRESS   NUMBER  NUMBER
-----------------------
0K        8       15..16
128K      7       13..14
256K      6       11..12
384K      5       9..10
512K      4       7..8
640K      3       5..6
768K      2       3..4
896K      1       1..2
-----------------------
```

3.6. J1-Connector

The Sun 1024 Video Board connects to the video monitor with the J1 connector. It is suggested that shielded and twisted pair cable is used for the connection between board and monitor. The J1 pinout is as follows ("\" indicates active low signals):

```
PIN    NAME            PIN    NAME
------------------------------------
1      GND             2      GND
3      VIDEO\          4      GND
5      VIDEO           6      GND
7      HSYNC\          8      GND
9      VSYNC\          10     GND
------------------------------------
```