Sun-2 Architecture Manual

Draft Version 0.5

Company Confidential

Sun Microsystems Inc.

15 December 1983

This document specifies the Sun-2 Architecture. Implementation dependent features are also described.

Table of Contents

1. Introduction

This document is the specification of the Sun-2 Architecture. It is intended as a reference for Sun-2 software, hardware, and systems implementors.

The main part of this document is independent of a particular implementations of the Sun-2 architecture. Implementation specific data, as well as timing information, is described in an appendix for each implementation.

An important goal of this document is correctness. Please report any errors, omissions, or oversights immediately so they can be corrected in future revisions.

2. Layers

The Sun-2 architecture is divided into three layers: The CPU, the MMU, and the Device Layer. Each of these layers is independent of the others. The following sections give an overview of these three layers.

2.1. CPU Layer

The CPU layer of the Sun-2 architecture is based on the Motorola 68010 instruction-set-processor. The processor is extended with a number of external devices: a bus error register, a system enable register, a diagnostic register, and an ID-PROM.

The ID-PROM contains a unique serial number and configuration data for a particular implementation of the architecture, describing the actual configuration of the MMU and the I/O layer.

2.2. MMU Layer

The MMU layer of the Sun-2 architecture defines how the CPU accesses and manages system resources and provides the function of multiple virtual and physical address spaces, address translation, protection, and sharing.

2.3. Device Layer

The Device layer of the Sun-2 architecture defines what devices are supported under the architecture and how these devices communicate with the system. These devices include main memory, the system bus, I/O devices, and others.

3. Definitions

In the subsequent description of the Sun-2 architecture the following terms are used:

DVMA: Direct Virtual Memory Access

CPU: Central Processing Unit

MMU: Memory Management Unit

PMEG: Page Map Entry Group

RES: Reserved

POR: Power-On-Reset

4. CPU Layer

The CPU layer of the Sun-2 architecture is based on the 68010 instruction-set-processor. The processor is extended with an external bus error register, an external system enable register, a diagnostic register, and an ID-PROM.

4.1. Access to Devices in CPU layer

The devices of the CPU layer are accessed in a separate address space decoded with the reserved function code "3" as the source or destination function of a 68010 MOVS instruction. This retains the full virtual address space for supervisor and user processes. A similar mechanism is used to access the MMU layer.

Individual devices are selected by the low-order address bits as follows:

A10..A4	A3	A2	A1	DEVICE
	0	1	0	ID PROM
	0	1	0	1 DIAGNOSTIC REGISTER
	0	1	1	0 BUS ERROR REGISTER
	0	1	1	1 SYSTEM ENABLE REGISTER

4.2. ID PROM

The purpose of the ID PROM is to provide basic information on the machine type and a unique serial number for software licensing, distribution, and access. In addition, the ID PROM stores the Ethernet address, the date of manufacturing, and a checksum.

The ID PROM is implemented as a 32-byte PROM, mounted in a socket, so it can be swapped into replacement boards. The 32 bytes are mapped to consecutive pages as follows:

REGISTER	ADDRESS	SIZE	TYPE
ID PROM 0	0x0008	BYTE	READ-ONLY
ID PROM 1	0x0808	BYTE	READ-ONLY
ID PROM 2	0x1008	BYTE	READ-ONLY
...	...	...	...
ID PROM 31	0xF808	BYTE	READ-ONLY

The content of the ID PROM is as follows:

Entry	Field	
(1)	Format	1 Byte
(2)	Machine Type	1 Byte
(3)	Ethernet Address	6 Byte
(4)	Date	4 Byte
(5)	Serial Number	3 Byte
(6)	Checksum	1 Byte
(7)	Reserved	16 Byte

In detail:

(1) *Format*. The format of the ID PROM. 1 for now.

(2) *Machine Type*. A number specifying an implementation of the architecture.

(3) *Ethernet Address*. This is the unique 48-bit Ethernet address assigned by Sun to this machine.

(4) *Date*. The date the ID PROM was generated. It is in the form of a 32-bit long word which contains the number of seconds since January 1, 1970.

(5) *Serial Number*. This is a 3-byte serial number.

(6) *Checksum*. The checksum is defined such that the longitudinal XOR of the first 16 bytes of the PROM including the checksum yields 0.

(7) *Reserved*. This is reserved for future expansion.

4.3. Diagnostic Register

The diagnostic register drives an 8-bit LED display for displaying error messages. A "0" bit written will cause the corresponding LED to light up, a "1" bit to be dark. Upon power-on-reset, the diagnostic register is initialized to 0 causing all LEDs to light up. The no-fault state is defined to be all ones, with no LEDs light up.

```
Initialization: none
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
DIAGNOSTIC LED  0xA     BYTE    WRITE-ONLY
-----------------------------------------
```

4.4. Bus Error Register

When a bus error occurs, the bus error register latches its cause to allow software to identify the source of the bus error. The bus error register is a read-only register, and its content is not initialized or cleared upon reset.

```
Interrupt:      none
Initialization: software read of register
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
BUS ERROR       0xC     WORD    READ-ONLY
-----------------------------------------
```

The fields of the bus error registers are defined as follows:

```
BIT     NAME            MEANING
-----------------------------------------------------------
    D0  PARERRL         Parity Error Low Byte
    D1  PARERRU         Parity Error Upper Byte
    D2  TIMEOUT         Timeout Error
    D3  PROTERR         Protection Error
    D4  (reserved)
    D5  (reserved)
    D6  (reserved)
    D7  PAGEVALID       1 => Valid Page, 0 => invalid page
    D8  (reserved)
  ..D15 (reserved)
-----------------------------------------------------------
```

In more detail, the bus error conditions are as follows:

- Page invalid (PAGEVALID = 0) means that the page referenced did not have a valid bit set.

- Protection error (PROTERR) means that the page protection bits or the PAGEVALID bit did not allow the kind of operation attempted.

- Parity errors (PARERRL and PARERRU) can occur only on read cycles from on-board memory (page type 0). Since parity errors are detected too late in the cycle to abort the current cycle, they abort the following cycle instead. If the following cycle does not recognize bus errors (see below) then the parity error will abort the next cycle that does recognize bus errors. In any event, the address in which the 68010 receives the bus error is unrelated to the address of the parity error, which is not available.

- Timeout results from a non-completed reference. This can occur when accessing non-existant devices on cycles that utilize a positive handshaking mechanism. The bus error address can be used to determine which device did not respond.

No bus error can occur during CPU accesses to the MMU, during interrupt acknowledge cycles, and during supervisor program accesses in boot state.

4.5. System Enable Register

The System Enable Register enables system facilities, provides soft interrupts, and controls booting. The System Enable Register can be read and written under software control and is cleared on power up (hardware reset) and watchdog reset, but not upon 68010 reset. Bits are assigned as follows:

```
Interrupt:      level 1, 2, and 3
Initialization: cleared on power-up-reset
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
SYSTEM ENABLE   0xE     WORD    READ/WRITE
-----------------------------------------
```

The fields of the system enable register are as follows:

```
SYSTEM ENABLE REGISTER FIELDS
------------------------------------------------------------
D0      EN.PAR          Enable Parity Generation
D1      EN.INT1         Cause Interrupt on Level 1
D2      EN.INT2         Cause Interrupt on Level 2
D3      EN.INT3         Cause Interrupt on Level 3
D4      EN.PARERR       Enable Parity Error Checking
D5      EN.DVMA         Enable Direct Virtual Memory Access
D6      EN.INT          Enable all Interrupts
D7      BOOT*           Boot State (0 => boot, 1 => normal)
D8..D15 Reserved
------------------------------------------------------------
```

When cleared after power-up or watchdog reset, all bits are initialized to 0. In this state, boot state is active, parity generation and checking is disabled, DVMA, soft interrupts and all other interrupts are disabled.

4.6. Processor Operation

This section gives details of the CPU operation in the Sun-2 architecture.

4.6.1. Reset

Three types of reset need to be distinguished: Power-On Reset, Watchdog Reset, and 68010 Reset.

Power-On Reset. Power-On Reset (POR) is active for 100 milliseconds after the power supply voltage reaches 4.5V. POR resets the 68010 and clears the System Enable register forcing boot state, and it resets the diagnostic register, lighting all the LEDs.

Watchdog Reset. The Sun-2 architecture provides a watchdog circuit which generates a signal equivalent to power-on reset (POR) whenever the 68010 halts with a double bus fault. The result of a watchdog reset is identical to a POR, as far as the 68010 CPU is concerned.

68010 Reset. When the 68010 executes a reset instruction, it resets all on-board and off-board I/O devices that offer an external reset function. No other devices are affected. Devices of the CPU layer such as the system enable register and the diagnostic register are not affected by 68010 Reset.

4.6.2. Booting

Upon Power-on Reset or Watchdog Reset, the system enable register is cleared, forcing boot state active and disabling all interrupts and parity errors. Boot state forces all supervisor program fetches to access the onboard PROM device independent of the setting of the memory management. All other types of references are unaffected and will be mapped as during normal operation of the processor.

4.6.3. Interrupts

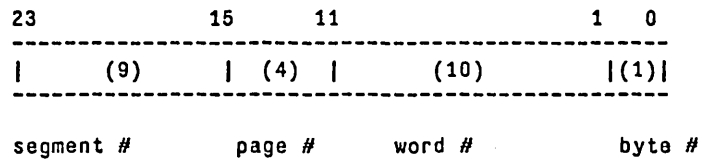
Interrupts are handled by the 68010 in autovector mode. Interrupts can be caused by various on-board and off-board devices, including the system enable register. The interrupt levels on which a device interrupts are described under each device.

5. Memory Management Unit

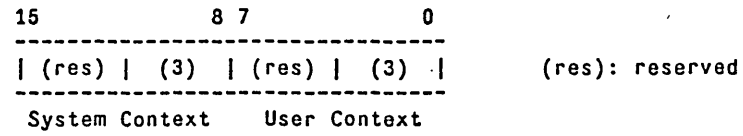
5.1. Summary

```
page size:              2 KBytes
segment size:           32 KBytes
process size:           16 MBytes
# of contexts:          8
# of segments/context:  512
# of pages/segment:     16
# of pmegs:             256
# of pages total:       4096
# of segments total:    4096
```

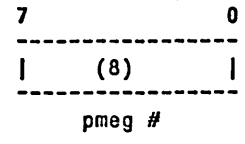
5.1.1. Virtual Address



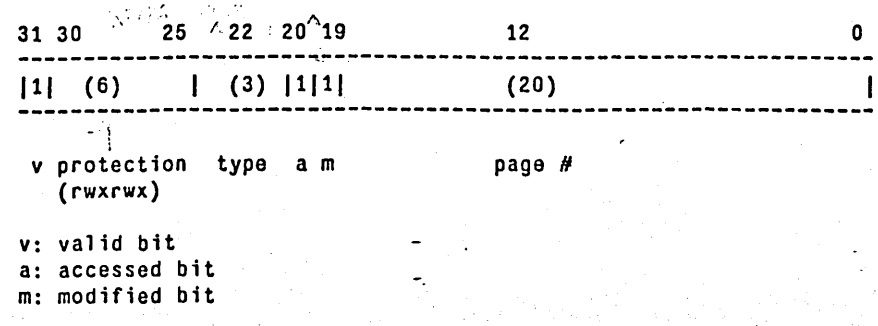
5.1.2. Context Register



5.1.3. Segment Map



5.1.4. Page Map



5.2. Access to MMU layer

Entries in the MMU are accessed with the same address they normally would translate, but in the address space decoded by the reserved function code "3" as the source or destination function of a MOVS instruction. The low-order address bits select which element of the MMU is modified. For accesses to the page map and segment map, the content of the user context register determines which context's map will be modified. Byte, word and long-word accesses to the maps are supported.

Thus, for user virtual address V, the map entries are accessed at location:

MAP	ADDRESS	SIZE	RELEVANT BITS
PAGE MAP	0 + V	LONG	V & 0xFFF800
SEGMENT MAP	4 + V	WORD	V & 0xFF0000 → FF8000
CONTEXT REG.	6 + V	WORD	V & 0x000000

5.3. MMU Overview

The Sun-2 Memory Management Unit provides address translation, protection, sharing, and memory allocation for multiple processes executing on the 68010 CPU. All CPU accesses to memory, on-board I/O, and to the system bus (P1-Bus) are translated and protected in an identical fashion. In addition, DVMA accesses by I/O channels also pass through the virtual memory management and thus operate in a fully protected environment.

The memory management consists of a context register, a segment map, and a page map. Virtual addresses from the processor are translated into intermediate addresses by the segment map and then into physical addresses by the page map.

The most important numbers for the memory management are a page size of 2048 bytes and a segment size of 32K bytes (giving 16 pages per segment). Up to 8 contexts can be mapped concurrently. The maximum virtual address space for each context is 16M bytes.

5.4. Contexts

The Sun-2 MMU is divided into 8 distinct address spaces or "contexts". The current context is selected by means of a 3-bit *context* register. To allow different address spaces for the supervisor and user, two alternate context values are provided. The MMU automatically uses the system context register whenever the 68010 issues a supervisor function code. The supervisor can address the user context via the 68010 MOVS instruction using a non-supervisor function code, by mapping the pages of interest into its own system context, or by sharing address space with the user by setting the two context values equal. The two context registers can be accessed as a word or separately accessed as the odd or even byte within a word. When read, the reserved bits are not defined.

5.5. Segment Map

The segment map has 4096 entries. It is indexed by the 9 most significant bits of the virtual address and 3 bits of the current context register. Thus, the segment map is divided into 8 sections of 512

entries each, with one section per context. Segment map entries are 8 bits wide, pointing to a page map entry group (*pmeg*).

5.6. Page Map

The page map contains 4096 page entries each mapping a 2K byte page. Page map entries are composed of a valid bit, protection field, type field, accessed and modified bits, and a page number.

The page map is divided into 256 sections of 16 entries each. Each section is pointed to by a segment map entry and is called a page map entry group, or *pmeg*.

5.6.1. Valid Bit

The valid bit determines whether a page map entry is valid or not. A valid bit of 1 means that the page map entry is valid and that the other fields of the page map entry determine how the reference is to be translated and protected. A valid bit of 0 means that an access to this page will be aborted, while the rest of the page map entry is ignored. In this case, the remaining bits of the page map entry may contain arbitrary information.

5.6.2. Protection Field

Access to pages can be controlled via the 6-bit protection field. From left (MSB) to right (LSB), the six bits correspond to "supervisor-read-write-execute" and "user-read-write-execute" privileges. This provides all 64 combinations of supervisor and user "rwxrwx". A "1" entry enables the corresponding capability, a "0" bit means that the respective capability is disabled.

5.6.3. Statistics Bits: Accessed and Modified

The accessed and modified bits are set, as the name implies, whenever a page is accessed or modified (written into). The statistics bits will not be updated when the page is invalid or when the protection code does not allow the attempted operation. In addition, these bits will not be updated in a cycle that aborts due to a parity error in the previous cycle. However, the statistics bits will be updated on all other cycles, including cycles that terminate due to timeout or cycles that cause parity errors.

5.6.4. Physical Address

The page map contains a 20-bit page number field. In conjunction with the 11-bit byte number, the page map thus can generate addresses up to 31-bit physical address bits. The architecture does not define, however, how many physical address bits are actually stored in the map, or how many physical address bits are implemented on the physical devices addressed. This specification depends on the implementation and is described in the implementation section.

5.6.5. PageType

The page type field provides for multiple physical address spaces, each starting at a physical address of 0. At the same time, the page type field decodes what busses and bus synchronization are used for a particular physical address space. The assignment of the page type field is described in the implementation section.

6. Device Layer

The device layers contains those elements of the system that are accessed through the memory management. This allows all devices to be protected, shared, and managed effectively in a multiprocess environment.

In the following, all the devices will be described. Each device listed has a brief description of its initialization, interrupts, reference, interrupts, access time, and register mapping.

```
REGISTER:
ADDRESS:
SIZE:
TYPE:
```

The assignment of devices to particular physical addresses and to particular page types is implementation dependent and is described in the implementation section for each machine type.

6.1. Main Memory

The main memory is the primary system storage. It has a size of up to 8 MBytes in increments of at least 512K Bytes. Main Memory is allocated consecutively starting from 0.

```
Interrupt:      none
Initialization: Parity needs to be initialized in software
Reference:      none
-------------------------------------------------
REGISTER        ADDRESS DATA    TYPE
-------------------------------------------------
WORD 0x000000   0x00000 WORD/BYTE       READ-WRITE
....
WORD 0x07FFFE   0x7FFFE WORD/BYTE       READ-WRITE
-------------------------------------------------
```

Parity is initialized by setting the "parity generation" bit in the system enable register and writing all of memory.

6.2. Monochrome Video Memory

The monochrome video memory is a dual-ported memory that provides video refresh and processor access on alternate cycles.

```
-------------------------------------------------
REGISTER        ADDRESS DATA            TYPE
-------------------------------------------------
WORD 0          0       WORD/BYTE       READ-WRITE
...
WORD 0x1FFFE    0x1FFFE WORD/BYTE       READ-WRITE
-------------------------------------------------
```

This memory is mapped to the display screen as follows:

Data bit 15 of Word 0 is the first visible pixel in the upper left corner of the display. Consecutive words are displayed along the horizontal scanline. After <display-width> number of pixels have been displayed, the next word is displayed at the beginning of the next horizontal line, up to <display-height> number of lines. <display-width> and <display-height> are implementation constants.

```
N = <display-width> / 16
M = <display-height>

 15            0 15            0 15            0  15            0
-------------------------------------------------------------------
| WORD 0        | WORD 1        | ...           | WORD N-1      |
-------------------------------------------------------------------
| WORD N        | WORD N+1      | ...           | WORD 2*N-1    |
-------------------------------------------------------------------
| WORD 2*N      | WORD 2*N+1    | ...           | WORD 2*N-1    |
-------------------------------------------------------------------
|  ...          |   ...         | ...           |    ...        |
-------------------------------------------------------------------
| WORD (M-1)*N  |   ...         | ...           | WORD (M-1)*N-1|
-------------------------------------------------------------------
```

The frame buffer can be updated in two ways. First, it can be read and written directly like memory. As such, it is visible as a 128 KByte block of 16-bit words. Second, the frame buffer can be written in copy mode as a side-effect of writing into main memory. This is achieved by selecting a base address and setting the copy enable bit in the video control register. The base address selects a 128K region of main memory. Data written into this selected region is also written into the frame buffer at the same offset within the 128K region.

6.3. Video Control Register

The video control register determines the operation of the video memory. It has the following fields:

```
Initialization: cleared on reset
Interrupt:      Level 4
-----------------------------------------------------------
REGISTER                ADDRESS DATA            TYPE
-----------------------------------------------------------
VIDEO CONTROL REGISTER  0       WORD/BYTE       READ-WRITE
-----------------------------------------------------------
BIT     NAME    MEANING
-----------------------------------------------------------
D0      RES     Reserved
D1..7   BASE    Copy memory base address A17..A23
D8..10  RES     Reserved
D11     AUDIO   Audio Enable
D12     INT     Interrupt Pending (read-only)
D13     INTEN   Interrupt Enable
D14     COPYEN  Copy Enable
D15     DISPEN  Display Enable
-----------------------------------------------------------
```

Base selects the base address for the copy update mode. The seven bits of the base address correspond to physical address bits A17 through A23. If the value of base matches the corresponding physical address bits during a write operation and copy enable is active, then a copy of the write data is stored in the frame buffer at the physical address modulo 128 KByte.

Audio Enable enables the sound generator.

Interrupt Pending indicates that a video interrupt has occured. When enabled, it interrupts the CPU on level 4. Video interrupt is set at the beginning of vertical retrace, that is, when the scanning of a display field just completed. The interrupt is cleared by momentarily turning off the interrupt enable bit.

Interrupt Enable allows video interrupts as described above.

Copy Enable enables the copy update mode to the frame buffer memory.

Display Enable turns on the video signal to the video monitor.

6.4. EPROM

Device EPROM is a pair of 28-pin sockets for 64K, 128K, or 256K EPROMs. Since the EPROMs are larger than a single 2K page, they are addressed directly with the low-order bits of the non-translated or virtual addresses from the CPU, even though they are enabled through a sequence of pages in the page map. Thus a group of pages with the EPROM page map entry will map to different sections of the EPROM.

```
Reference:      none
Interrupt:      none
Initialization: none
-----------------------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------------------
WORD 0          0       WORD    READ-ONLY
WORD 1          2       WORD    READ-ONLY
....
WORD 8192       16384   WORD    READ-ONLY       (2764s)
WORD 16384      32768   WORD    READ-ONLY       (27128s)
WORD 32768      65536   WORD    READ-ONLY       (27256s)
-----------------------------------------------------
```

The EPROM device also contains the boot code. In boot state, all supervisor program fetches are forced to fetch from the EPROM device, independent of the setting of the memory management.

6.5. Parallel Port

The parallel port is a non-latching 16-bit input port. Since the input data is non-latched, the data may change in the moment of being read. For best results, the data should be reread until stable data is obtained.

```
Interrupt:      none
Initialization: none
Reference:      none
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
INPUT PORT      0       WORD    READ-ONLY
-----------------------------------------
```

6.6. Serial Port

Serial ports are implemented with the Zilog 8530 SCC (serial communication controller). The SCC features two high-speed, fully symmetrical and highly programmable serial channels with built-in baud-rate generators. The clock input to the SCC is a 4.9152 MHz clock, independent of the CPU clock.

The SCC is mapped as follows:

```
Interrupt:      Level 6
Initialization: Needs to be initialized in software
Reference:      Zilog 8530 SCC data sheet
Recovery Time:  1.6 microseconds
-------------------------------------------------
REGISTER        ADDRESS DATA    TYPE
-------------------------------------------------
CH B CONTROL    0       BYTE    READ/WRITE
CH B DATA       2       BYTE    READ/WRITE
CH A CONTROL    4       BYTE    READ/WRITE
CH A DATA       6       BYTE    READ/WRITE
-------------------------------------------------
```

6.7. Keyboard/Mouse UART

These serial ports are implemented with the Zilog 8530 SCC (serial communication controller). The SCC features two high-speed, fully symmetrical and highly programmable serial channels with built-in baud-rate generators. The clock input to the SCCs is a 4.9152 MHz clock, independent of the CPU clock. Control lines are not used.

The SCC is mapped as follows:

```
Interrupt:      Level 6
Initialization: Needs to be initialized in software
Reference:      Zilog 8530 SCC data sheet
Recovery Time:  1.6 microseconds
-------------------------------------------------
REGISTER        ADDRESS DATA    TYPE
-------------------------------------------------
CH B CONTROL    0       BYTE    READ/WRITE
CH B DATA       2       BYTE    READ/WRITE
CH A CONTROL    4       BYTE    READ/WRITE
CH A DATA       6       BYTE    READ/WRITE
-------------------------------------------------
```

6.8. Timer

An AMD 9513 timer chip with five 16-bit timers is provided. The clock input to the 9513 is a 4.9152 MHz clock, independent of the CPU clock. Gate inputs 4 and 5 are driven by supervisor state. Gate input 1 is wired to FOUT. The timer is mapped as follows:

```
Interrupt:      Level 7 for Timer 1, Level 5 for Timer 2 through 5.
Initialization: Internal Reset whenever power supply drops below 3.0V.
Reference:      AMD 9513 programming book
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
TIMER DATA      0       WORD    READ/WRITE
TIMER COMMAND   2       WORD    READ/WRITE
-----------------------------------------
```

Note the synchronization requirements of the 9513 timer. Before writing into a counter, the counter's clock source must be disabled first.

Initialization of the 9513 timer is special in that the chip has an on-chip power-on reset that initializes the chip whenever the power supply voltage is less than 3V. The chip is not affected by power-on resets, watchdog resets, or 68010 resets.

6.9. RasterOp Processor

The RasterOp Processor is a device that performs bitblts or rasterops on main memory.

```
Interrupt:      none
Initialization: none
Reference:      SUN RasterOp Processor data sheet
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
REGISTER        0..30   WORD    READ/WRITE
-----------------------------------------
```

The RasterOp processor is only active in user state and when the RasterOp bit is set, which is done by writing data bit 15 at address 30 in the RasterOp Processor register set. The RasterOp processor is automatically disabled in system state, independent of the setting of the RasterOp bit.

When active, the RasterOp processor converts CPU write-cycles to memory into read-modify-write cycles consisting of three steps. In the first step, the CPU write data is loaded into the source register of the RasterOp processor. In the second step, the old data from the memory being written into is loaded into the RasterOp processor destination register. In the third step, the data from the RasterOp function unit is written into the memory, replacing the old data previously present.

6.10. Encryption Processor

The Encryption processor is an AMD 9518/8068 data ciphering processor providing high-speed NBS DES encryption. To access an internal register in the 9518/8068, the address register must be written first. Once the address register is setup, the selected register can be accessed repeatedly.

```
Initialization: none
Interrupts:     none
Reference:      AMD 9518/8068 data sheet.
Recovery Time:  1.6 microseconds
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
DATA REGISTER   0       BYTE    READ/WRITE
ADDRESS REG.    2       BYTE    WRITE-ONLY
-----------------------------------------
```

6.11. Real Time Clock

The Real-Time Clock maintains time of day and a calendar. A battery powers the clock when the main power is off. The real-time clock is based on the National 58167 chip which is addressed as 32 byte locations.

```
Initialization: none
Interrupts:     none
Reference:      National Semiconductor 58167 data sheet.
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
REGISTER 0      0       BYTE    READ/WRITE
REGISTER 1      2       BYTE    READ/WRITE
...
REGISTER 31     62      BYTE    READ/WRITE
-----------------------------------------
```

6.12. Sound Generator

The sound generator is based on the Texas Instruments 76489. It is addressed as a single register and is write-only. The audio output is enabled by the audio enable bit in the video control register.

```
Initialization: none
Interrupts:     none
Reference:      Texas Instruments 76489 data sheet.
Recovery Time:  8 microseconds
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
SOUND CHIP      0       BYTE    WRITE-ONLY
-----------------------------------------
```

The sound generator requires a write recovery time of 8 microseconds. That means that it can only be written once every 8 microseconds. Writes within the 8 microsecond period are ignored.

deleted featwe 12/20/83

6.13. Ethernet Interface

The Ethernet Interface is based on the Intel 82586 chip. Configured in maximum mode, the 82586 can directly address 24-bits of virtual memory. When the 82586 becomes active, it performs data read-write operations in supervisor mode.

The 82586 is connected to the system in a permanent byte-reversed mode, i.e. 82586 bits 0 through 7 are connected to 68000 bits 8 through 15 and vice versa. This causes Ethernet data to be stored in memory in 68010 byte order, whereas 82586 control blocks in memory are byte swapped.

Overall operation of the Ethernet Interface is controlled by the Ethernet control register that can be read or written in the Ethernet page.

```
Initialization: cleared on all resets
Interrupts:     Level 3
Reference:      Intel 82586 data sheet.
-----------------------------------------
REGISTER        ADDRESS DATA    TYPE
-----------------------------------------
CONTROL REG.    0       BYTE    READ/WRITE
-----------------------------------------
```

The fields of the Ethernet control register are assigned as follows:

```
ETHERNET CONTROL REGISTER FIELDS
-------------------------------------------------------
    D0  INT     Interrupt Pending (Read-Only)
    D1  ERR     Error Pending (Read-Only)
    D2  RES     Reserved
    D3  RES     Reserved
    D4  INTEN   Interrupt Enable
    D5  CA      Channel Attention
    D6  LOOPB*  0 => Loopback, 1 => Normal Operation
    D7  RESET*  0 => Ethernet Reset, 1 => Normal Operation
-------------------------------------------------------
```

INT signals Interrupt from the 82586. *ERR* indicates that a Bus Error occured during an 82586 channel operation, inhibiting further channel activity. To reset the *ERR* condition, the *RESET* bit in the Ethernet control register must be activated. *INTEN* enables 82586 interrupts to the CPU. *CA* signals channel attention to the 82586. *LOOPB** controls whether the front-end encoder/decoder is configured in loopback mode (LOOPB* = 0) or connected to the transceiver cable (LOOPB* = 1). *RESET* initializes the 82586 when active (RESET* = 0) and allows normal operation when inactive (RESET* = 1). It also clears the *ERR* condition when active.

7. DVMA: Direct Virtual Memory Access

Implementations of the Sun-2 may provide DVMA (Direct Virtual Memory Access) capability. DVMA allows P1-Bus masters to directly access onboard memory with a virtual address. Direct virtual memory access avoids the dual mapping problems of DMA (direct memory access) in a virtual memory environment. DVMA translates and protects all accesses within the system in an identical fashion. DVMA can be enabled or disabled under software control via the DVMA enable bit in the system register.

DVMA is implemented according to the following specification:

Protection. Protection applies to DVMA the same way as to the 68010. DVMA cycles use the "supervisor data" function code. Thus the supervisor read or write capability in the page map has to be enabled to allow the corresponding type of access. If the respective capability is not set, the attempted DVMA cycle is aborted.

Parity Errors. DVMA read cycles that cause a parity error are aborted.

Statistic Bits. The update and modify bits are set on DVMA cycles that execute sucessfully the same way as on 68010 cycles. Cycles aborted due to protection error do not change these bits.

Error Handling. DVMA cycles can fail because of two reasons: protection error and parity error. An error is signalled to the controlling P1-Bus master in two ways. First, by asserting the bus error line, if such a mechanism exists, and second, by inhibiting the P1-Bus data transfer acknowledge (DTACK). The latter is interpreted by the P1-Bus master as an access to a nonexistent device and will cause a timeout exception. To allow the second type of error handling, DVMA masters must support a timeout mechanism. It is recommended that the timeout period be as short as possible because on-board operation cannot proceed until the DVMA master ends the bus cycle. In order to guarantee real-time response, DVMA masters should provide timeout periods in the range of tens of microseconds.

Selfreference. Two types of self-reference are possible. First, accesses from the main processor to the DVMA space on the P1-Bus. This type of access will not complete and timeout, unless another device exists at that address. Second, accesses from the P1-Bus through the memory management back to the P1-Bus. The result of this transfer is implementation dependent, but it will not perform a valid data transfer.

Deadlock. Since P1-Bus mastership is arbitrated independently from the 68010 bus mastership, the possibility exists that the 68010 is attempting to access the P1-Bus while the P1-Bus is attempting to access the on-board bus. If this condition occurs, deadlock is avoided by suspending the 68010 bus cycle and yielding the on-board bus to the P1-Bus DVMA cycle. The 68010 cycle will resume when the P1-Bus DVMA cycle is completed.

8. Implementation Information for Machine Type 1

Machine Type 1 is the Multibus or IEEE-796 Bus implementation of the architecture.

8.1. MMU Implementation

The Version-1 Architecture implements a page number field that stores 12 bits. It thus supports a physical address of 23 bits, capable of addressing 8 MBytes. The other physical address bits in the page map are not implemented. When read, the not implemented bits are not defined.

8.2. Physical Address Assignments

The decoding of the page type field is described in the table below, together with the number of address bits the page types use or decode.

Type	Address	Device	Wait States
0	23-bit	On-Board RAM	
	[0x000000]	Physical Memory 1..4 MBytes	0
	[0x700000]	BW-Frame Buffer	0 (Write), 4..8 (Read)
	[0x780000]	Keyboard/Mouse UART	0 (Write), 4..8 (Read)
	[0x780800]	Sound Generator	0 (Write)
	[0x781800]	Video Control Register	0 (Write), 4..8 (Read)
1	14-bit	On-board I/O	
	[0x000000]	EPROM	1
	[0x000800]	RESERVED	1
	[0x001000]	ENCRYPTION PROCESSOR	1..5
	[0x001800]	PARALLEL PORT	1
	[0x002000]	SERIAL PORT	1
	[0x002800]	TIMER	1
	[0x003000]	RASTEROP PROCESSOR	1
	[0x003800]	REAL-TIME CLOCK	4..8
2	20-bit	P1-Bus Memory	
	[0x000000]	0..1 MByte 796-Bus Memory Space 2 + device access time	
3	20-bit	P1-Bus I/O	
	[0x000000]	0..1-MByte 796-Bus Memory Space 2 + device access time	

8.3. Interrupt Assignments

The following table summarizes the interrupt level assignments for the devices that have been described in this manual. In addition, the P1-Bus can cause interrupts on all levels.

```
-----------------------------------------------------
    7   TIMER1
    6   Serial Port
    5   TIMER2..5
    4   VIDEO
    3   System enable register EN.INT3
    2   System enable register EN.INT2
    1   System enable register EN.INT1
-----------------------------------------------------
```

8.4. DVMA Implementation

Mapping. DVMA responds to the low-order 256K Byte memory space on the P1-Bus. This space is mapped to the low-order 256K Byte of the top 1 MByte of the system context virtual address space.

```
P1-Address                  Virtual Address
-----------------------------------------
[0x00000..0x3FFFF]          [0xF00000..0xF3FFFF]
```

DVMA Cycles proceed in supervisor context and supervisor function code. Only transfers to the main memory device are allowed.

8.5. CPU Timing

This basic timing of the CPU in this implementation is as follows:

```
CPU clock cycle:        80 nanoseconds (12.5 MHz)
CPU basic cycle:        320 nanoseconds
Timeout period:         5 microseconds
```

8.6. P1-Bus Access Times

This section describes the access times of the P1-Bus. P1-Bus I/O devices are identically to P1-Bus Memory except that they are located in a separate address space.

The timing of P1-Bus accesses depends on two factors: the access time of the P1-Bus device and the cost of P1-Bus aquisition if the Sun-2 Processor currently does not own P1-Bus mastership. In addition, some P1-Bus devices might have slower cycle times than access times which will increase effective access time in cycle-time limited transfers.

The total number of wait states for a P1-Bus access can be computed by the following formula:

2 WS (overhead) + 3 WS (if board is currently not P1-Bus master) + access time of P1-Bus device divided by the clock period of the CPU rounded up to the nearest integer number.

8.7. DVMA Access Time

DVMA cycles from the P1-Bus are serviced after the current CPU cycle completes and after pending memory refresh cycles are executed. Thus DVMA cycles exhibit a variable access time that

ranges from 0.7 microseconds in the best case to 1.5 microseconds worst case with an average of about 1.0 microseconds.

After a DVMA cycle has executed, a CPU cycle will start before another DVMA cycle is granted. This means that the cycle time for DVMA is one DVMA cycle plus at least one CPU cycle. Thus the DVMA cycle time will be in a range of 1.1 to 1.9 microseconds with an average of 1.4 microseconds, as long as the DVMA master can generate transfers at this rate.

8.8. P1-Bus Reset

P1-Bus Reset is the reset to the system bus. Power-On Reset, Watchdog Reset, and 68010 Reset will all assert P1-Bus Reset. Other P1-Bus devices may also assert P1-Bus Reset, but this will have no effect on the Sun-2 Board.

8.9. Video Memory

Video Memory is implemented as follows:

```
-----------------------------------------------------
<display-width> 1152 pixels
<display-height>900 lines
<display-depth> 1 pixel
-----------------------------------------------------
```

Write accesses to the video memory are buffered. Thus a single write will complete without wait states. A subsequent operation, whether read or write, will have to wait until the frame buffer has completed the requested operation. Write accesses to the video memory via the copy mode will cause the same behavior as direct write accesses. Read accesses to the video memory are not buffered and must wait until the cycle completes.

2. User Guide

2.1. Programming

The 2050 Board implements the Sun-2 Architecture, Machine Type 2. The full architecture is documented in the Sun-2 Architecture Manual and no attempt is made to repeat this information here. However, this section does describe the features specific to this implementation of the architecture.

2.1.1. MMU Implementation

The MMU of this machine type implements a page number field of 12 bits. It thus supports a physical address of 23 bits, capable of addressing 8 MBytes. The other physical address bits in the page map are not implemented. On a read cycle, the not implemented bits read back as 0.

2.1.2. Physical Address Assignments

Type	Address	Device	Wait States
0	23-bit	Memory Bus	
	[0x000000]	Physical Memory 1..8 MBytes	0
1	23-bit	I/O Bus	
	[0x000000]	BW-Frame Buffer	2 (Write), 4..8 (Read)
	[0x040000]	Video Control Register	2
	[0x7F0000]	EPROM	2
	[0x7F0800]	Ethernet Interface	2
	[0x7F1000]	Encryption Processor	2..8
	[0x7F1800]	Keyboard/Mouse Interface	2
	[0x7F2000]	Serial Port	2
	[0x7F2800]	Timer	2
	[0x7F3000]	RasterOp Processor	2
	[0x7F3800]	Reserved	2
2	23-bit	P1-Bus or System Bus	
	[0x000000]	0..8 MByte VME 24-bit address	1 + device access time
3	23-bit	P1-Bus or System Bus	
	[0x000000]	8..16 MByte VME 24-bit address	1 + device access time
	[0x7F0000]	64 KByte VME 16-bit address	1 + device access time

Accesses to the VME Bus incur an additional 2 wait states access time if the 2050 board is not currently bus master.

9. Implementation Information for Machine Type 2

Machine Type 2 is the VME-Bus/Eurocard implementation of the architecture.

9.1. MMU Implementation

The MMU of this machine type implements a page number field of 12 bits. It thus supports a physical address of 23 bits, capable of addressing 8 MBytes. The other physical address bits in the page map are not implemented. When read, the not implemented bits are not defined.

9.2. Physical Address Assignments

Type	Address	Device	Wait States
0	23-bit	Memory Bus	
	[0x000000]	Physical Memory 1..8 MBytes	0
1	23-bit	I/O Bus	
	[0x080000]	EPROM	2
	[0x080800]	Ethernet Interface	2
	[0x081000]	Encryption Processor	2..5
	[0x081800]	Keyboard/Mouse Interface	2
	[0x082000]	Serial Port	2
	[0x082800]	Timer	2
	[0x083000]	Reserved	2
	[0x083800]	Reserved	2
	[0x700000]	BW-Frame Buffer	1 (Write), 4..10 (Read)
	[0x720000]	Video Control Register	1 (Write), 4..10 (Read)
2	23-bit	P1-Bus or System Bus	
	[0x000000]	0..8 MByte VME Bus Space	1 + device access time
3	23-bit	P1-Bus or System Bus	
	[0x000000]	8..16 MByte VME Bus Space	1 + device access time

9.3. Interrupt Assignments

The following table summarizes the interrupt level assignments for the devices that have been described in this manual. All these interrupts are autovectored.

7	TIMER1
6	Serial Port
5	TIMER2..5
4	VIDEO
3	Ethernet or system enable register EN.INT3
2	System enable register EN.INT2
1	System enable register EN.INT1

In addition, the VME-Bus can cause vectored interrupts on all levels. Individual VME-Bus interrupt levels can be disabled with jumpers.

9.4. DVMA Implementation

Mapping. DVMA responds to the most significant 1 MByte of the VME-Bus physical address space. This space is mapped to the most-significant 1 MBytes of the system context virtual address space.

```
P1-Address                Virtual Address
-----------------------------------------
[0xF00000..0xFFFFFF]      [0xF00000..0xFFFFFF]
```

9.5. Video Memory

Video Memory is implemented as follows:

```
-----------------------------------------------------
<display-width> 1152 pixels
<display-height>900 lines
<display-depth> 1 pixel
-----------------------------------------------------
```

Read accesses are unbuffered and will cause 4 to 8 wait states. Write accesses to the video memory are buffered. However, subsequent read or write accesses will have to wait until the video memory has completed the requested operation. Write accesses to the video memory via the copy mode will cause the same behavior as direct write accesses.

9.6. CPU Timing

This basic timing of the CPU in this implementation is as follows:

```
CPU clock cycle:        80 nanoseconds (12.5 MHz)
CPU basic cycle:        320 nanoseconds
Timeout period:         5 microseconds
```

9.7. P1-Bus Access Times

This section describes the access times of the P1-Bus. P1-Bus I/O devices are identically to P1-Bus Memory except that they are located in a separate address space.

The timing of P1-Bus accesses is comprised of three elements: one, overhead; two, the cost of P1-Bus aquisition in case the Sun-2 Processor is not currently P1-Bus master, and three, the actual access time of the P1-Bus device.

The total number of wait states for a P1-Bus access can be computed by the following formula:

1 WS (overhead) + 2 WS (bus aquisition time if bus is idle) + access time of P1-Bus device divided by the clock period of the CPU rounded up to the nearest integer number.

9.8. DVMA Access Time

DVMA cycles from the P1-Bus are serviced after the current CPU cycle completes and after pending memory refresh cycles are executed. Thus DVMA cycles exhibit a variable access time that ranges from 0.7 microseconds in the best case to 1.5 microseconds worst case with an average of about 1.0 microseconds.

After a DVMA cycle has executed, a CPU cycle will start before another DVMA cycle is granted. This means that the cycle time for DVMA is one DVMA cycle plus at least one CPU cycle. Thus the DVMA cycle time will be in a range of 1.1 to 1.9 microseconds with an average of 1.4 microseconds, as long as the DVMA master can generate transfers at this rate.

9.9. P1-Bus Reset

P1-Bus Reset is the reset to the system bus. Power-On Reset, Watchdog Reset, and 68010 Reset will all assert P1-Bus Reset. Other P1-Bus devices may also assert P1-Bus Reset, but this will have no effect on the Sun-2 Board.