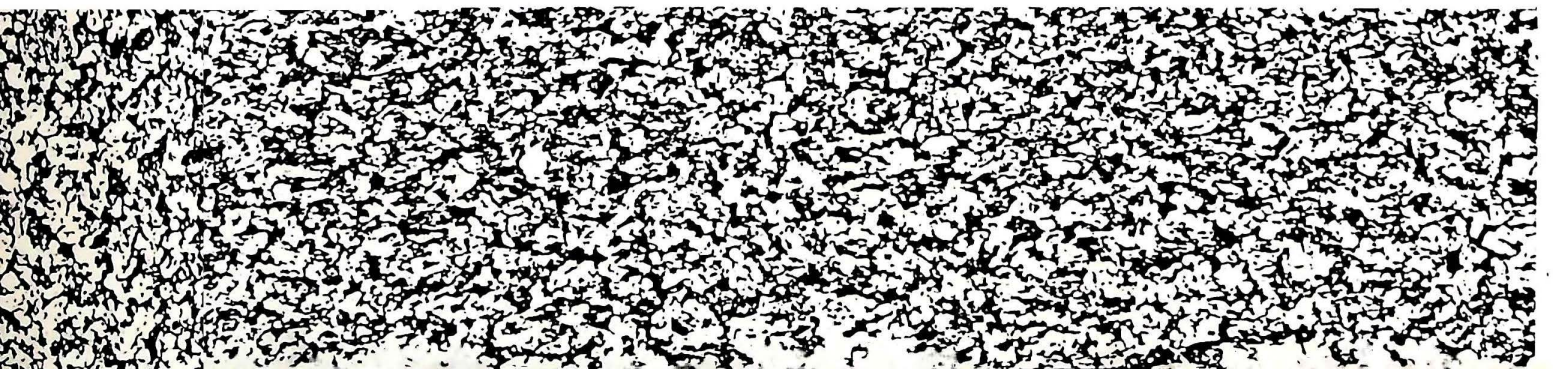
VME-68K30
Hardware Reference Manual

An NBI
Company

Integrated Solutions



VME-68K30
Hardware Reference Manual

Integrated Solutions
1140 Ringwood Court
San Jose, CA 95131
(408) 943-1902

490247 Rev. A

November 1988

PREFACE

This manual describes the Integrated Solutions VME-68K30 processor board. The text includes a product overview, specifications, and configuration and programming information. The manual has six sections, the content of which are briefly described in the following.

Section 1 describes the features and architecture of the VME-68K30 from both the single end user and the OEM point of view. Hardware features are described in terms of the functionality and operation of on-board modules, such as the VDMA and VATM addressing maps, the VME and PVSB bus interfaces, and the OCTART communications port. Special attention is given to the on-chip and on-board caches, since the CPU cache is a new feature in the ISI product line.

No details are given about PAL logic, timing diagrams, and board level signals. These must be obtained separately from engineering schematics and timing diagrams from ISI.

Section 2 describes the VSB (PVSB) developed by ISI for VME-68K30 access to high speed memory. Information about configuring the memory board with jumper and switch settings is contained in this section. This information is applicable to 4, 8, and 16 MB versions of the board.

Section 3 describes VME-68K30 addressing space which includes both VMEbus space and PVSB space.

Section 4 describes ISI defined space on the PVSB. On-board devices are addressed through this space. Their location and register descriptions to the bit level are described in this section along with programming information.

Section 5 describes jumper and switch settings used to configure the VME-68K30. Pin assignments for all connectors are also described.

Section 6 describes the PROM monitor and diagnostics.

Table of Contents

List of Figures

List of Tables

Section 1: Introduction

1.1 VME-68K30 Features

The VME-68K30 is a double-wide, VME based CPU card with a high-performance MC68030 CPU. The VME-68K30 implements the following features in the latest VLSI technology:

- 25MHz CPU Operation
- Supports 64 Mbytes of high speed memory through a private VME Subsystem Bus (PVSB)
- VMEbus compatible with full implementation of all functional modules
- Configurable for single or multiprocessor environments
- Floating point coprocessor strapped to microprocessor speed

1.2 Performance Comparisons

Table 1-1 compares the read/write speed of a VME-68K30 with its MC68030 microprocessor to the speed of other ISI boards that use a MC68020 processor. The improved access time of the MC68030 with two internal caches and a write pipeline is further enhanced by an external, 64 Kbyte cache on the VME-68K30. The large external cache allows the K30 to continuously outperform earlier CPU boards even when multiple processes and rapid context switching are normal system parameters.

As Table 1-1 shows, the write pipeline reduces sequential write time by more than 50 percent. A single write takes 3 clocks (120ηs) but a subsequent pipelined write requires only one additional clock for a 140ηs average.

If a read operand is found in cache, the K30 will fetch it in less than half the time that other boards require to go to memory. When a read operand is not found in cache, performance is slightly degraded, but the large, external cache reduces the need for frequent memory accesses under almost all operating conditions. The table is included only to show the how the pipeline and caches can affect access time. No conclusions about improved system performance or efficiency should be attempted without complete system analysis.

Table 1-1. Comparison of MC68030 with MC68020

Board	Processor	MHz	Write1	Write2	Read	Cache Read Hit	Cache Read Miss	Non-Cache Read
VME-68K20	MC68020	16.67	180ηs	360ηs	240ηs			
VME-68225	MC68020	25	160ηs	320ηs	200ηs			
VME-68K30	MC68030	25	120ηs	140ηs		80ηs	280ηs	280ηs

1.3 System Level Overview

The VME-68K30 is configurable for either single processor or multiprocessor operation, and can be used as either a cluster or server node in cluster installations. Any cluster in an installation has access to both its own high-speed memory and to the high speed memory of all other clusters.

The state-of-the art MC68030 gives the VME-68K30 board its processing power. The true functionality of the board in multiprocessor systems derives from its dual interface to the global VMEbus and its own PVSB. The PVSB is a high speed bus used only for memory access. Figure 1-1 shows the relation between these two buses and the processor board.

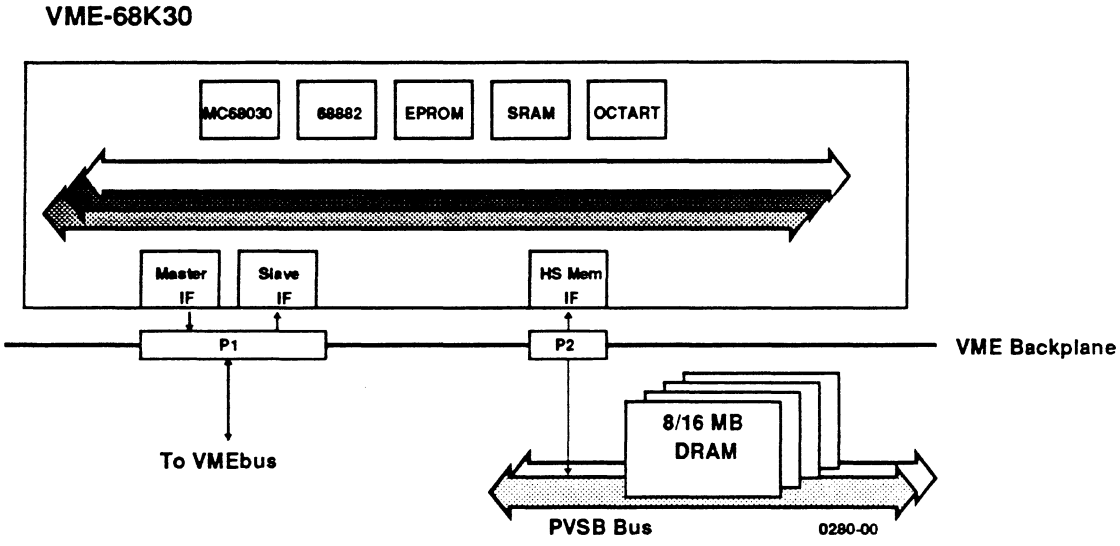


Figure 1-1. VME-68K30 on VMEbus and PVSB

1.4 VME-68K30 Hardware Overview

The basic hardware units on the VME-68K30 include:

- A MC68030 CPU
- A MC68882 floating point coprocessor
- A 64 Kbyte on-board cache
- Memory access through a high Performance VMEbus Subsystem Bus (PVSB)
- A global VMEbus address translation map (VATM)
- Socket for variable sized PROM (128 Kbytes standard)
- A single IC provides eight RS-232-C serial ports and four, 16-bit counter/timers
- A 2 Kbyte static RAM (SRAM)
- A Battery backed real-time clock
- A word swap buffer that accommodates 16 and 32-bit VMEbus ports

- Soft and hard resets
- Six LED indicators identify accesses and diagnostic failures

A block diagram showing the major hardware components is shown in Figure 1-2.

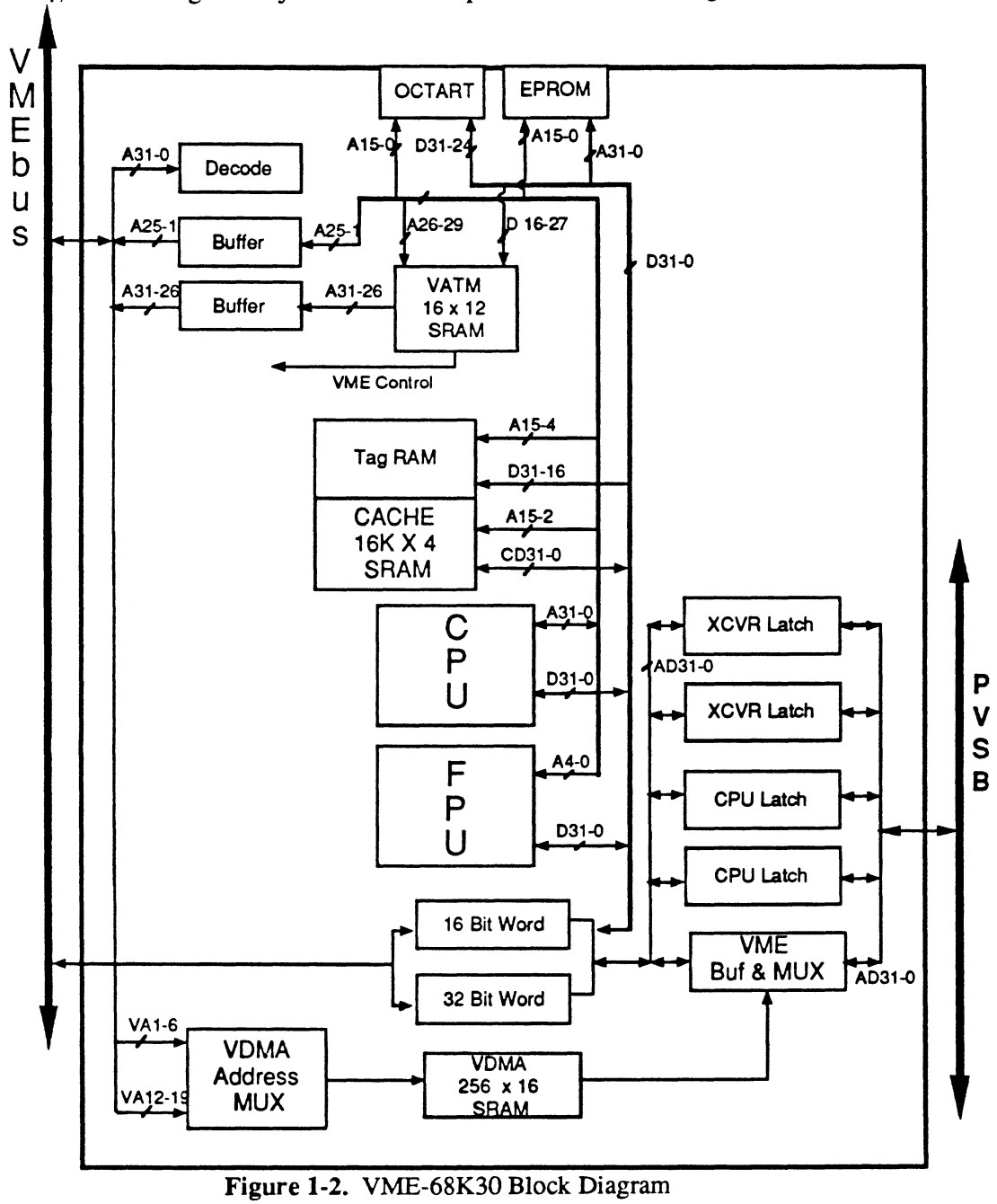


Figure 1-2. VME-68K30 Block Diagram

1.4.1 MC68030 CPU

The MC68030 CPU is capable of processing six million instructions per second (6 MIPS). It is the most advanced of Motorola's 68000 microprocessor series and is code compatible with

previous versions.

The following list briefly summarizes the MC68030 features:

1. 32 addressing lines access a 4 gigabyte physical addressing range.
2. 256 byte caches store most recently used data and instruction.
3. A write pipeline has space for three sequential writes to I/O or memory.
4. The inter-processor interrupts operate in multiprocessor environments.
5. An internal MMU translates logical to physical address in hardware and protects each addressing region from illegal access.
6. Read-Modify-Write cycles are implemented to insure data integrity.
7. Dynamic bus sizing automatically determines device port sizes on a cycle by cycle basis.

The Harvard architecture is implemented on multiple internal buses in the MC68030 which allows internal caches, MMU, and external bus controller to operate in parallel. The individual steps required for instruction execution are thus overlapped to increase processing speed.

1.4.2 Floating Point Coprocessor

In select benchmark tests, the MC68882 floating point coprocessor performs four times as fast as the MC68881. On average, it provides over 1.5 times the performance of the MC68881 at the same clock frequency. In ISI systems the floating point unit operates at CPU speed, typically 25 MHz.

The MC68882 is a full implementation of the *IEEE Standard for Binary Floating-Point Arithmetic* (754) with the addition of a full set of trigonometric and transcendental functions, on chip constants, and a full 80-bit extended precision real-time data format. Since all data transfers are performed by the main processor after an MC68882 request, memory management, bus errors, and bus arbitration instructions function as if they were executed by the MC68030.

Communication between coprocessors is asynchronous.

1.4.3 Caches

Internally, the MC68030 has two 256 byte instruction and data caches which are described in *Motorola User's Manual, MC68030UM/AD*, from Motorola Inc, 1987.

External to the microprocessor, the VME-68K30 has a 64 Kbyte on-board cache that combines instructions and data. The external cache is briefly described in the following. Its relation to system memory and main memory is shown in Figure 1-3.

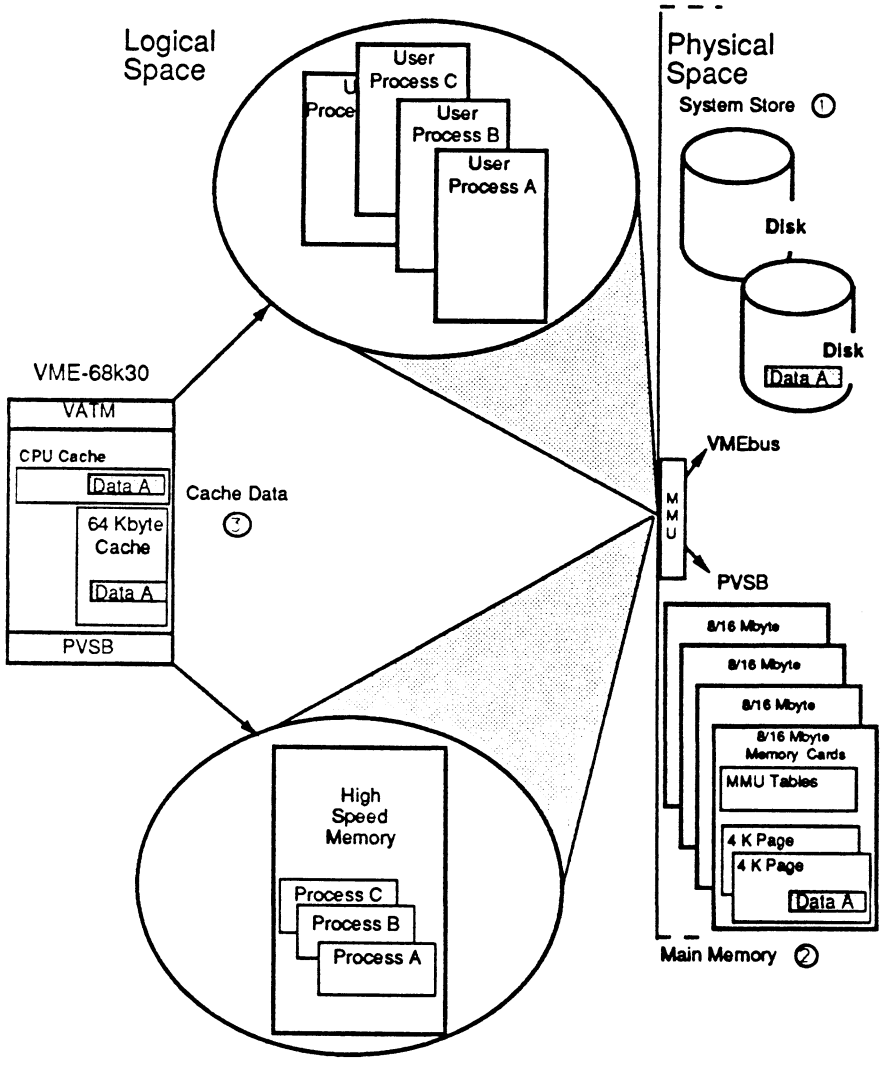


Figure 1-3. Cache Data and Disk Data

The on-board cache is a one-way set associative or direct mapped cache implemented in high speed DRAM. Figure 1-3 shows the relation of CPU and on-board cache to system memory and disks.

As Figure 1-3 illustrates, data - which is represented in the figure as Data A - can reside simultaneously in four places in an ISI system with VME-68K30 processors. All data that is related to all processes resides on disks or system store which is shown as (1) in the figure. Data that is related to a running process is taken from disk and transferred to main memory where it is stored in DRAM as shown in (2). The kernel is responsible for moving data from disk where the access is very slow to DRAM where the access is much faster. It moves data on a demand basis. When a running process calls for data, the kernel moves it from disk to the faster access DRAM one page at a time. Each page has 4K of data. When the process is no longer running, or requires more than the 4K page, the kernel swaps out the old page and brings new data from disk to the main memory in DRAM.

As the process uses data, again say Data A in the figure, it does not simply replace it, but maintains it in cache, which is shown in part (3) of the figure. The assumption is that if a process requires data once, it will require the same data again. If it is cached on-board it can be fetched much more quickly than if it were stored in main memory or disk. When many processes are running, data in the small 256 byte internal caches on the MC68030 are quickly swapped out. But a copy is retained for a much longer period in the 64K external cache and can quickly be retrieved when the process runs again.

1.4.4 Cache Operation

To avoid initiating a memory cycle during a CPU read or write cycle, hardware first searches the internal and then the external cache for the required instruction operand.

If the operand is found in either cache, the access to main memory is inhibited, the operand is given to the CPU, and the next instruction is executed. Finding the operand in either cache constitutes a cache hit which reduces the number of memory cycles required to fetch operands from main memory and greatly improves system efficiency. A status register bit allows a cumulative record of hits to be maintained for accounting purposes and system analysis.

A cache miss occurs if the operand is not found in either cache and the memory cycle must be executed. Successive misses tend to reduce overall system efficiency because of the cache search.

1.4.5 Cache Coherency

A cache contributes to system efficiency if it maintains a high ratio of hits to misses. This requires that the cache be coherent, which means data in the cache is related to the processes being run, rather than to I/O transfers or other data not related to instruction operands.

To maintain cache coherency, ISI has provided three different areas in high speed memory that— when they are written— cause the cache to be updated in different ways or not be updated at all. The following briefly describes each area while maps are presented in Section 2:

1. Cache Invalidate. Data written to cache invalidate is not cached. Moreover, any operands located on the same tag line as the written data are invalidated when the cache invalidate area is written.
2. Write allocate. When data is written to this space the cache is updated whether it contained an entry before or not, and regardless of the state of that entry.
3. No-write allocate. When this space is written the cache is updated only if a previous entry existed in the cache. Otherwise it is not updated

1.4.6 Cache Construction

Cache hardware compares operand addresses with cached data addresses at three levels. Operand address bits A27-16 are compared to cache addresses to determine if a cache entry exists for the 256 Mbyte block that the CPU accesses. Simultaneously, operand addresses A15-A4 are compared to cache entries to determine of an entry exists within a 64K block of the current access. If these two comparisons match, valid bits in the cache tag RAM are checked against A3-2 to see if the cache contains the specific 32-bit long word that matches the operand address. If all matches are true a cache hit occurs.

Figure 1-4 shows the logical construction of the cache.

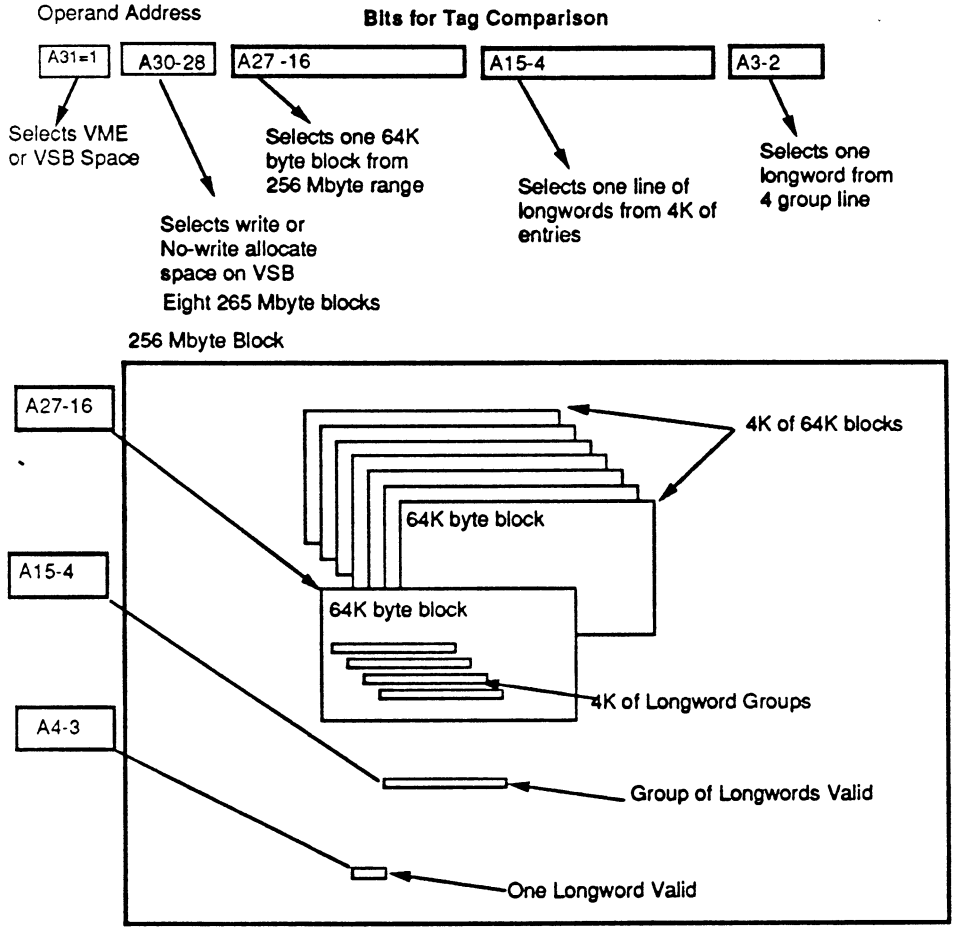


Figure 1-4. External Cache

Each cache tag entry contains complete information about cache data for the hardware comparators. Its format is shown in Figure 1-5.

Address of Cached Data			Cached Data			
Valid Bit	Index	Index	LWord	LWord	LWord	LWord
XXXX	A27-16	A15-4	32-0	32-0	32-0	32-0

Figure 1-5. Format of Cache Tag Entry

If the valid bit is clear, signifying no cache entry to match the CPU access, a bus error is asserted by cache logic and the the bus interface unit (BIU) in the CPU initiates a memory cycle. The internal and external caches are updated after a miss, unless a no write allocate or cache invalidate area of high speed memory is being written.

1.4.7 Cache Burst Fill

Burst mode may or may not be implemented on your version of the VME-68K30 and it may or may not enhance processing speed, depending on how much context switching is done and how the applications are written. In general burst mode enhances performance if a program progresses linearly through data, but degrades performance when the program often jumps to different instruction sets. If your processor is capable of burst mode fills, you will be notified of that option in the PROM prompt when the system is first booted. Whether burst mode enhances performance can only be determined by time trials or experience in using this mode of operation.

If burst mode is set during a write cycle, the four long words in a single cache tag entry are updated in cache fill burst mode. A burst fill updates the entire cache line. The burst is allocated by the Motorola 2-1-1-1 synchronous cache fill. The first pipelined write to cache requires two clock states; subsequent writes require one.

All cache writes except synchronous burst fill consume 3 clock cycles. This includes a write through cycle where the cache is not updated, the write allocate cycle, where it is, and write misses where the address of an operand is not found in the cache.

Only the most recently used operands are maintained in the limited space of the internal cache. The CPU continually over writes them to make room for new data during processing. The advantage of the larger external cache is that it extends the store of recently used operands and rapidly restores internal cache contexts in a multi-processing environment.

1.4.8 MMU

When the MMU is enabled, it translates logical addresses into physical addresses. The MMU concludes a translation when it has searched the translation tree and matched a logical address to its physical equivalent. An on board jumper is provided to activate the MMUDIS.

There are two accesses that do not require the MMU.

1. CPU space is always mapped from the logical to the physical 1:1.
2. Logical addresses in a transparently translated block are used as physical addresses, without modification, and without protection checking. Two translation registers (TT0 and TT1), each capable of defining a 16 Mbyte address space are available to define transparent blocks. The MMU and translation registers are fully described in Section 9 of the *Motorola User's Manual, MC68030UM/AD*, from Motorola, 1987.

1.4.9 EPROM

A single, 32-pin EPROM socket accepts standard 27XXX series devices with 32 to 512 Kbytes of storage. The EPROM contains a monitor, limited diagnostics, initialization instructions and boot strap loading. Initialization is described in Section 4 of this manual. The diagnostics are described in Section 6.

1.4.10 Serial Ports

The Signetics SCC2698A is a single MOS-LSI chip with eight universal, asynchronous communication channels. It is the primary user and peripheral devices interface to the VME-68K30. The chip also has four programmable timer/counters, all of which are used by the ISI kernel. The *SCC 2698 Octal Universal Asynchronous Receiver/Transmitter Product*

Specification Guide, 853-1267 9872 from Signetics gives complete programming information for the chip. The programmable registers are located in ISI defined space beginning at C000 0000. For details see Section 4 of this manual.

In terms of signals generated, the channels provide RS-232-C request to send (RTS), clear to send (CTS), receive data (RxD), and transmit data (TxD) signal lines in data terminal equipment (DTE) mode. All outputs are TTL compatible. Channel operating modes include full-duplex, automatic echo, and local and remote loop back. One channel has full modem control.

The baud rate of each receiver/transmitter channel is independent and can be selected from one of the following:

- One of eighteen internal clocks speeds with a 50 to 38.4K baud range.
- A 16x, clock derived, programmable counter/timer.

The current configuration allows jumper selection from 1200, 2400, 9600, or 19200 baud for the console. Section 5 describes jumper selection.

The baud rate generator and counter/timer are driven by an external crystal.

The chip also has a power down mode in which oscillator inputs are not processed, but all command, status and data registers are saved.

Error detection includes parity, framing and overruns, false start bit, and line break detection and generation.

For pin assignments see the J1 connector description in Section 5 of this manual.

1.4.11 Battery Backed Clock

A 2K byte static RAM maintains such information as the machine serial number, board serial number, base art work number, board revision level, auto boot arguments, and the machine internet address.

The on-board, real-time clock has an internal crystal, power-failure detection, and battery. It maintains the date in year, month, day, hour, minute and seconds. The battery backed clock and 2Kx8 SRAM are both on a single chip. Calibration is controlled by software, and worst case battery storage life is 11 years, at 70°C. The clock automatically corrects for 28, 29, 30 and 31 day months.

Typical accuracy is ±1 minute per month. Writing the clock to update the current time registers does not disturb clock operations.

Power fail circuitry detects supply voltages within 4.75 to 4.5 volt window and write protects RAM after asserting unselect. The same circuitry tests battery voltage and sets a flag if voltage falls too low for dependable operation.

The clock is programmed in ISI defined space. See Section 4 of this manual for details.

1.5 VMEbus Interface

Functional modules on the VME-68K30 provide the following VMEbus interfaces.

1. A full function location monitor generates an on-board interrupt when its assigned addresses are detected on the VMEbus. The location monitor recognizes Short I/O, Standard, and Extended addressing so the board can be located anywhere on the VMEbus.
2. Master/slave modules:
    a. The master functional module initiates and directs data transfer bus (DTB) cycles to slave modules.
    b. The slave functional module detects and responds to DTB cycles initiated by a master and directed to it. When the VME-68K30 is in slave mode, other VMEbus masters can access its PVSB memory.
3. Bus grant and request modules:
    a. A four level arbiter that accepts requester signals BR0*-BR3* and grants the bus to one requester at a time is enabled by jumpers on the VME-68K30. In a cluster installation only one processor designated "cluster 0" or "server" has the bus arbiter enabled.
    b. On all processors in a system, a four level requester arbitrates for master status and accepts the appropriate bus grant (BGIN0-3) in signal to acknowledge its request. The processors also propagate both the bus grant in and bus grant out (BGOUT0-3) signals if arbitration is being granted on a request level other than its own, or if it has no on-board requests.
4. Interrupts:
    a. An interrupter module generates an interrupt request and provides Status/ID when acknowledged.
    b. An interrupt handler module detects interrupt requests from other bus masters and responds with a request for status identification (Status/ID).
    c. A daisy chain driver generates an interrupt acknowledge (IACK) when an interrupt handler acknowledges a request.
5. The VME-68K30 responds to system reset (SYSRESET*) signals from other masters when in slave mode or generates the SYSRESET signal when configured as a server and in master mode.

As the foregoing suggests, in a multiprocessor environment such as a cluster installation, the responsibility for generating or responding to standard VMEbus signals is different for a server (cluster 0) and individual clusters (cluster1-7). Table 1-2 illustrates the differences in functional modules implemented on a VME-68K30 when it is used as a server rather than a cluster.

A brief summary of functional modules on cluster and server CPUs is presented in Table 1-2.

Table 1-2. Cluster and Slave Functional Modules

VMEbus Functional Module	Activated	
	Cluster 0	Cluster 1-7
IACK Daisy Chain Driver	Yes	Yes
Interrupt Handler	Yes	No
System Clock Driver	Yes	No
Power Monitor	Yes	No
Bus Timer	Yes	No
Arbiter	Yes	No
Master	Yes	Yes
Slave	Yes	Yes
Serial Clock Driver	No	No
Location Monitor	No	No
Requester	Yes	Yes
Interrupter	Yes	Yes

NOTES

1. System clock (SYSCLK) signals are generated on the VME-68K30 when it is the server (Cluster 0). An ISI slot jumper card also generates this signal. Use only one system clock in a system. Disable all other sources.
2. AC failure detection and signaling (ACFAIL) is available in ISI systems on a separate board that does not require a VMEbus slot.
3. Serial Data (SERDAT*), Serial Clock (SERCLK), and System Failure (SYSFAIL) are not generated or used by ISI systems or boards.

1.6 VMEbus Address

A number of parameters are associated with the VME-68K30 address in VMEbus space, all of which are selected by a single rotary switch. The switch sets the address of the processor board in three VME spaces simultaneously by creating the following windows into the board address space.

1. A 2 Kbyte window in VMEbus Short I/O space which includes the boards 256 VDMA registers used for programmable access to PVSB space.
2. A 1 Mbyte window into VMEbus Standard address space.
3. A 255 Mbyte window into VMEbus Extended address space.

If the rotary switch is set to "0" it tells the software that the board is the server in a cluster system, and that its base address will be VMEbus address 0000 0000. It also enables the on-board arbiter so the board will grant bus requests for the VME to other clusters and DMA masters in the system.

Other clusters in the system will be set with the same size windows as the server, but the rotary switch setting will increase the base address by the size of the window allocated each cluster by the hardware setting of the switch. For more details on the switch setting, see Section 5 in this manual.

1.7 VMEbus Address Translation Map (VATM)

The MC68030 uses a map shown in block diagram in Figure 1-6 to access the VMEbus. The map is loaded with sixteen offset addresses that include address modifiers AM5-0. An access to the VMEbus selects one of the sixteen offsets which are then sent out on the VMEbus with local address lines A26-0.

The map is built in three 16 x 4 bit static RAMs. After programming, all accesses to the VMEbus go through the map. Section 4 of this manual contains programming details.

After the VATM hardware is programmed, local bus addresses A30-27 select one of the sixteen offsets that include VME AM5-0 and VA31-27. Within VMEbus space, the offsets access a 128 Mbyte block and local address lines A26-0 select one address within that block.

Signal VME 16 enables a word swap buffer when the CPU is writing data to the VMEbus. When VME 16 is set to "1" the buffer breaks a 32-bit long word into two 16-bit words for devices that can only accept word wide transfers.

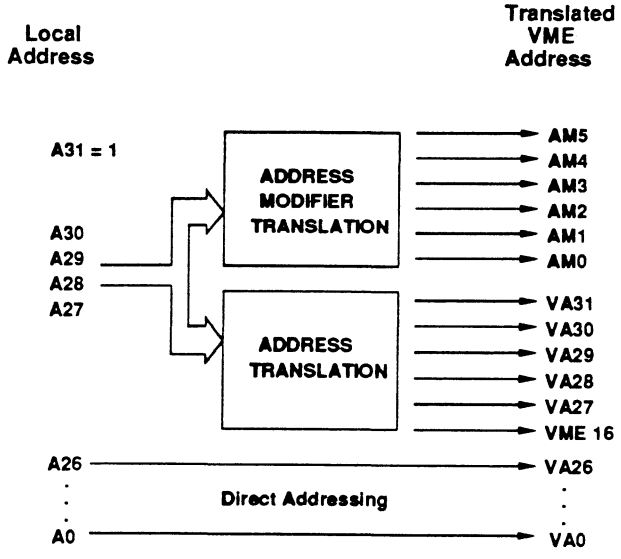


Figure 1-6. MC68030 Accesses to VMEbus

1.8 VDMA Map

The VDMA is a map onto the PVSB that is implemented in a 256 x 16 SRAM. The SRAM registers are physically located on-board, but the map is loaded by writing registers in VME Short I/O space.

When the map is loaded, its registers allow any DMA bus master, including another cluster, to access a 256 Mbyte region of contiguous memory through a 1 Mbyte window in Standard I/O or through a 256 Mbyte window in extended I/O. Details on programming the map and illustrations of the mapping access of the VDMA are located in Section 2 in this manual.

1.9 Programming Devices on the PVSB

The PVSB is a local bus designed to off load main memory accesses from the global VMEbus. It has 32-bit multiplexed data and address lines and separate lines for bus arbitration and control. On the PVSB, the memory cards function as slaves. However, the cards do not provide full slave functionality, so all accesses from the MC68030 or remote VMEbus masters are arbitrated through VME-68K30 logic. Section 2 gives a complete description of the PVSB.

1.10 Resets

Microprocessor reset and System Bus reset are configurable for master or slave operation.

Each CPU board recognizes power on reset (POR), cluster reset, and VME reset. On-board stake pins allow for an optional push button reset.

VME-68K30 boards configured for master mode supply reset to the VMEbus as well as software resets.

Slaves pass bus reset to their CPU. Sources for the various resets on-board or from the bus are shown in Table 1-3.

Table 1-3. Reset Sources in Master and Slave Mode

Reset Source	Master	Slave
Power on Reset	yes	yes
VME Reset	yes	yes
Push Button Stake Pins	yes	yes
Generate VME Reset	yes	no
CPU HALT	option	option
Cluster	yes	yes
Software Generated	yes	yes (not used)

After a reset, the sequence of operations is as follows:

1. EPROM is enabled
2. The MC68030 fetches the initial SSP and PC from location 0000 0000.
3. The OCTART, sio, and baud rate are checked and initialized. This includes printing the ISI logo to the console screen with such SRAM contents as version number, CPU operating speed, and PVSB or VSB memory type.
4. Memory is probed and configured which includes printing the size of configured memory to the screen.
5. EPROM contents are copied into RAM duplicate space.
6. Program execution jumps into duplicate space, which transfers EPROM execution to PVSB DRAM.
7. Normally EPROM is disabled at this point and UNIX is booted. However, if the on-board switches have been set for extended diganostic testing, the diagnostics run out of duplicate space. (See Sections 5 and 6 to set the board for diagnostic options.)

1.11 Indicators

DS1 CPU HALT indicator (puce)
DS2 VMEbus Slave indicator (yellow)
DS3 VMEbus Master indicator (green)
DS4 VSYSFAIL indicator
DS5 Diagnostic code
DS6 Diagnostic code
DS7 Diagnostic code

The diagnostic codes are described in Section 6.

Section 2: High Perfromance PVSB

2.1 Introduction

PVSB† memory cards provides up to 64 Mbytes of main memory for a VME-68K30 processor. The memory cards are populated with 4, 8, or 16 Mbytes of high speed DRAM. In a multiprocessor installation, all memory cards can be accessed by any DMA device or any cluster in the installation. The relation between four VME-68K30s and their PVSB memory boards to the VMEbus is shown in Figure 2-1.

2.2 PVSB Features

- 32 multiplexed address and data lines (AD31-0)
- Proprietary subset of *standard* VSB signals
- Block transfers into ISI defined space††
- Supports 256 Mbyte addressing range (A27-0)
- Supports direct and virtual DMA from other VMEbus masters

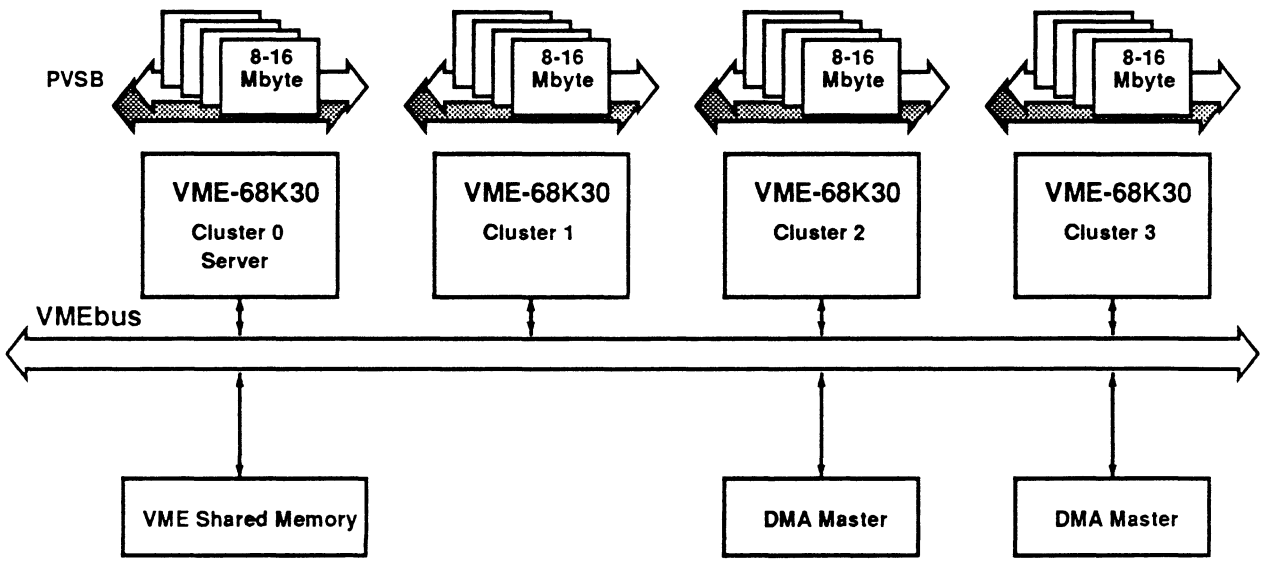


Figure 2-1. Clusters and PVSB Memory Boards

† PVSB stands for the high *P*erformance VSB bus and memory cards that are proprietary to ISI. ISI memory cards can only be used on the PVSB bus. The bus does not support other types of VSB cards.

†† While the MC68030 does not support block transfers, hardware modules on the VME-68K30 and PVSB 8/16 M boards make transfers appear to be in block mode for DMA devices on the VMEbus.

2.3 The VME-68K30 as PVSB Master

Viewed from the VME-68K30, the PVSB is a local bus with one function: it offloads high-speed memory accesses from the global VMEbus.

Since the PVSB is dedicated to high speed memory, access is streamlined and memory cards do not require full slave functionality. By eliminating complicated handshaking protocols the PVSB reduces access time and enhances system performance. Functional modules such as bus grant/request are not needed since the VME-68K30 is the only master. Wait states are not needed because the processor knows the memory timing cycle. Interrupts are limited to one signal (IRQ*) that indicates a parity error.

The VME-68K30 identifies memory boards by slot position on the PVSB. The position of each board is set by geographical address jumpers (GA0-2) and cannot be overridden by software.

Devices on the VMEbus identify and access memory boards directly by the cluster number of their associated processor or through the VDMA map on the VME-68K30. The remainder of this section describes how a cluster accesses its own PVSB memory and how a DMA device on the VME, which includes another cluster, can access all available DRAM memory in a cluster configuration.

2.3.1 Functional Modules

Functional modules implemented on the VME-68K30 to access the PVSB include a bus arbiter with prioritized refresh, a bus control module, and an address translation map (VDMA).

2.3.2 PVSB Arbitration

In order of descending priority, arbitration between the processor and other VMEbus masters for PVSB access is resolved as follows:

1. A VME-68K30 generated refresh cycle
2. VME-68K30 access
3. Access from a VMEbus master while the VME-68K30 is in slave mode

The refresh cycle is interleaved between each processor or bus master access, as well as being guaranteed access when the refresh timer has timed out.

A block diagram of the functional modules on the PVSB-8/16 MB memory board is shown in Figure 2-2.

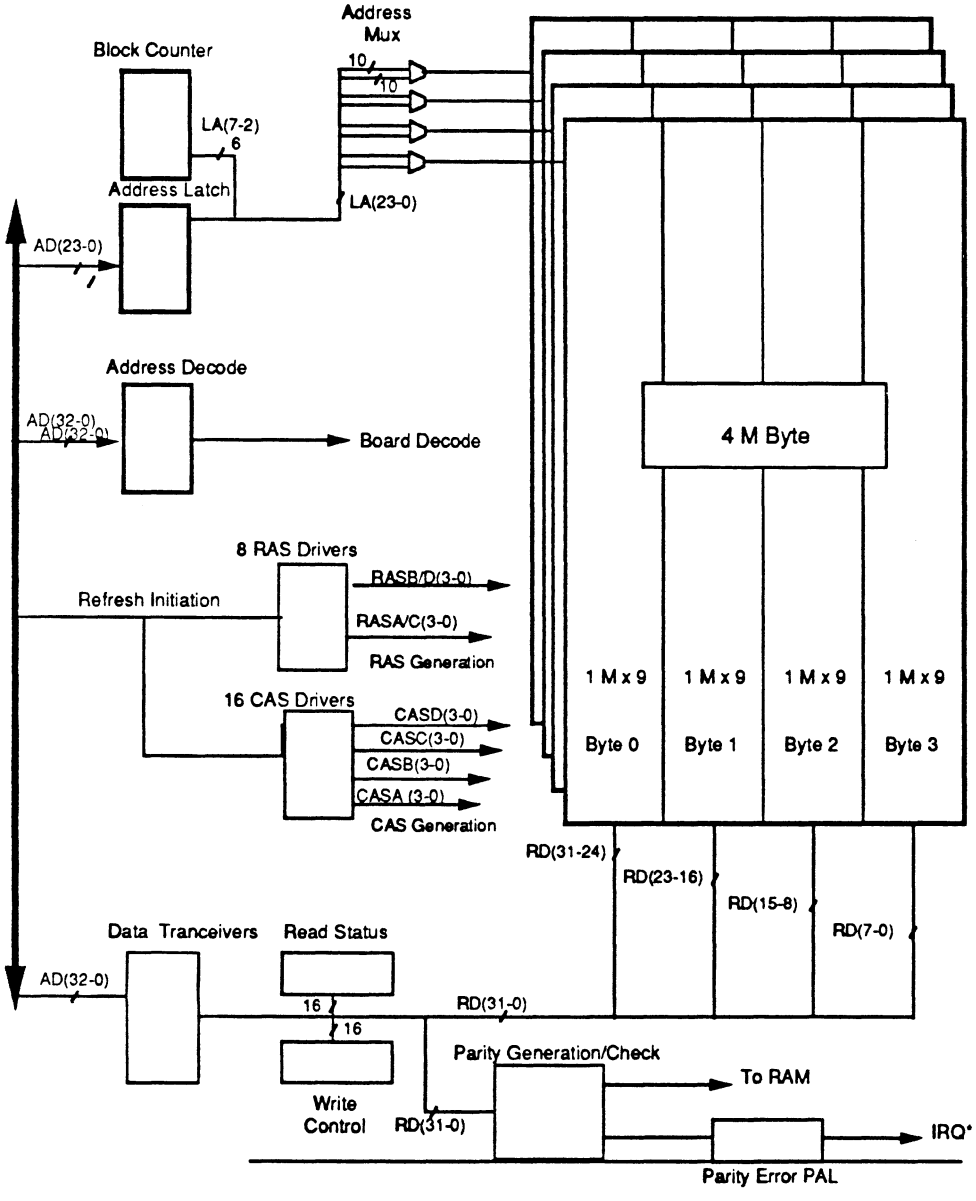


Figure 2-2. Block Diagram of PVSB Memory Board

2.3.3 Address Lines and Control Signals

The VME-68K30 implements the following subset of VSB address lines and control signals for the PVSB.

AD00-AD31 Address/Data Lines. The AD02-AD31 lines determine the address of a long word in memory. Lines AD0-AD1 determine byte position. SIZE0-1 signals determine the number of bytes

Byte Accessed	AD01	AD00
Byte 0	low	low
Byte 1	low	high
Byte 2	high	low
Byte 3	high	high

BG3-0 Since the PVSB is dedicated to high speed memory, bus request and grant protocols required to accommodate a variety of bus slaves and masters were eliminated in favor of faster memory access. Memory boards simply decode their address from AD31-0.

SPACE0-1 In ISI defined space accesses to address range 8000 0000 to BFFF FFFF are detected by hardware that generates the SPACE0-1 codes. These in turn select one of four address regions on the PVSB:

SPACE0	SPACE1	256 Mbyte Block Access
0	0	Interrupt Acknowledge (not used)
0	1	Block Transfer (not used)
1	0	I/O Space (memory board addresses)
1	1	System Space

SIZE1-0 SIZE1-0 codes are used in conjunction with address and data lines A0-1 to determine the size and position of byte transfers. Size codes specify a one to four byte transfer. Byte locations in storage must be consecutive. The SIZE and A1-0 codes are decoded from the DS0-1* and A1* lines of the MC68030.

Data Type	Size1	Size0	Address Space Location			
Quad Byte	0	0	Data0	Data1	Data2	Data3
Triple Byte	1	1		Data1	Data2	Data3
Double Byte	1	0			Data2	Data3
Single Byte	0	1				Data3

ASACK0-1 **Not used.** The PVSB generates *ACK for all accesses.

GA2-0 Jumpers set the geographical address on each memory board. The board recognizes its address when decode logic compares registers containing GA2-0 settings with address lines AD23-16.

Control Lines

PAS* The VME-68K30 drives the physical address strobe (PAS) low to start the address broadcast phase of a PVSB cycle. It remains low during the cycle. In response, the card drives ACK low when it has captured an address.

To end a cycle, the master drives PAS high, and the slave responds by driving its ASACK signal high.

AC **Not implemented** The Address decode Complete (AC) signal used with WAIT allows the slave to prolong the address broadcast phase to fit their decode timing.

WR*	The processor drives the WR* line during the address broadcast phase to inform slave memory boards of the direction of data transfer. WR* is high when the processor retrieves data from the responding slave and low when the processor writes to the responding board.
LOCK*	The LOCK signal on the ISI implementation of the VSB generates a REFRESH signal on the memory board that inhibits its address latches and thus prevents the board from being accessed when DRAM is being refreshed.
DS*	The processor drives DS* low to indicate to the selected slave board that the address broadcast phase is complete and that the data transfer phase is in progress.
WAIT*	**Not implemented** This signal allows boards with different response times to force a bus master to wait until the slave responds to an access. With a dedicated PVSB, the signal is not required.
ACK*	During write cycles, the memory drives ACK* low to inform the processor that it has captured the data on AD31-AD00. During read cycles the responding memory board drives ACK* low to inform the processor that valid data has been placed on all or some of the address/data lines. The processor monitors the falling edge of the ACK* signal for the completion of either cycle.
ERR*	**Not implemented** The ERR* signal is not required for parity errors. IRQ* is reserved for that function.
IRQ*	The memory board drives IRQ* low to generate an interrupt to the processor, and the processor responds by polling each board to find parity error information. In the ISI implementation of the PVSB, this is the only function of IRQ*.
CACHE*	**Not Implemented**

2.4 PVSB Initialization

On power up, the VME-68K30 CPU masks all interrupts and goes into supervisory mode. To access and initialize the memory boards, it sets the function code to 3. It then enables the internal cache and checks the status of main memory on the PVSB.

PVSB memory before initialization is shown in Figure 2-3.

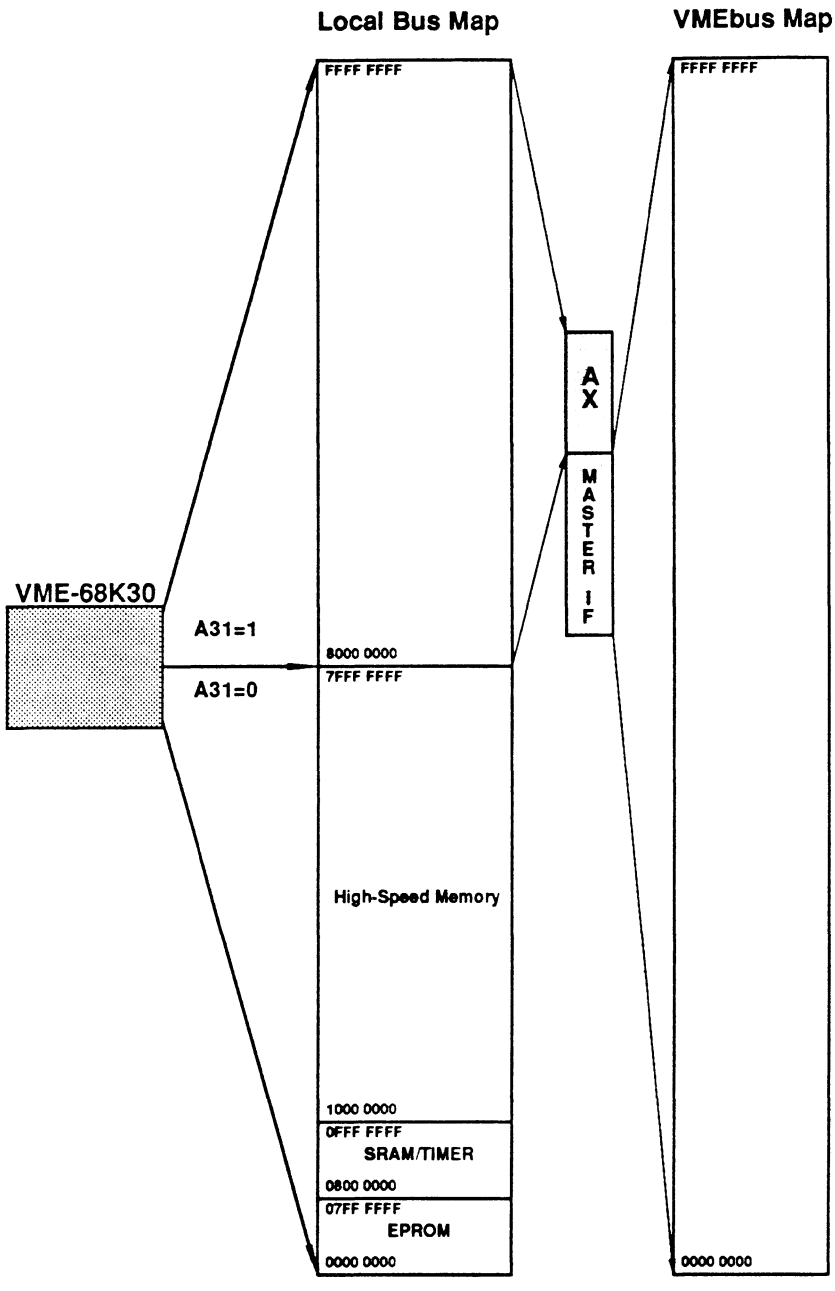


Figure 2-3. PVSB Address Space Before Initialization

NOTE

In Figure 2-3, AX is Address Translation, IF is Interface, SRAM is Static RAM, and EPROM is Erasable Read Only Memory.

2.5 ISI Defined Space

ISI defined space includes PVSB I/O space, local I/O space, and cache invalidate space. These regions are accessed when the function code in the MC68030 is set to 3 (FC=3). If A31=0 and the function code is 3, cache invalidate space is accessed. Any write to this space will invalidate all four corresponding entries in a cached line. This is the only time an entire line rather than a single valid entry is invalidated. Cache invalidate space is shown in Figure 2-4.

If A31=0 and the CPU accesses ISI defined space in the address range 8000 0000—BFFF FFFF, the access is to one of four contiguous memory blocks on the PVSB. If the access is to C000 000—FFFF FFFF, the access is to local I/O devices that are programmed on the PVSB.

The four contiguous memory blocks are defined as follows:

1. ADDRESS SPACE R/W 8000 0000 - 8FFF FFFF
   IACK space is not used by the VME-68K30
2. ADDRESS SPACE R/W 9000 0000 - 9FFF FFFF†
   Alternate space is not used by the VME-68K30.
3. I/O ADDRESS SPACE R/W A000 0000 - AFFF FFFF
   I/O address space is used by the CPU to set parameters on the PVSB boards. The write control and read status registers for each memory board are located in this space.
4. SYSTEM ADDRESS SPACE R/W B000 0000 - BFFF FFFF
   The system address space stores user and supervisory programs and data.

The standard VSB SPACE0-1 codes are generated in hardware when addresses in the range 8000 0000—BFFF FFF are detected. The PVSB interprets SPACE0-1 codes as shown in Table 2-1.

Table 2-1. SPACE1-0 Code Interpretation

SPACE0	SPACE1	Access Space
0	0	Not used by the VME-68K30
0	1	Not used by VME-68K30
1	0	I/O Address Space (PVSB I/OAS)
1	1	System Address Space (PVSB SAS)

As shown in Figure 2-4, I/O address space is a 256 Mbyte region that begins at A000 0000 and extends to AFFF FFFF. The bottom 8 Mbytes of this region are reserved for memory board addresses.

† Alternate address space within the ISI defined region is reserved for block transfers. The MC68030 does not support block transfers, but accepts them from VMEbus masters, so it appears to support them.

Within the 8 Mbyte region reserved for board addresses, memory boards reside on one of eight possible 64 Kbyte boundaries. The total address space reserved for boards is 512 Kbytes (64K x 8), but due to electrical limitations on the backplane, only four memory boards are allowed with each processor.

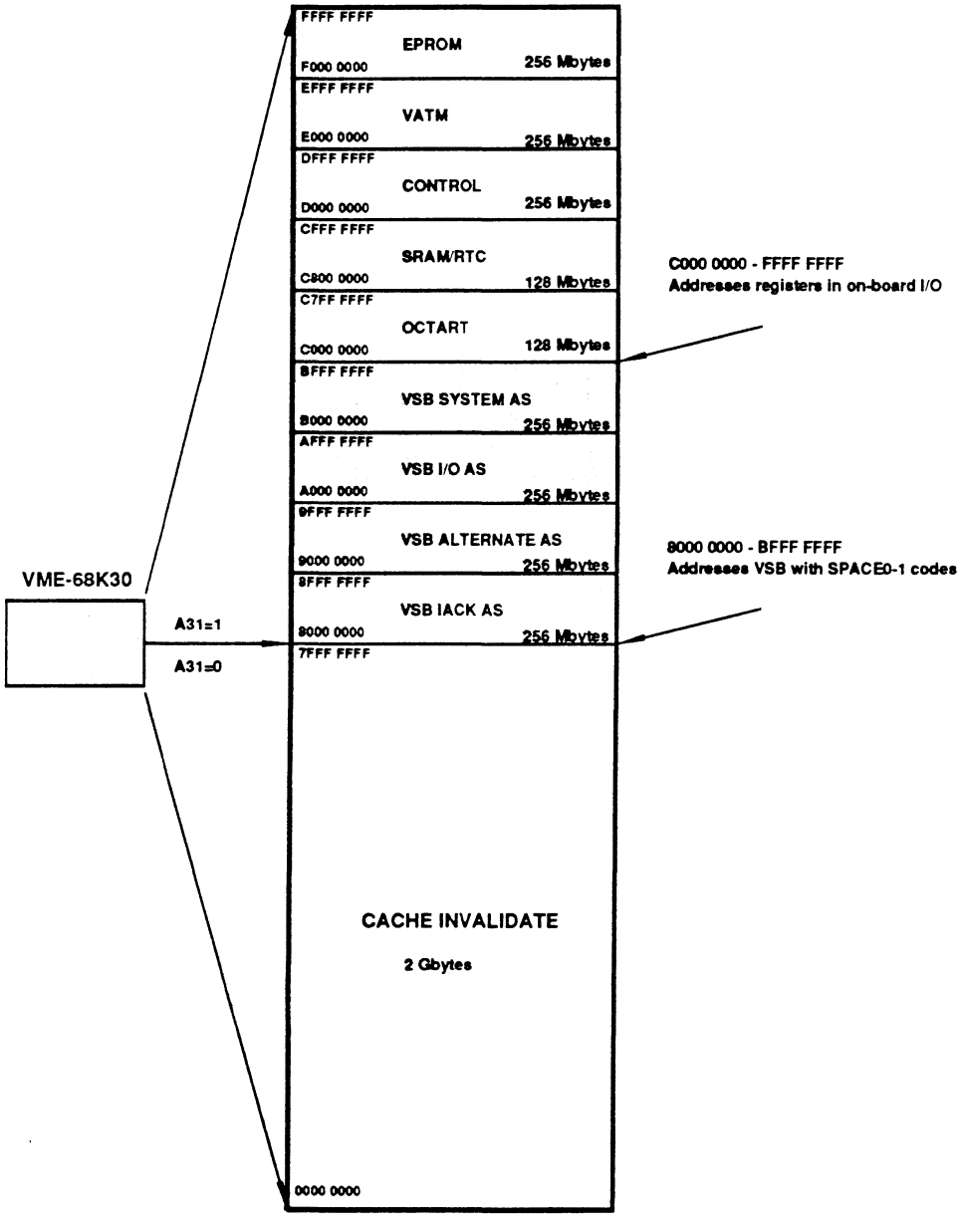


Figure 2-4. ISI Defined Space on the PVSB

2.6 Memory Board Initialization

On the PVSB, each board recognizes that it is being accessed through address decode logic. The logic compares PVSB address/data lines AD23-16 to on-board geographic address jumpers (GA2-0). The jumpers are set during hardware configuration and cannot be overridden by software.

While the VME-68K30 probes memory boards, it displays the a message similar to the following on the screen:

Probe and configure PVSB memory
A00*N* 0000 0000 000*Y*
A00*N*+*1* 0000 0000 000*Y*
A00*N*+*2* 0000 0000 000*Y*

The slot number is designated *N* in the display. The processor looks for 16 Mbyte cards first, 8 Mbyte cards next, and 4 Mbyte cards last. Since cards can be placed in any slot position, the first 16 Mbyte card that the CPU finds may not be in slot one, but it will be the first slot displayed. The total memory found on each card is displayed in the position designated by *Y* in the example. Memory is counted in 4 Mbyte units so a 16 Mbyte card would be shown with a 3 (0-3 units) in the *Y* position. In the example, three lines are displayed which tells you three slots have been found with memory cards.

The initialization message concludes with a line that indicates memory is initialized. A dot is displayed for each 4 Mbyte unit located.

Initialize all memory.....

A status and control register is located on each memory board addressing boundary. As the CPU polls through each status register it determines whether a memory card is present. If a slot does not respond, the processor assumes the slot is empty. If a card is present, the CPU determines how much memory it contains.

The location of the registers on each board is shown in Table 2-2.

Table 2-2. Memory Board I/O: SPACE0-1 = (10_b)

Register	# Bytes	R/W	Addresses	Range Reserved
Write Control	64K	W	XX0Z 0000	XX0Z 0000 - XX0Z FFFF
Read Status	64K	R	XX0Z 0000	XX0Z 0000 - XX0Z FFFF
Full Range	512K	R/W	NA	XX00 0000 - XX07 FFFF

In Table 2-1, the actual value of *Z* for each memory board is determined by geographic addressing jumpers GA2-0. Values of "Z" range from 0-7.

2.6.1 Status Register

The bit assignments for the status register on each memory board are shown in Figure 2-5. The register is read only and one word wide.

Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Board ID						RS1	RS0	PE	EBT	MG	MB	PEB*n*			

Figure 2-5. PVSB Memory Board Status Register (Read Only)

The upper byte of the status register identifies the board as a PVSB memory board and indicates the amount of on board DRAM.

ID — Bits D15-10 are always set low to identify the board in this slot as PVSB memory.

RS0-1 — These two bits show the presence of 4, 8 or 16 Mbytes of DRAM. The bits are set by jumper E-7 and interpreted as follows:

Size	RS1	RS0	Total Memory
0	0	0	4 Mbyte DRAM, 1M x 1
1	0	1	8 Mbyte DRAM, 1M x 1
2	1	1	16 Mbyte DRAM, 1M x 1

The lower byte in the status register provides parity error information. A parity error is signaled by IRQ* on the PVSB and results in a level 7 interrupt on the VME-68K30. Reading the lower byte of the control status registers on the PVSB identifies the board that produced the error and the group, bank, and byte which caused it. Parity must be disabled and re-enabled by software to remove the IRQ assertion. The group, bank, and byte location of an error is sufficient to identify which zig-zag inline package (ZIP) caused the error.

PE — Parity Enable, indicates whether parity has been set.

EBT — Error Block Transfer, indicates a block or byte transfer was in progress.

MG — Memory Group, indicates the group in which the error occurred.

MG — Memory Bank, indicates the bank in which the error occurred. The memory group (MG) and memory bank (MB) bits are interpreted as follows:

Memory Group	Memory Bank	Offset Memory Bank	1M x 1 DRAM
0	0	A	XX0X XXXX
0	1	B	XX4X XXXX
1	0	C	XX8X XXXX
1	1	D	XXCX XXXX

PEB3 — Indicates a Parity Error in Byte 3 (D7-0). Active low.

PEB2 — Indicates a Parity Error in Byte 2 (D15-8). Active low.

PEB1 — Indicates a Parity Error in Byte 1 (D23-16) Active low.

PEB0 — Indicates a Parity Error in Byte 0 (D31-24). Active low.

2.6.2 Write Control Register in PVSB I/O

The write control register indicates both the 16 MB address range which the board will decode and the offset within that range where decoding will begin. The offset is restricted to the size of RAM available.

Bit assignments for the 16 bit control register on each memory board are shown in Figure 2-6. CPU bit assignments refer to MC68030 internal register bits. The CPU bits are stored as D15-8 in the memory board register as shown. The register is one word wide. It is cleared on VME Reset.

CPU Bit #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reg Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE	Unused					Base Address (Megs)								BOA	

Figure 2-6. PVSB Control/Status Register Write

PE — Parity is enabled by setting the most significant bit (#15) in this register.

Unused — Control bits 14-10 have no assigned function.

Base Address — The base address of each PVSB memory board is set on a 4 Mbyte boundary starting with 0000 0000. Boards are cumulative for each processor. As an example, if two 16 Mbyte boards are installed with a processor, the address of the first would be set to 0, the address of the second would be set to 4.

BOA — The base offset address bits are interpreted as shown. PVSB address bits A/D 31-24 are compared with BS1-0 bits to decode a 16M range.

Base	BS1	BS0	Base Offset Address
0	0	0	XX00 0000
1	0	1	XX40 0000
2	1	0	XX80 0000
3	1	1	XXC0 0000

Note: The eight bit pattern stored in register bits 9-2 is represented by XX.

Firmware on each memory card writes the amount of on-board memory into BOA1-0. The CPU reads the value for each card and keeps a cumulative count of all memory present. Each card is recorded on the boundary reached by the highest address of the last card read, so all cards have contiguous addressing in PVSB.

2.7 Initialized PVSB Address Space

Figure 2-7 shows PVSB space (0000 0000 to 7FFF FFFF), the space accessed after memory initialization is complete and function codes are set to 1, 2, 5 or 6.

2.8 System Space: FC=1,3,5,and 6

The MC68030 uses ISI Defined Space for I/O activity. System space is reserved for user and supervisory data and programs, and is shown in Figure 2-7. System Space is divided into eight 256 Mbyte blocks as shown, and the blocks alternate between allocate and no-allocate writes to cache. The differences are defined as follows.

- **Write allocate.** When hardware on the VME-68K30 detects an access to an address in write allocate space, the cache is updated regardless if it contained an entry before or not, and regardless of the state of that entry.
- **No-write allocate.** When hardware on the VME-68K30 detects an access to this space it updates the cache only if a previous entry existed in the cache for this space, otherwise not.

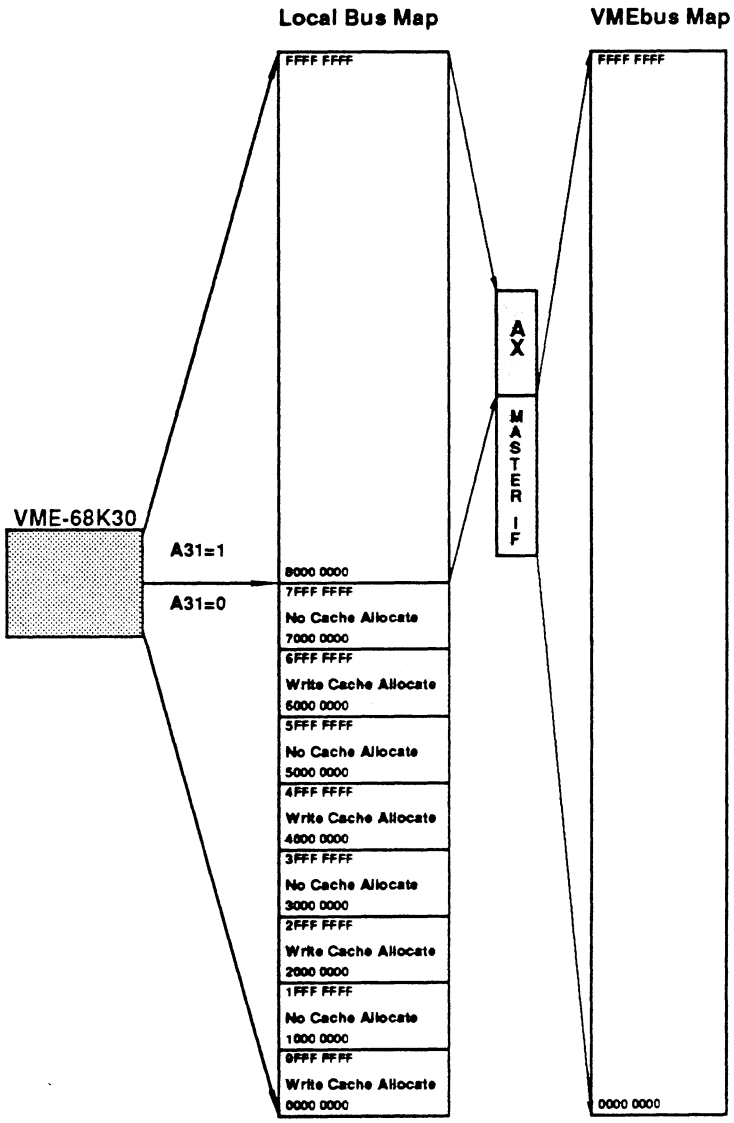


Figure 2-7. PVSB Address Space After Initialization

2.9 PVSB Access

After initialization VME-68K30 accesses to PVSB memory occur under two conditions. The first is when A31=1 and the Motorola function code is three (FC=3). The second is when A31=0 and the function code is 2, 3, 5, or 6. The settings of function codes and A31 are summarized in Table 2-3.

Table 2-3. PVSB Addressing Modes

VME-68K30 Accesses			
Bus I/D	Cluster I/D	Function Code	Address Description
A31=1	—	FC=3	ISI Defined Address Space
A31=0	—	FC=1, 2, 5, 6	System Address Space
Virtual DMA (VDMA) Accesses			
A31=0	A22-20	FC=1, 2, 5, 6	Virtual VMEbus Standard Address access
A31=0	A30-28	FC=1, 2, 5, 6	Virtual VMEbus Extended Address access
A31=0	A30-28	FC=1, 2, 5, 6	Direct Extended Address access

Details on virtual and direct accessing into the PVSB are given in Section 3 which discusses VME-68K30 addressing space.

2.10 PVSB Configuration

As shown in Figure 2-8, there are eight jumper sets that configure the PVSB memory board. For quick reference, they and their functions are listed in Table 2-4.

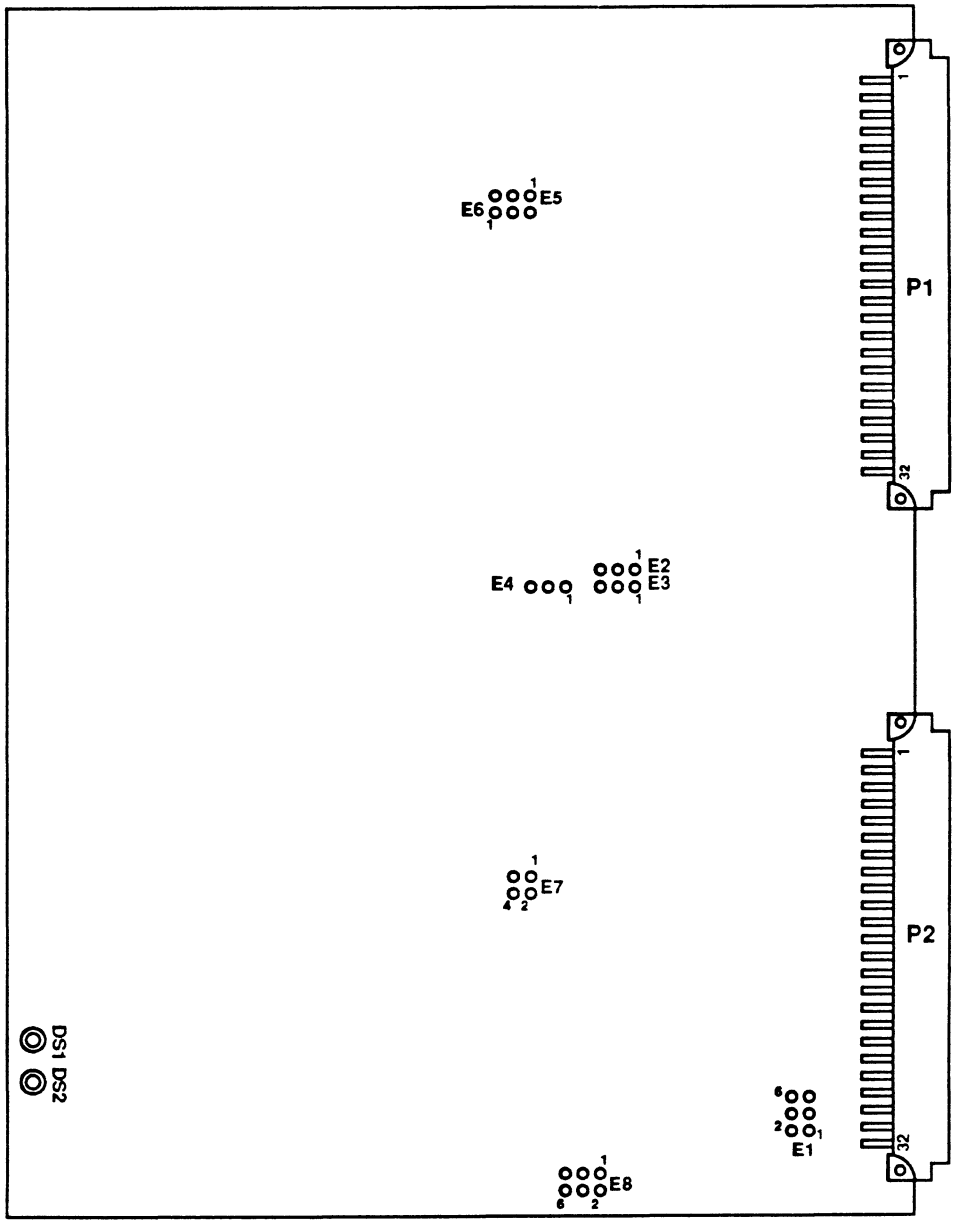


Figure 2-8. PVSB Memory Board Jumper Positions

Table 2-4. Jumper Descriptions

Jumper	Function
E1	Connects the GA2-0 address lines from the PVSB to on board registers. Leave open in all ISI installation using the VME-68K30 designed for PVSB memory cards. Interleaves memory banks A and C, B and D.
E2-E3	Identifies whether one or more than one memory banks is installed on a board.
E4	Sets RAS precharge and refresh cycle lengths.
E5-E6	Set the available on-board RAM in 4 Mbyte increments.
E7	The I/O base address jumpers are used instead of the GA2-0 signal lines
E8	from the PVSB to set the board base address when the memory board is used with the VME-68K30.

2.10.1 E1: GA2-0 Input Enable

Leave E1 unjumpered when using the VSB-8/16MB memory boards with the VME-68K30 processor, as shown in Table 2-5. Defaults are shown in **boldface**.

Table 2-5. E1: GA2-0 Input Enable

Signal	Jumper
GA2	**E1 out**
GA1	**E1 out**
GA0	**E1 out**

2.10.2 E2-E3: Bank Interleave

Interleaving high-speed memory alternates memory bank access which allows memory access to one bank while the other is in a precharge refresh cycle. Each bank consists of 4 Mbytes of memory. Banks are installed to create boards of 4, 8 or 16 Mbytes capacity. Only the 8 and 16 Mbyte boards are interleaved. Since a 4 Mbyte board has only one bank of memory it can not be interleaved. The VME-68K30 runs interleaved memory at the same speed as a non-interleaved memory in current versions so interleaving does not improve access time. Set jumpers E2 and E3 according to the population of your VSB-8/16MB as shown in Table 2-6.

Table 2-6. E2-E3 Bank Interleaving

E2	E3	Description
1-2	1-2	2 or 4 Banks - Interleave
1-2	2-3	Not Valid
2-3	1-2	Not Valid
2-3	**2-3**	**1 Bank - No Interleave**

2.10.3 E4: Number of Banks

The VSB-8/16MB will have one or more 4 Mbyte banks of memory. Set jumper E4 to identify whether there are one or more banks on board as shown in Table 2-7.

Table 2-7. E4: Number of Banks

Number of Banks	Jumper
More than one bank	E4 1-2
Only one bank	E4 3-4

The RAM size jumpers (E-7) must correspond to the E4 setting when there are one or more than one banks installed.

2.10.4 E5-E6: Refresh Cycle Length

Table 2-8 shows jumper settings for E5 and E6 that match the refresh cycle length to the DRAMs used. 100 η DRAMs, used with a 25 MHz machine, and 120 η DRAMs are used with a 20 MHz machine. The refresh cycle is set by E-5 for memory banks B and D. Memory banks A and C are set by E-6. All boards should be set to a 100 ηs delay since slower delay times give unexpected results, even with the fastest DRAM available.

Table 2-8. Refresh Cycle Length

Jumper	Pin	Delay
E5	1-2	80 ηs
	2-3	**100 ηs**
E6	1-2	80 ηs
	2-3	**100 ηs**

2.10.5 E7: RAM Size

Set E7 jumpers as shown in Table 2-9 to indicate the amount of on-board RAM available. The ISI processor board reads these jumpers out of the status register on each board, then determines how much total memory is available on the PVSB.

Table 2-9. E7: RAM Size

E7 1-2	E7 3-4	Size	Total Memory
in	in	0	4 Mbyte
in	out	1	8 Mbyte
out	out	3	16 Mbyte

2.10.6 E8: Base Address on PVSB

Each card recognizes its address on the PVSB by the settings of base address jumpers E8 1-6. Address ranges are described in Table 2-10.

The lower 8 Mbytes of PVSB I/O space is used for board identification and configuration. Jumpers GA2-0 set the base address on a 64 Kbyte boundary. If a location does not respond during initialization, the processor assumes that the slot is empty or contains a board without slave functionality. The default is to have at least one memory board.

Table 2-10. Base Address on PVSB

GA2	GA1	GA0	PVSB Slot	Address Range for the VME-68K30	
1-2	3-4	5-6			
in	**in**	**in**	**0**	**00**	**0000-00 FFFF**
in	in	out	1	01	0000-01 FFFF
in	out	in	2	02	0000-02 FFFF
in	out	out	3	03	0000-03 FFFF
out	in	in	4	04	0000-04 FFFF
out	in	out	5	05	0000-05 FFFF
out	out	in	6	06	0000-06 FFFF
out	out	out	7	07	0000-07 FFFF†

2.11 LEDs

The high speed memory board has two LEDs that are hard wired to parity and access circuitry. Interpret them as shown in Table 2-11.

Table 2-11. LED Interpretation

DS1	DS2	Interpretation
on	off	DS1 indicates the memory board is being accessed
off	on	DS2 indicates a parity error has been detected

† The VSB does not specify this slot and this setting would normally not be used.

2.12 PVSB Bus Signals on P2

The physical interface between the processor board and PVSB is through the P2 connector. The P2 connector is a standard 96 pin connector organized into three rows labeled A, B, and C. Rows A and C on the connector are reserved for high-speed memory. Row B is used for VMEbus signals according to VMEbus specifications[†].

Figure 2-1 shows pin assignments and signal mnemonics for the P2 connector.

Pin Number	Row A Signal Mnemonic	Row B Signal Mnemonic	Row C Signal Mnemonic
1	AD00	+5V	AD01
2	AD02	GND	AD03
3	AD04	—	AD05
4	AD06	R	AD07
5	AD08	E	AD09
6	AD10	S	AD11
7	AD12	E	AD13
8	AD14	R	AD15
9	AD16	V	AD17
10	AD18	E	AD19
11	AD20	D	AD21
12	AD22	GND	AD23
13	AD24	+5V	AD25
14	AD26	—	AD27
15	AD28	—	AD29
16	AD30	—	AD31
17	GND	F	GND
18	IRQ*	O	GND
19	DS*	R	GND
20	WR*	—	GND
21	SPACE0	—	SIZE0
22	SPACE1	GND	PAS*
23	LOCK	—	SIZE1
24	ERR*	V	GND
25	GND	M	ACK
26	GND	E	AC
27	GND	b	ASACK1*
28	GA0	u	ASACK0*
29	GA1	s	CACHE*
30	GA2	—	WAIT*
31	BGIN*	GND	BUSY*
32	BREQ	+5V	BGOUT*

Figure 2-9. Connector P2 Pin Assignments

[†] See *The VMEbus Specification HB212, Rev C.1*, from Motorola.

Section 3: VME-68K30 Addressing Space

3.1 Introduction

This section describes the address space seen by the VME-68K30 when it views the VMEbus. It also describes all the available PVSB space in an installation as viewed from devices on the VMEbus. In a cluster installation, devices on the VME include all clusters.

The logical and physical address space of a single VME-68K30 is shown in Figure 3-1.

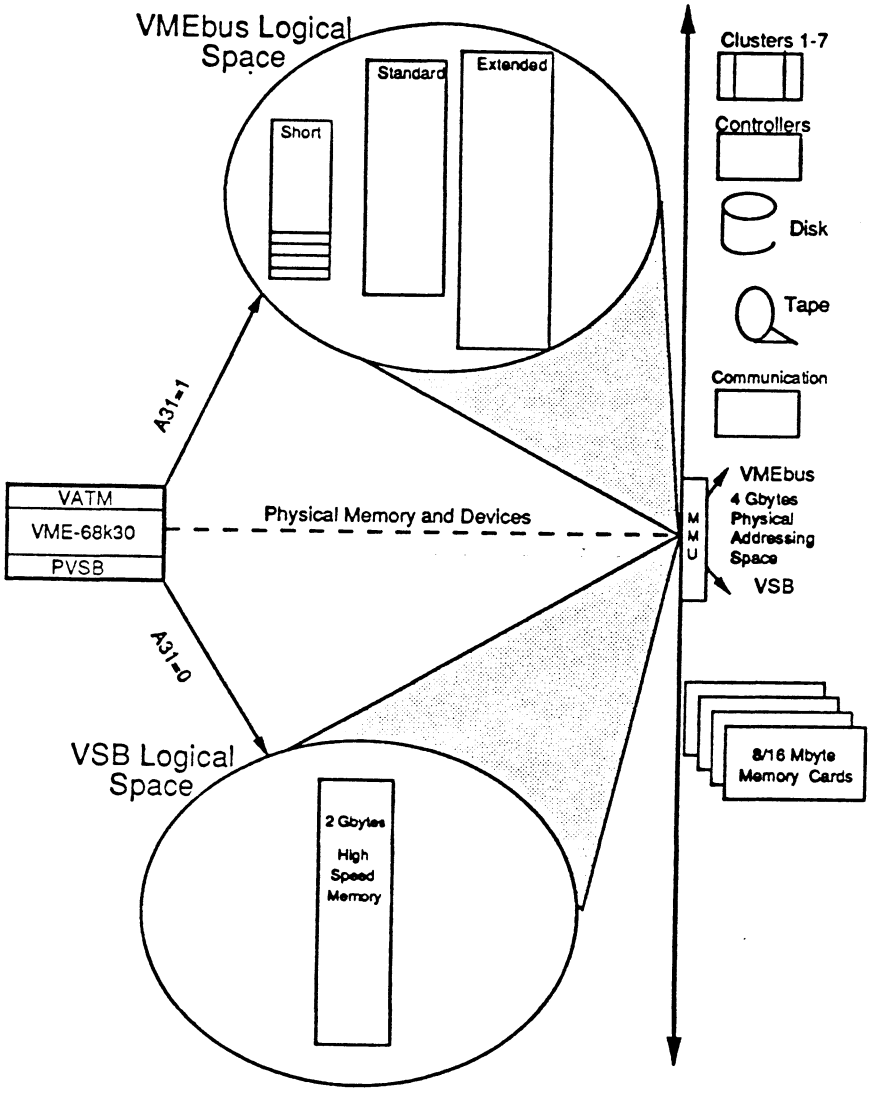


Figure 3-1. VME-68K30 Addressing Space

The 4 Gbyte physical addressing range of the MC68030 is equally divided between accesses to high speed memory when address bit A31=0 and accesses to VMEbus space when A31=1. If the function code is set to three, this division does not hold, but the exception is fully explained in Section 4.

The processor accesses devices on the VMEbus or the PVSB or it is accessed by devices from the VMEbus in one of four ways.

1. A processor has access to its own high speed memory in PVSB space.
2. A processor has access to any VMEbus device through the VATM map.
3. A cluster or any VMEbus device has virtual access to any PVSB memory through the VDMA map.
4. A cluster or any VMEbus device also has direct access to any PVSB memory through its cluster number.

Figure 3-2 illustrates these access methods. The arrows in the illustration show the direction of access. Maps on the VME-68K30 are illustrated in the arrows when they are required to gain access.

A fifth communication channel which combines several of the basic four occurs when a cluster requests the server to access a device on the VME. Figure 3-2 includes the cluster/server access combination.

Processor accesses through the VATM to VMEbus space is covered in this section while programming the VATM is described for Section 4 where all PVSB I/O registers are defined.

Processor access to ISI defined space was partially described in the preceding section and is fully described in Section 4, so it is not covered here.

Virtual DMA (VDMA) accesses are fully covered in this section including VDMA map programming.

Cluster access to VME devices through the server are similar to other VATM accesses from the server point of view. From the cluster's point of view, each request is initiated by writing to an area of memory shared by the cluster and server rather than to the device itself. Since this request is handled transparently by the kernel, it is not described in detail in this manual.

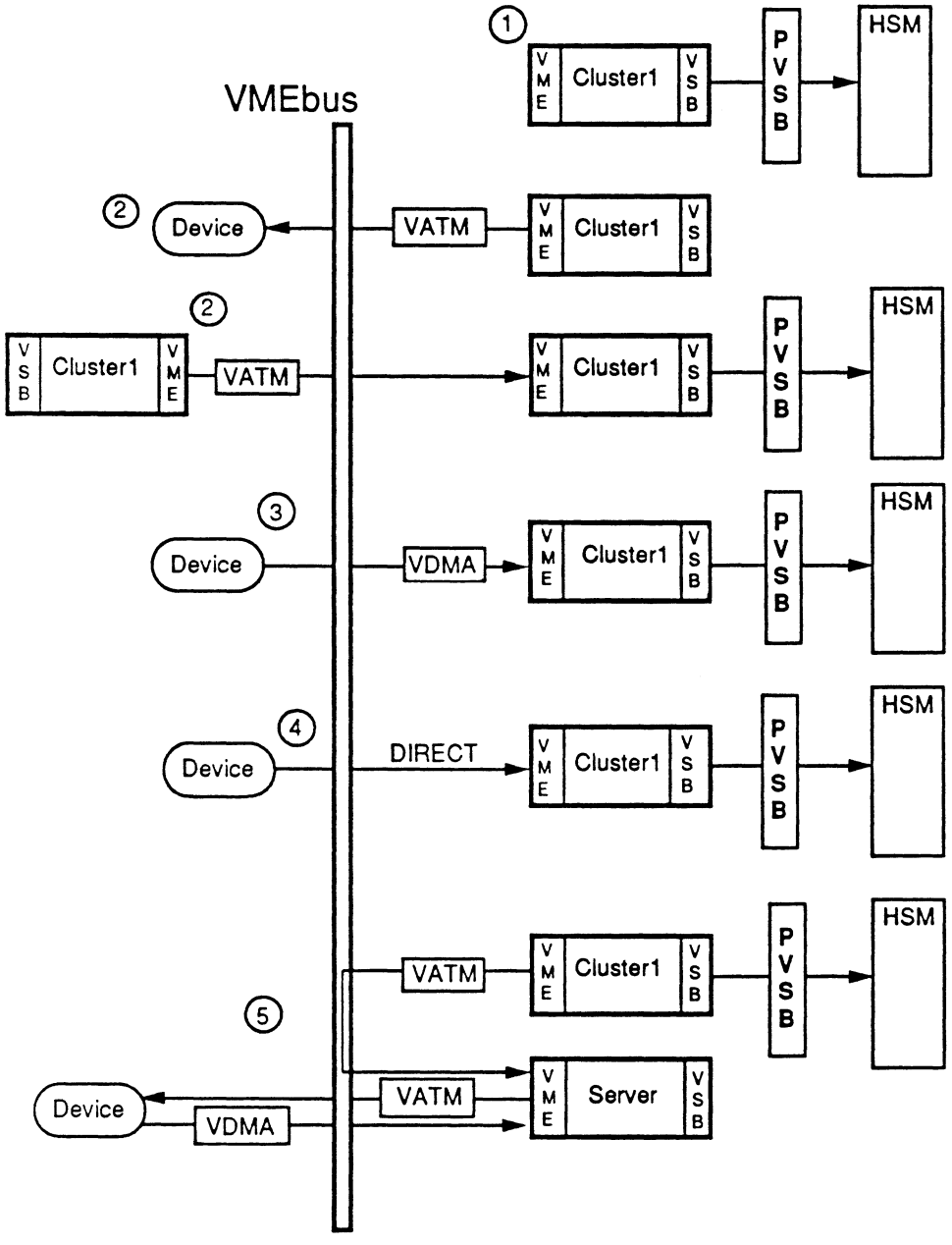


Figure 3-2. VME-68K30 Bus Access

3.2 Function Codes and A31

Function codes (FC2-0) replicate the 32-bit addressing range of the MC68030 through eight contexts. Each context, with the exception of CPU space identified by FC=7, is protected with a privilege level by the on-chip memory management unit (MMU). The integrity and exclusive use of each context is thus maintained. The microprocessor maintains its own exclusive use of the context addressed when FC=7 and the MMU is disabled.

Decoding logic on the VME-68K30 monitors function codes in conjunction with A31 to identify and direct accesses to user program and data space, supervisory program and data space, CPU space, and ISI defined space.

Motorola reserved function codes FC=0 and FC=4 are not discussed in this manual. The ISI use of other function codes is described in the following:

FC = 7 During exception processing, break point operations, and coprocessor cycles, the CPU accesses high speed memory without the memory management unit (MMU). A one to one (1:1) relation exists between the addresses generated by the CPU and physical address space in high speed memory when the CPU accesses this space.

FC = 1, 2, 5, and 6 User program and data space and supervisory program and data space are collectively called system space in this manual. System space is accessed when FC=1, 2, 4, or 5. In system space, address line A31 divides the 4 Giga byte addressing range of the MC68030 between the VMEbus and the PVSB. If A31 = 1, an access goes to the VMEbus. If A31=0, an access goes to the PVSB.

When the MC68030 accesses the VMEbus (A31 = 1) its local addresses are translated into VME addresses and address modifiers (AM0—5) by a VMEbus master address translation map (VATM). The VATM is programmable.

When A31-0 the MC68030 accesses the PVSB. Address lines A30-28 are used as an index into one of eight 256 Mbyte regions of cache write allocate or no write allocate space. Address lines A27-1 identify a single address in one of the eight regions.

FC = 3 Through ISI defined space, the CPU accesses on-board I/O, interrupts from the PVSB, and PVSB memory board registers. Using this space saves both hardware and standard address space, particularly where address translation maps are concerned. ISI defined space also contains a large (2 Gbyte) space reserved for cache invalidate accesses. Buffering data in a cache invalidate space maintains cache coherency.

A summary of the relation between function codes and A31 to addressing space is shown in Table 3-1.

Table 3-1. A31 and Function Codes FC0-2

Function Code				Address Line A31			
				Local and PVSB Bus			VMEbus
FC2	FC1	FC0	Context	A31=0	A31=1	Cache Control	A31=0
0	0	0	Motorola reserved, not used	—	—	—	—
0	0	1	User data, execution	—	PVSB	Allocate	System
0	1	0	User program, execution	—	PVSB	Allocate	System
0	1	1	ISI Defined	—	PVSB	Invalidate	—
				Local	—	—	—
1	0	0	Motorola reserved	—	—	—	—
1	0	1	Supervisor data, execution	—	PVSB	Allocate	System
1	1	0	Supervisor program, execution	—	PVSB	Allocate	System

3.3 CPU Space (FC=7)

CPU address space does not include instructions or operands. It is used by the processor to communicate with coprocessors or to acknowledge interrupts. In ISI installations, the MOVES instruction is limited to the ISI Defined Space, since its use in CPU space can interfere with system operation. For more details about this address space, see Section 4 of the *MC68030 User's Manual*, MC68030UM/AD from Motorola.

3.4 VATM Map (FC=1, 2, 5, 6)

Through the VMEbus Address Translation Map (VATM), the VME-68K30 accesses Extended, Standard, and Short I/O space on the VMEbus. Its own base address is also located in all three spaces.

The VATM has sixteen 12-bit registers that store address modifier codes AM5-0 and address lines VA 31-27. Address lines A26-2 are direct mapped onto the bus. As bus master, the processor generates all VMEbus AM codes, but in slave mode it does not decode AM0-3.

When acting as bus master, local addresses A30 through A27 become an index into the VATM which supplies AM5-0 and address lines VA 31-27 as shown in Figure 3-3.

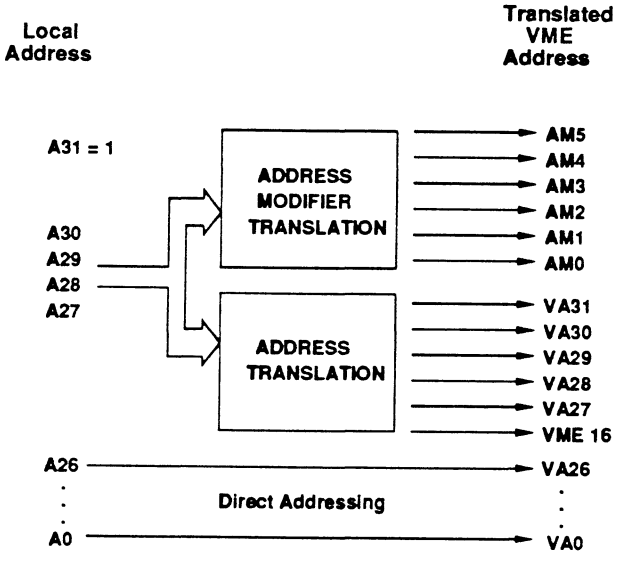


Figure 3-3. Local Access to VATM

3.5 Programming the VATM Map

The VTAM map located in ISI defined space and its registers and programming are defined with other registers in that space in Section 4.

3.6 Access to PVSB

Memory cards do not respond to multiple masters or provide full slave functionality. But any VMEbus master can access space on any PVSB memory card through the address lines, data paths, and mapping modules of its associated VME-68K30.

To access a VME-68K30 a DMA device must know its VME address or cluster number on the VMEbus. The cluster number is set by a rotary switch on a VME-68K30. Decode logic reads the rotary switch setting and interrupts the CPU when it detects a VME access. Setting the rotary

switch establishes the board's address in Short, Standard, and Extended space as shown in Figure 3-4.

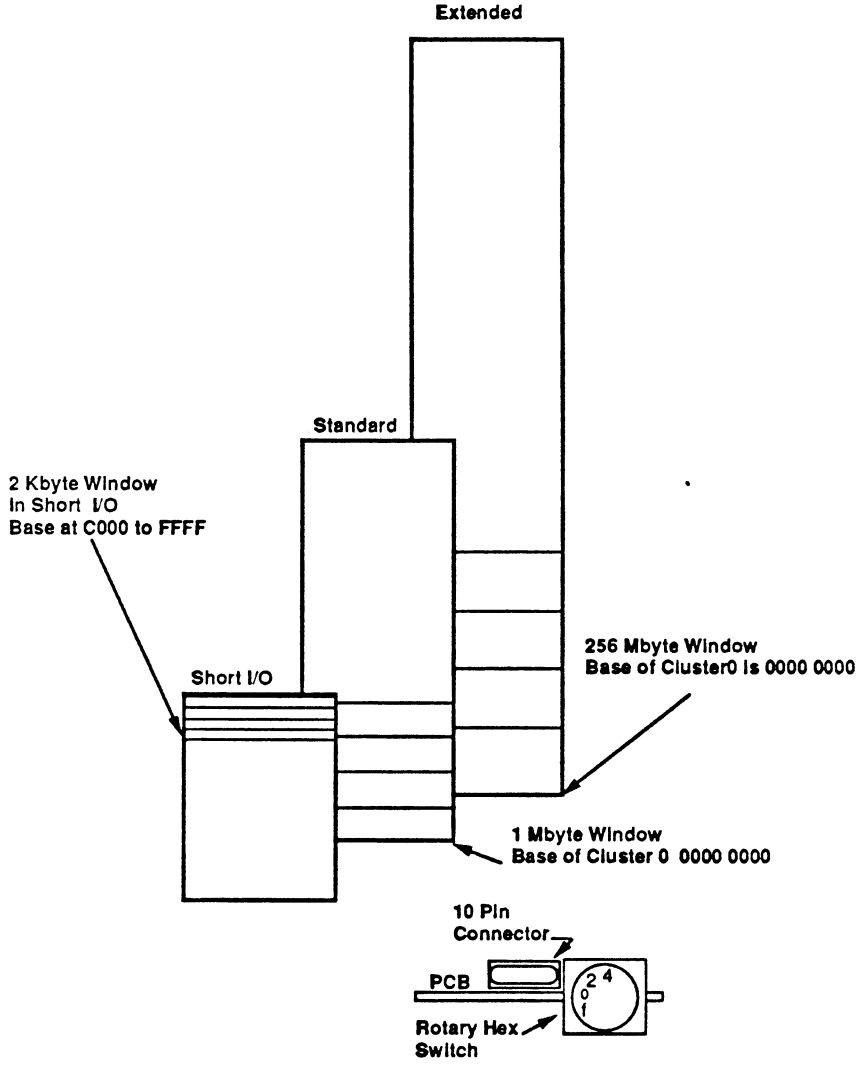


Figure 3-4. Base Address of Cluster

Clusters are numbered 0 — 7. Cluster 0 is the server. Table 3-2 shows the base address for each cluster in a system.

Table 3-2. Cluster ID In VME Standard and Extended Space

Cluster	Extended VME Window	Standard VME Window
0	[0-000_b]xxx xxxx	xx[0-000_b]x xxxx
1	[0-001_b]xxx xxxx	xx[0-001_b]x xxxx
2	[0-010_b]xxx xxxx	xx[0-010_b]x xxxx
3	[0-011_b]xxx xxxx	xx[0-011_b]x xxxx
4	[0-100_b]xxx xxxx	xx[0-100_b]x xxxx
5	[0-101_b]xxx xxxx	xx[0-101_b]x xxxx
6	[0-110_b]xxx xxxx	xx[0-110_b]x xxxx
7	[0-111_b]xxx xxxx	xx[0-111_b]x xxxx

3.7 Programming the VDMA Registers

The VDMA map is located in a 256 x 16 bit SRAM on the VME-68K30. With the function code set to 5, the VDMA registers are accessed in Short I/O in the addressing range from C000 to FFFF. The registers are programmed with the bit pattern shown in Figure 3-5.

Bit #	15-14	13-6	5-4	3-1
ID	Short I/O	1 of 256 Registers	Remote Reset/Interprocessor Interrupt	Cluster ID

Figure 3-5. VDMA Register Bit Pattern

The bit pattern in Figure 3-5 is interpreted as follows:

1. 15-14: Set high to access Short I/O at address C000.
2. 13-6: Selects one of 256 programmable registers in Short I/O. The registers are loaded with 16 data bits that translate PVSB addresses A/D 27-0.
3. 5-4: Bits 5 and 4 are decoded as follows:

Bit	5	4	Description
	0	0	Program VDMA
	0	1	Reset Board
	1	0	Interprocessor Interrupt
	1	1	Reserved

4. 3-1: The cluster number is selected by bits 3-1, with 000 selecting Cluster0 (the server) and so on.

After the VDMA map is programmed, it will allow virtual or extended access into high speed memory as summarized in Table 3-3.

Table 3-3. Direct and Virtual Translation Values

VME-68K30 Address Mode	VME Address Space	Address Modifier	Address ID	Cluster ID	Address Translation
Direct DMA	Extended	AM	A31=0	A30-28	LA27-12→PVSB A27-12 (Direct)
Virtual DMA	Extended	AM	A31=0	A30-28	LA19-12→PVSB A27-12 (Virtual)
Virtual DMA	Standard	AM	A23=0	A22-20	LA19-12→PVSB A27-12 (Virtual)

When viewed in extended address space from another cluster or DMA device on the VME, each cluster PVSB is a window into a 255 Mbyte block of contiguous memory. When viewed in standard address space, each cluster PVSB is a 1 Mbyte window into a 255 Mbyte block of contiguous memory. Direct memory accesses are only possible in extended space, while virtual accesses are supported in both standard and extended space. Virtual accesses are routed through the VDMA address translation maps. To directly access memory, only the cluster needs to be identified on the VMEbus, then access goes through the 255 Mbyte window on the VMEbus.

The upper 1 Mbyte of VMEbus Extended addressing space can only be accessed in virtual mode. Figure 3-6 shows the PVSB space available from a VMEbus master access.

3.8 Direct PVSB Access: Extended Space

Direct addressing into PVSB space requires that A31=0 and A30-28 identify a cluster. Address bits A27-0 then access the 256 Mbyte region in PVSB space from 0000-0000 to 0FFF-FFFF. This space is always allocated to write cache access. Figure 3-6 is a block diagram showing this access.

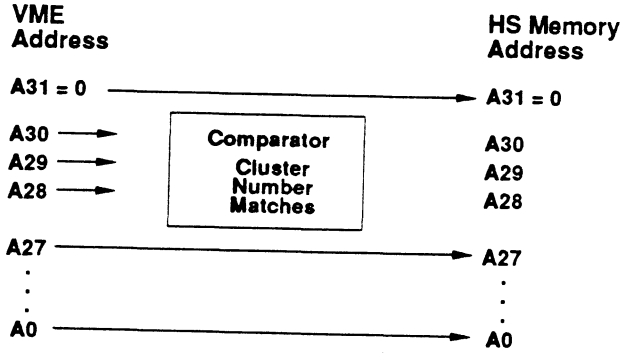


Figure 3-6. Direct Access from VME Master to PVSB

3.9 Virtual DMA Access to PVSB Memory

DMA addressing into PVSB memory can also be virtual. Virtual addresses use a translation map and therefore allow devices without VMEbus Extended Address capability to reach extended address space. Virtual addressing is supported in both Standard and Extended space.

3.10 Virtual PVSB Access: Standard Space

VMEbus masters that are only capable of accessing Standard space access PVSB space when A23=0 and a cluster number is identified by A22-20. VME address bits (VA19-12) are translated into PVSB bits A/D27-12. Bits within the 4 Kbyte page boundary VA11-0 bypass the map and are not translated. A block diagram of the map is shown in Figure 3-5.

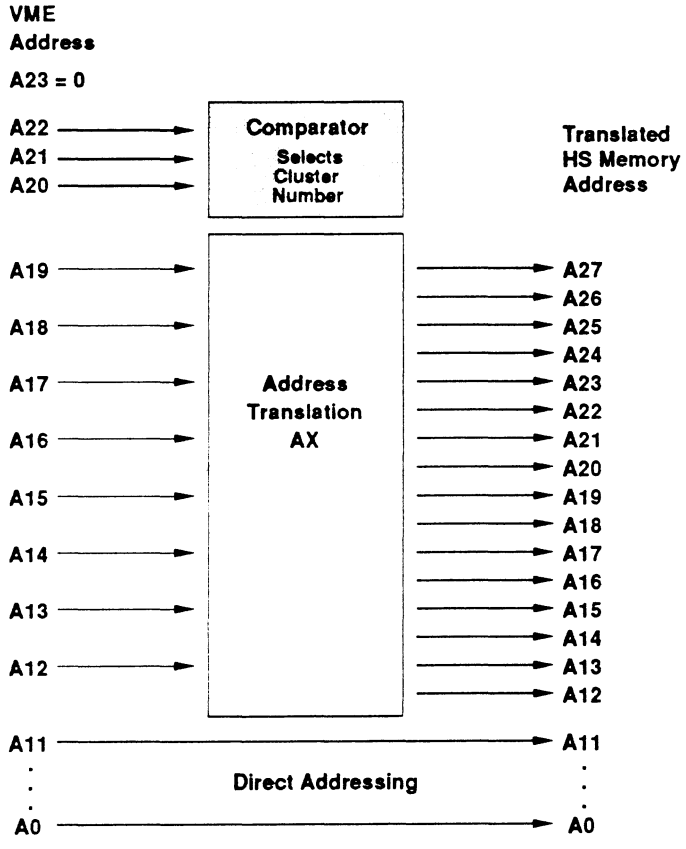


Figure 3-7. Standard Virtual DMA Access

3.11 Virtual PVSB Access: Extended Space

In Figure 3-6, if VMEbus address line A31 is 0 and A27-20 equal FF_H a DMA device accesses PVSB memory located in extended space. The address translation map, which is a programmable RAM array, maps A19-12 from the VME side into A27-12 on the PVSB side. This is the same translation hardware used in standard accesses.

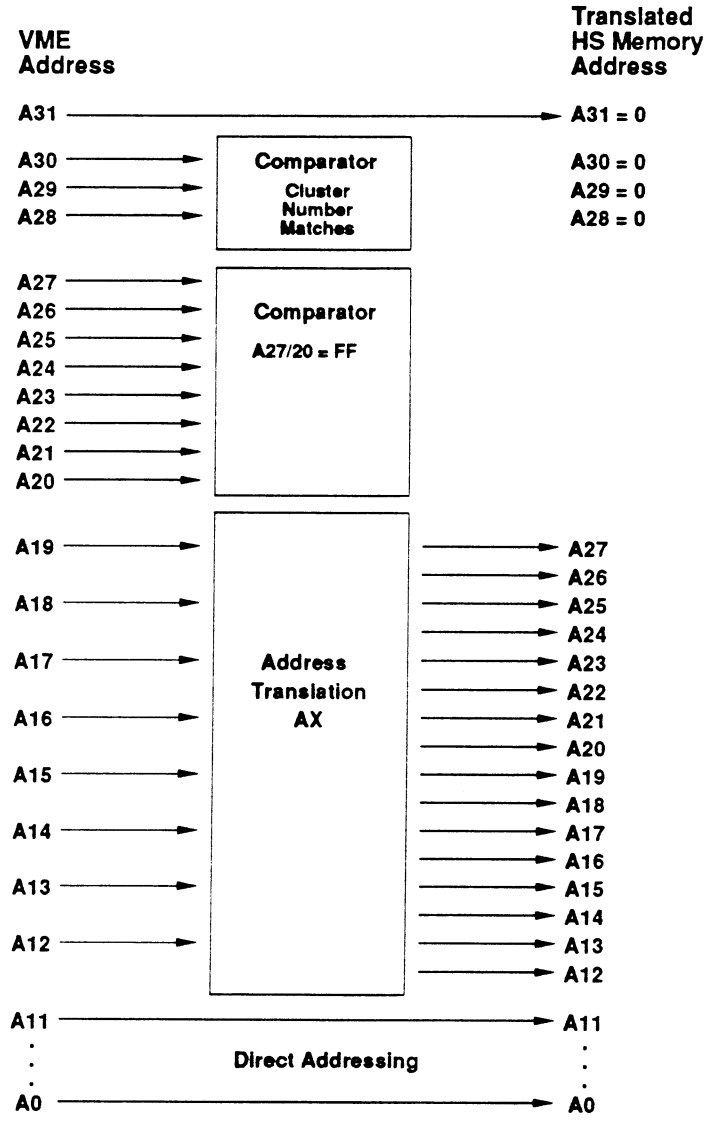


Figure 3-8. Extended Virtual DMA Access

Section 4: ISI Defined Space

4.1 Introduction

Set the MC68030 function code to 3 to access ISI defined space. On the VME-68K30, ISI has reserved this space for on-board control and status registers, for selected PVSB (Performance VSB) access, and for a 2 Gbyte cache maintenance buffer. These spaces are illustrated in Figure 4-1.

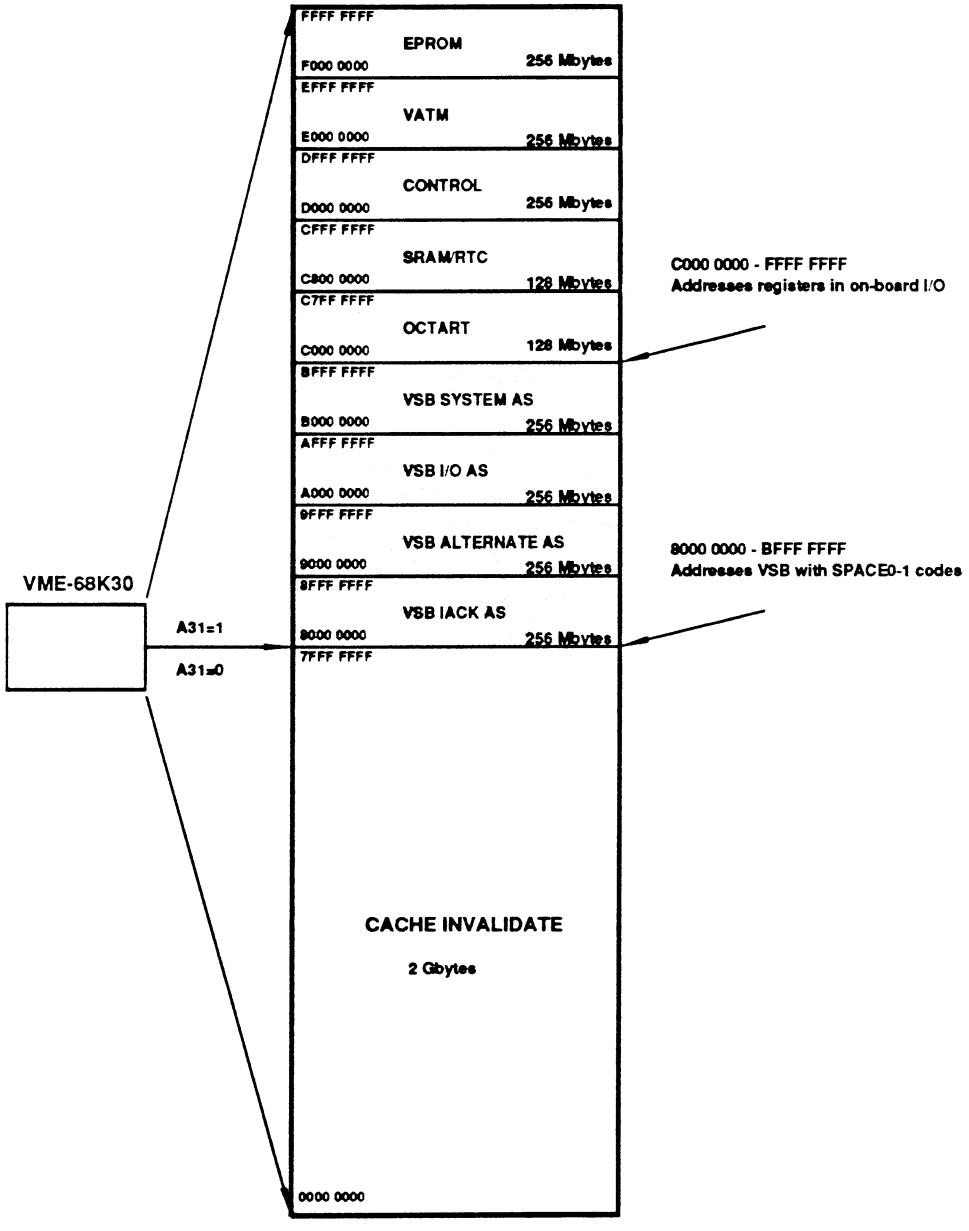


Figure 4-1. ISI Defined Space

4.2 ISI Defined Space 0000 0000 — FFFF FFFF

This section describes ISI defined space. The space contains registers for memory boards and local I/O devices. Registers in each region are described to the bit level. The space also contains a large cache maintenance buffer. The buffer is accessed when A31 is equal to zero rather than one.

4.3 Cache Invalidate Space 0000 0000 — 7FFF FFFF

Address 0000 0000 — 7FFF FFFF
Access R/W Word
Space 2 Gbyte

Writing to this space invalidates any cache tag entries that matches the written address as well as any other valid entries in the same cache tag line. Since data written to this buffer invalidates cache entries, the VME-68K30 uses it to maintain cache coherency when writing I/O devices or performing DMA transfers.

No data bits are significant at this 32-bit port.

For more cache control information, see subsection 7.3 in this section.

4.4 Register Access on the PVSB

When the following three conditions are true, VME-68K30 hardware accesses local I/O and memory registers in ISI defined space.

- The addressing range is between 8000-0000 and FFFF-FFFF
- Address line A31 is equal to 1
- The CPU function code is set to three (FC=3)

4.5 I/O & System Space 8000 0000 — BFFF FFFF

A range of addresses on the PVSB are qualified by two standard PVSB signals, SPACE1 and SPACE0. The decoding of the SPACE signals is presented in Table 4-1. Hardware supplies the appropriate SPACE code from PAL logic when it detects an access to the the address range 8000 0000 — BFFF FFFF, when the function code is 3, and when A31=1. The access is then directed to the appropriate PVSB space.

Table 4-1. PVSB SPACE0-1 Address Qualifiers

Space 0	Space 1	Description	Address
low	low	Not used	8000 0000-8FFF FFFF
low	high	Not used	9000 0000-9FFF FFFF
high	low	PVSB I/O Space	A000 0000 AFFF FFFF
high	high	System Space	B000 0000 BFFF FFF

4.5.1 PVSB IACK Space

Address 8000 0000 — 8FFF FFFF
Access R/W
Space 256 Mbyte

IACK space is not used by the VME-68K30.

4.5.2 PVSB Alternate Space

Address 9000 0000 — 9FFF FFFF
Access R/W
Space 256 Mbyte

Alternate space is reserved for block transfers. It is not used by the VME-68K30.

4.5.3 PVSB I/O Space

Address A000 0000 — AFFF FFFF
Access R/W
Space 256 Mbyte

This is a 32 bit port and data lines (D31-0) are significant. The lower 8 Mbytes of this space are reserved for PVSB memory board addresses. Also see Section 2 for register definitions.

4.5.4 PVSB System Space

Address B000 0000 — BFFF FFFF
Access R/W
Space 256 Mbyte

The port for this general purpose space is 32 bits wide and all data bits (D31-0) are significant.

4.6 On-Board Device Registers C000 0000 — EFFF FFFF

An address in the range C000 0000 to EFFF FFFF is mapped to registers physically located in on-board devices. These include an 8 port universal asynchronous receiver transmitter (OCTART), a scratch pad SRAM storage area with accompanying real time clock (RTC), a VMEbus Address Translation map (VATM), and control registers. On-board EPROM is located above these devices at F000 0000 to FFFF FFFF.

4.6.1 OCTART

Address C000 0000-C3FF FFFF
Access R/W Byte
Space 256 Mbyte
Reserved C000 0000-3F

Registers accessed in this space are physically located on the Signetics SCC2698 Octal UART (OCTART). This device contains eight asynchronous serial communication ports and four 16-bit counters. None of the counters are available for external use. The eight ports are configured for RS-232 signals. Port 7 supports full modem control with RS-232 signals RxD, TxD, RTS, CTS, DTR, DSR, and CD. Ports 0 - 6 signals include RxD, TxD, RTS, and CTS in DTE mode.

MC68030 data bits D31-24 are significant and correspond to OCTART data bits D7-0 respectively. Data bits D23-0 are undefined. Also refer to the SCC2698 data sheet from Signetics for register addresses and bit assignments.

Note

In the following four tables, the column labeled "Schematic" with Multi-Purpose Input and Multi-Purpose Output entries (MPI and MPO) refers to OCTART port designations on ISI schematics. They are not significant to the general reader or programmer.

Table 4-2 lists the registers in Port A.

Table 4-2. Input Port A. C000 000D

Bit	Schematic	Definition	Active
D 7	MPI3b	MC68030 Internal cache hit rate/4.	low
D 6	MPI2b	Cluster # Bit 2	low
D 5	MPI3a	Cluster # Bit 0	low
D 4	MPI2a	Cluster # Bit 1	low
D 3	MPI1b	General purpose DIP switch bit 5	low
D 2	MPI0b	Data carrier detect, channel 1	high
D 1	MPI1a	General purpose DIP switch bit 0	low
D 0	MPI0a	Data carrier detect, channel 0	high

D7 Internal cache hit/4 is used to build a cumulative record of cache hits

D6 Cluster # Bit 2 sets the cluster number (0-7) from the on-board rotary switch. The setting is written into D6, D5, and D4.

D5 Cluster # Bit 0

D4 Cluster # Bit 1

D3 Poles 5 and 4 on the general purpose DIP switch determine when the CPU will jump out of diagnostics during initialization. See the description of LED interpretation in Section 6 for switch interpretation.

D2 Data carrier detect, channel 1

D1 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 0.

D0 Data carrier detect, channel 0

Table 4-3 lists the registers in Port B.

Table 4-3. Input Port B. C000 001D

Bit	Schematic	Definition	Active
D 7	MPI3d	VME ACFAIL*	low
D 6	MPI2d	VME SYSFAIL*	low
D 5	MPI3c	VME Interrupt Enable	high
D 4	MPI2c	General purpose DIP switch bit 3	low
D 3	MPI1d	General purpose DIP switch bit 0	low
D 2	MPI0d	Data carrier detect, channel 3	high
D 1	MPI1c	60Hz clock tick	
D 0	MPI0c	Data carrier detect, channel 2	high

D7 VME ACFAIL*

D6 VME SYSFAIL* A system failure is indicated by the fourth light from the right on the VME-68K30. ISI systems do not set SYSFAIL* on during initial power up or after reset to indicate that controllers and CPUs are running internal diagnostics.

D5 VME Interrupt Enable

D4 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 3.

D3 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 0.

D2 Data carrier detect, channel 3.

D1 This 60Hz clock tick is tied to output channel A.

D0 Data carrier detect, channel 2.

Table 4-4 lists the registers in Port C.

Table 4-4. Input Port C. C000 002D

Bit	Schematic	Definition	Active
D 7	MPI3f	Geographic address bit 2	high
D 6	MPI2f	Geographic address bit 0	high
D 5	MPI3e	Geographic address bit 1	high
D 4	MPI2e	General purpose DIP switch bit 1	low
D 3	MPI1f	General purpose DIP switch bit 2	low
D 2	MPI0f	Data carrier detect, channel 5	high
D 1	MPI1e	Cache miss counter (Internal or External)	low
D 0	MPI0e	Data carrier detect, channel 4	high

D7 Geographic address bits (GA2-0) are used only with VME-68K30 cards designed for third party VSB memory cards. The GA bits have no significance for processors that use the PVSB memory cards and their jumpers must be left off.

D6 Geographic address bit 0

D5 Geographic address bit 1

D4 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 1.

D3 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 2.

D2 All general purpose DIP switches are defined in Section 6: Diagnostics. This register bit contains switch setting 5.

D1 The cache miss counter is used to monitor internal and external cache misses.

D0 Data carrier detect, channel 4 high

Table 4-5 lists the registers in Port D.

Table 4-5. Input Port D. C000 003D

Bit	Schematic	Definition	Active
D 7	MPI3h	PVSB IRQ bit	low
D 6	MPI2h	Floating point processor used bit	high
D 5	MPI3g	Interrupt enable	high
D 4	MPI2g	Software trap	high
D 3	MPI1h	MC68030 Interrupt pending	low
D 2	MPI0h	Data carrier detect, channel 7	high
D 1	MPI1g	External cache hit rate/4	low
D 0	MPI0g	Data carrier detect, channel 6	high

D7 The standard VSB signal IRQ* would normally signal an interrupt, but in ISI systems it indicates a parity error.

D6 The floating point processor is enabled when D6 is set, disabled otherwise.

D5 When interrupt enable is set high, interrupts from the VMEbus are disabled, but it does not affect local interrupts.

D4 The software trap bit sends a level 1 interrupt to the CPU.

D3 The MC68030 Interrupt Pending signal is a standard Motorola signal.

D2 Data carrier detect, channel 7.

D1 This supplies a counter tick divided by four for the external cache hit.

D0 Data carrier detect, channel 6.

The 16-bit counter timers on the OCTART are not available for external programming or use.

4.6.2 SRAM/Real Time Clock

Address C800 0000 - CBFF FFFF.
Access R/W Byte
Space 256 Mbyte

Data bits D31-24 are significant at this 8 bit port. Bits 23-0 are undefined. The real time clock combines a clock function and SRAM on the same chip. The SRAM chip has 2 Kbytes of storage, while the RTC occupies the last 8 bytes of SRAM space. The SRAM/ RTC is multiple mapped through this address range. Clock registers are defined in Table 4-6.

Table 4-6. Real Time Clock Register Addresses

Address	Access	Description
C800 07F8	R/W	Control Register
C800 07F9	R/W	Seconds
C800 07FA	R/W	Minutes
C800 07FB	R/W	Hour
C800 07FC	R/W	Day
C800 07FD	R/W	Date
C800 07FE	R/W	Month
C800 07FF	R/W	Year

Data bits in the clock status register are defined in Table 4-7.

Table 4-7. Status Register

Bit #	Data
D7	Not Used
D6	Not Used
D5	Not Used
D4	Not Used
D3	Implemented But Not Used
D2	HY93C46 SK Input (Clock)
D1	HY93C46 CS Input (Chip Select)
D0	HY93C46 Data Input

The *MOSTEK MK48T13-20 Data Sheet* provides detailed information on the SRAM/RTC device, including complete register definitions.

4.6.3 Control Register

Address D000 0000-DFFF FFFF
Access W Word
Space 256 Mbyte

All bits in this register are cleared to zero on reset and all active low bits are asserted (on) after a reset. Data bits D29-16 are significant. Data bits D31-30 and D15-0 are undefined. Data bits D26-24 are not used. The port is 16 bits wide.

Table 4-8. Control Register Addresses

Bit		Definition	Active
CPU	Reg		
D31	D15	Not implemented	
D30	D14	Not implemented	
D29	D13	Diagnostic Defined DS5	low
D28	D12	Diagnostic Defined DS6	low
D27	D11	Diagnostic Defined DS7	low
D26	D10	Implemented, not used	
D25	D9	Implemented, not used	
D24	D8	External Cache Enable	high
D23	D7	VME SYSFAIL*	low
D22	D6	FPUSED	high
D21	D5	PROM Enable	high
D20	D4	Cache Clear	high
D19	D3	VME Reset Generation	high
D18	D2	VME IE	high
D17	D1	IE	high
D16	D0	Software trap	high

Entries in the Write Control field are defined as follows:

D29 LED7 — This LED and LED5 and LED6 are defined by ISI diagnostics. See Section 6 of this manual for diagnostic status displayed through the LEDs.

D28 LED6 — Reserved for ISI diagnostics

D27 LED5 — Reserved for ISI diagnostics

D26 — Implemented not used

D25 — Implemented not used

D24 CACHEN — This bit enables the external cache.

D23 SYSFAIL* — The SYSFAIL* indicator is the green LED on the VEM-68K30. Some boards on the VMEbus issue SYSFAIL* until their internal diagnostics are complete. The main system processor (Cluster0 or Server) knows the system is operational when all boards have set SYSFAIL* high. Not all ISI boards have this feature, but all VME-68K30s do.

† When active, this bit indicates the floating point coprocessor is in use.

D22 FPUSED	This bit sets a semaphore on the FPU and prevents processes from accessing it. If a second process attempts to use the FPU when FPUSED is set, it receives a bus error.
D21 PROMEN	The PROM is at location 0000 0000 1FFF FFFF when this bit is low, for instance, when the board is reset. When the bit is one, the PROM is at its normal operating location. (See map of ISI defined space, Figure 4-1.)
D20 Cache Clear	Setting this bit clears all cache entries. The cache must be disabled before it is flushed, then enabled again before this bit is turned off.
D19 VME Reset	If this bit is enabled on the server (Cluster0) it will reset every board in the system except the server. It must be used with caution to avoid resetting devices while in undefined states.
D18 VME IE	Setting this bit enables VMEbus interrupts only
D17 IE	Setting this bits enables all local and VME interrupts.
D 16 Software trap	

4.6.4 VMEbus Master Address Translation Map (VATM)

Access W Word
Address E000 0000-EFFF FFFF
Space 256 Mbyte

All accesses from the VME-68K30 to the VMEbus are made through the VATM map. The map consists of sixteen 16-bit registers of on-board SRAM that are loaded through VATM space on the PVSB.

When the map is used, address bits A30-28 select one of sixteen registers in the map and address modifiers AM5-0 and address lines A31-27 are placed on the VMEbus from the map. Address lines A26-0 are written "through" with the mapped address lines and modifiers to make up the complete VMEbus address. The relation between programming bits and registers in the VATM is summarized in Table 4-9.

A "1" in bit position D16 causes the VMEbus to respond as a 32 bit port. Otherwise, the port is 16 bits wide.

Table 4-9. VME Address Translation Register

Bit	Definition	Register
D31	Not implemented	—
D30	Not implemented	—
D29	Not implemented	—
D28	Not implemented	—
D27	VMEbus Address Modifier #5	R11
D26	VMEbus Address Modifier #4	R10
D25	VMEbus Address Modifier #3	R09
D24	VMEbus Address Modifier #2	R08
D23	VMEbus Address Modifier #1	R07
D22	VMEbus Address Modifier #0	R06
D21	VMEbus Address Bit #31	R05
D20	VMEbus Address Bit #30	R04
D19	VMEbus Address Bit #29	R03
D18	VMEbus Address Bit #28	R02
D17	VMEbus Address Bit #27	R01
D16	VME 16 Bit Port Enable	R00

4.6.5 Programming the VATM

The VATM map is programmed by loading its registers beginning at E000 0000 and extending to E000 00FF. One of sixteen programming registers located in this region is selected by address lines A4-1 during programming. The first three register addresses are shown in the following table. The sequence is repeated for all sixteen registers.

VATM Address Decode				
A4	A3	A2	A1	VATM Register
0	0	0	0	E000 0000
0	0	0	1	E000 0002
0	0	1	0	E000 0004

The registers are loaded with data bits D27-16 as shown in Figure 4-2.

When data bit 16 is set, the map is programmed to access devices with 16-bit data paths. Otherwise it is set for a 32-bit path.

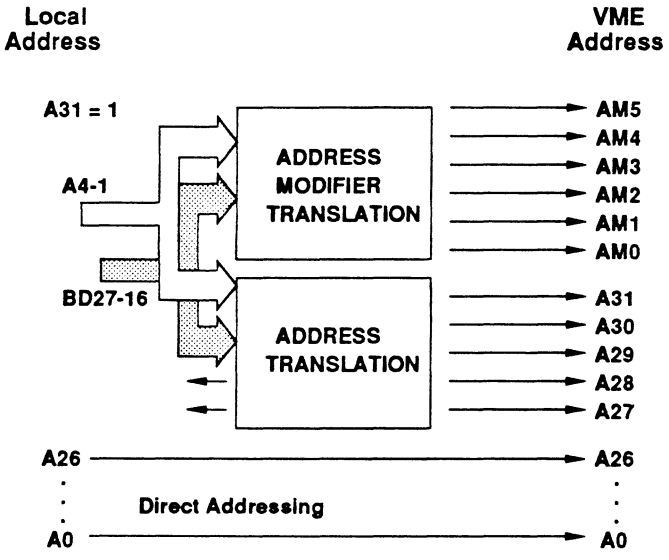


Figure 4-2. Loading VATM Registers

4.6.6 EPROM

Address F000 0000-FFFF FFFF
Access R Byte
256 Mbyte

Data bits D31-24 are significant. Data bits D23-0 are undefined. This is an 8 bit port.

Section 5: Configuration

5.1 Introduction

The purpose of this section is to define the hardware jumpers that configure the VME-68K30 board for operation in single and multiprocessor environments and to identify pinouts on connectors.

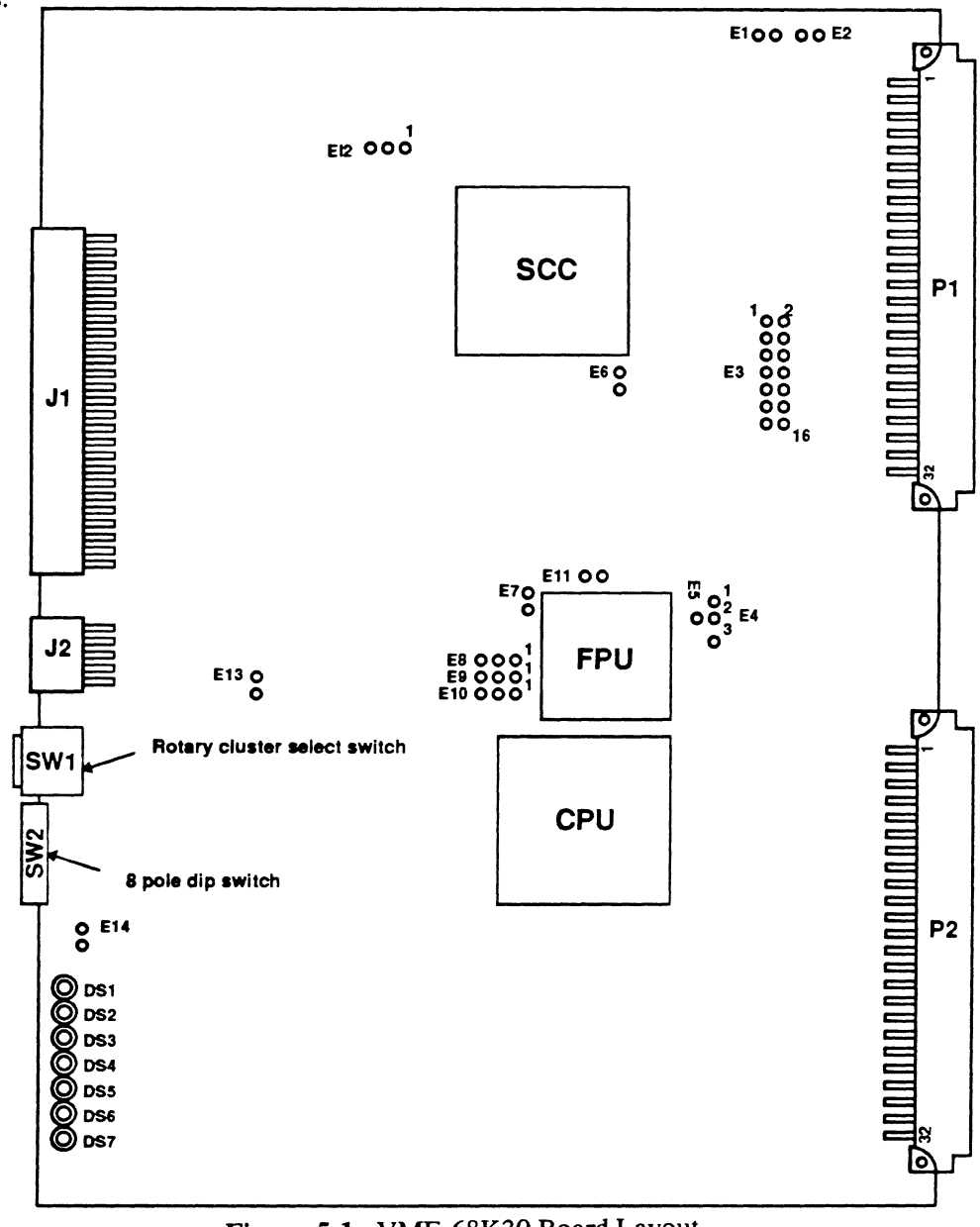


Figure 5-1. VME-68K30 Board Layout

5.2 Jumper Selections

The jumpers and switches shown in Table 5-1 are listed in numerical order. Use Table 5-1 to identify the jumpers associated with various on-board or interface functions, then turn to the referenced subsection for instructions on setting the jumper or switch.

Table 5-1. Jumper Number and Function

Jumper #	Section	Definition
E1-E2	5-1-1	Bus Request/Grant Level Select
E3	5-1-2	VMEbus IRQ1-7
E4-E5	5-1-3	Floating Point Clock Speed
E6	5-1-4	PROFILE
E7	5-1-5	MMU Disable
E8-E9-E10	5-1-6	EPROM Size Configuration
E11	5-1-7	Cache Disable
E12	5-1-8	VME Arbitration
E13	5-1-9	External Reset
E14	5-1-10	Watchdog Timer
SW1	5-1-11	Set Cluster Number
SW2	5-1-12	General Purpose Switch (Defined by Firmware)

5.2.1 E1-E2: Bus Request/Grant Level Select

The VMEbus Request/Grant level is selected by jumpers E1 and E2, for any one of the four levels allowed by the VMEbus specification. Install jumpers as shown in Table 5-2 to set the VME-68030 bus request and grant levels.

Although there are several options in the VMEbus specification for prioritizing bus usage all ISI systems implement the priority (PRI) request/grant scheme, with level 3 requests receiving the highest selectable priority during bus arbitration. ISI defaults are shown in Table 5-2, and default settings work well in most ISI installations. However, there is no reason why setting each cluster at a different level, or all clusters at levels different from the ISI default would not work to advantage in some installations. One rule must be observed: on each individual board, the level of bus request and the level of bus grant must be the same value, or the board cannot be notified when it has been granted the bus.

The basic criteria for evaluating the performance of different bus grant and request levels in any system is equitable distribution of bus usage among all processes. If one or several processes in a system seem unduly slow, revising the bus request scheme may improve overall system performance by making a more equitable distribution of the bus as a system resource.

One other rule applies. The system master (Cluster0) should be located in the lowest card slot in the system and must be located so no other cards that act as bus master are located below it. This requirement is established by the hardware implementation of the daisy chain that propagates the granting and requesting signals over the bus. It cannot be violated.

Table 5-2. VMEbus Request/Grant Level E2 - E1

E1	E2	Bus Request/Grant Level
Open	Open	Level 0
Open	1-2	Level 1
1-2	Open	Level 2
1-2	1-2	Level 3

ISI does not suggest Bus Grant/Request defaults for the server (cluster 0) and clusters in an installation.

5.2.2 E3: VMEbus Interrupt Request Level IRQ1-7

The interrupt request level is set between 1 and 7 with jumper E3. The jumpers and corresponding request level are presented in Table 5-3. The VME-68K30 allows software to disable jumper selected interrupts by setting the interrupt enable (INTEN) bit in OCTART control port C to off. The VMEbus interrupt enable bit in port B controls VMEbus interrupts.

The interrupt pending bit in port C gives software control of the MC68030 interrupt pending feature. For more information on programming this bit for proper operation of the microprocessor, see the *MC68030 User's Manual* MC68030UM/AD, Sections 5-8, 8-11, and 8-13 from Motorola. All control bits are described in greater detail in Section 4 of this manual.

The same PAL logic that controls on-board and VMEbus generated interrupts also generates the auto-vectored interrupt signal (AVEC) which instructs the microprocessor to autovector local interrupts.

Three interrupt sources generate a level 7 interrupt to the MC68030. They are ACFAIL, HSIRQ6 (generated by the memory board when a parity error is detected), and VME SYSFAIL.

Table 5-3. VMEbus Interrupt Request Level (IRQ1-7) JP4

Jumper	Bus Request Level	Server Default	Cluster Default
E3 1-2	IRQ7	in	out
E3 3-4	IRQ6	in	out
E3 5-2	IRQ5	in	out
E3 7-2	IRQ4	in	out
E3 9-2	IRQ3	in	out
E3 11-2	IRQ2	in	out
E3 13-2	IRQ1	in	out

5.2.3 E4 E5: Floating Point

The CPU board will accept either the 68881 or the 68882 floating point unit. The 68882 is standard. The clock rate for the floating point co-processor is set by jumpers E4 and E5. Settings are defined in Table 5-4. Jumpers installed as shown in the first column select the designated clock speed. The default is 25 MHz.

Table 5-4. Floating Point Clock Speed Jumpers

Jumper	FPU Clock Speed
E4 1-2	25MHz
E4 3-2	12.5MHz
E5 1 E4 2	16MHz

5.2.4 E6: PROFILE

PROFILE is not implemented. Leave E6 open at all times or your system may crash.

5.2.5 E7: MMU Disable

Installing a jumper on E7 asserts the MMUDIS* input signal to the MC68030 and disables the on-chip MMU. The assertion of MMUDIS* does not flush the address translation cache (ATC) so all entries are available when the line is deasserted. See the Motorola *MC68030 Enhanced 32-Bit Microprocessor User's Manual*, part number MC68030UM/AD, for a description of the use of this line with an emulator.

Leave this jumper out.

5.2.6 E8 E9 E10: EPROM Size Configuration

Three sets of jumpers (each with three individual posts) allow selection of various sized EPROMs from 32K to 512K of long word (4-byte) storage. The jumpers used to set the size of on-board EPROM are shown in Table 5-5.

ISI systems are shipped with 128K bytes of EPROM.

Table 5-5. On-Board EPROM Size

# Bytes	EPROM Size	E8	E9	E10
32K	27256	1-2	1-2	1-2
64K	27512	1-2	1-2	2-3
128K	**271024**	**1-2**	**2-3**	**2-3**
256K	272048	1-2	2-3	2-3
512K	274096†	2-3	2-3	2-3

5.2.7 E11: Cache Disable

The MC68030 cache inhibit input (CIIN*) is asserted (low) when jumper E-11 is installed. When asserted, this MC68030 signal prevents data from being loaded in the on-chip instruction and data caches.

† The 274096 is not currently available.

The ISI default is to leave this jumper out.

5.2.8 E12: VME Arbitration

Jumper E12 determines how a VME-68K30 will relinquish the bus when it is bus master. If no jumper is installed, the VME-68K30 responds as a release when done (RWD) master. If the jumper is installed on pins 1-2 the response is to release on request (ROR). If a jumper is installed on pins 2-3, the response is to release the bus when receiving a bus clear (BCLR) signal. BCLR is a standard VMEbus signal. It is issued by a system master when a device requires the bus and has a higher priority than the device using the bus. Since not all ISI boards capable of being master decode the BCLR signal, it is not used in ISI systems.

None of the requests for bus use will force a master from the bus; all allow the master to finish a transfer, or interrupt it at a recoverable point, before stopping its use of the bus. If jumpered for RWD, a master relinquishes the bus after every transfer, which may force it to arbitrate unnecessarily during multiple transfers, but leaves the bus open to multiple use. If jumpered for ROR, a master holds the bus even if it has no use for it until a request arrives. This allows multiple transfers from one device, but necessitates arbitration and the bus is never idle. Selection of one release protocol over another is a matter of bus traffic or in some cases prejudice. The ISI default is RWD which leaves the jumper out.

5.2.9 E13: External Reset

Jumper pins E13 are provided as stake pins for hard wiring an external reset switch to the VME-68K30 board. Pin E13-2 is ground.

5.2.10 E14: Watchdog Timer

Install a jumper at E14 to enable a watchdog reset on HALT. Without the jumper HALT is terminal. The default is to leave the jumper out.

5.3 SW1 Rotary Switch: Set Cluster Number

The rotary switch sets a cluster number on the VME-68K30, and selects VME slave addresses. Assign each cluster a unique cluster number with the switch settings shown in Table 5-6.

The server in a system is set to cluster 0. It performs I/O access for all other clusters in the system as well as VMEbus master arbitration.

Table 5-6. Cluster Select Switch

Switch Setting	Cluster Number
0	Cluster 0
1	Cluster 1
2	Cluster 2
3	Cluster 3
4	Cluster 4
5	Cluster 5
6	Cluster 6
7	Chapter 7

In single processor systems, the default setting is Cluster 0.

Settings on the switch are clearly marked, but the index is partially hidden by the adjoining 10 pin connector. The index is located clockwise 270° from vertical. The setting for a board as cluster 0 (the server) is shown in Figure 5-2, with indicators for clusters 2, 4, and f shown as reference. For a map of board addresses, see Section 1.

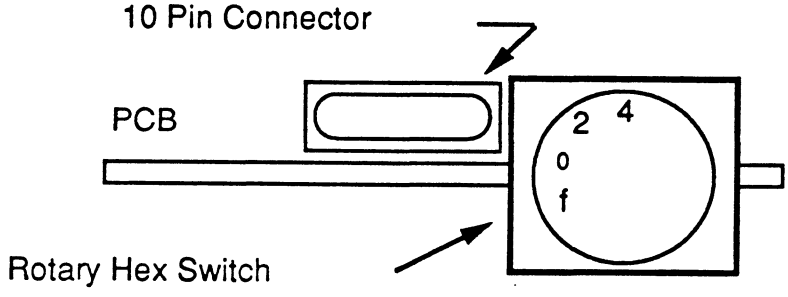


Figure 5-2. Setting The Rotary Switch (SW2) for Clusters

5.4 General Purpose Switch Module

A six position general purpose switch (SW-2) selects firmware defined options. The bits are set in the board status register in the OCTART. The definition of bits set in this switch are summarized in Table 5-7 and described in the following list.

Table 5-7. General Purpose Register Switch

General Purpose Switch						
Boot		Res.	Baud Rate			Definition
SW2-5	SW2-4	SW2-3	SW2-2	SW2-1	SW2-0	
off	off					System Boot
off	on					Reserved
on	off					PROM Diagnostics
on	on					Alternate PROM Diagnostics
		off				Reserved
		on				Reserved
			off	off	off	2400 Baud
			off	off	on	Reserved
			off	on	off	Reserved
			off	on	on	19200
			on	off	off	Reserved
			on	off	on	9600
			on	on	off	1200
			on	on	on	Reserved

System Boot — The default setting is to boot the system after power up when the system is first turned on or when the reset button has been pressed. The options available after booting the system are described in Section 6 of this manual.

PROM Diagnostics — Setting GP5 on and GP4 off causes the system to go to the monitor before it checks system memory. This setting is used to check on-board hardware from the monitor without testing system memory.

Alternate PROM — Setting both GP5 and GP4 on goes to the monitor after a preliminary test of memory.

Reserved

Baud Rate — The baud rate settings are self-explanatory. Set the rate according to whether you are using a WorkStation (1200 baud) or terminal (9600 baud) with your system.

5.5 LED Indicators

LED indicators have the assignments shown in Table 5-8.

Table 5-8. LED Indicators

LED	Description
DS1	CPU Halt
DS2	VME to PVSB access (yellow)
DS3	MC68030 to VME access (green)
DS4	MC68030 system failure (SYSFAIL) asserted
DS5	Diagnostic encoded
DS6	Diagnostic encoded
DS7	Diagnostic encoded

The diagnostic encoding is described in Section 6.

5.6 Connectors

There are four sets of connectors on the VME-68K30 board (see Figure 5-1). The P1 connector carries VMEbus signals to the backplane. Rows A and C on the P2 connector carry PVSB signals, while Row B carries VMEbus signals, per the VMEbus specification.

The J1 and J2 connectors carry serial RS-232-C signals to external devices. J1 is the 50 pin connector that carries signals from the 8 channel OCTART. J2 is a ten pin connector that carries RS-232-C signals to the system console.

In all ISI systems, the VME-68K30 is factory shipped so that two channels from the OCTART are available at the rear break out panel. Channel 0 is wired to the "Console" connector and channel 1 is wired to "Port 0". The connectors are labeled on the rear break out panel of all systems.

All eight channels can be ordered from ISI to come out through the 50 pin connector J1, and be internally wired from J1 on the VME-6830 to ports on the system break out panel. When this is the configuration Channel 0 and Channel 1 are duplicated on J1 and J2. They are therefore mutually exclusive.

5.6.1 Signal Line States

A signal line is always in one of two levels or in transition between these levels. Whenever the term *high* is used, it refers to a high TTL voltage level ($\geq$ + 2.0 V). The term *low* refers to a low TTL voltage level ($\leq$ + 0.8 V). A signal line is *in transition* when its voltage is moving between +0.8 V and +2.0 V.

Two possible transitions, called edges, can appear on a signal line. A rising edge is defined as the time period during which a signal line makes its transition from a low level to a high level. The falling edge is defined as the time period during which a signal line makes its transition from a high level to a low level.

5.6.2 VMEbus Connector P1

Connector P1 carries VMEbus signals including all seven interrupt levels and four-level bus arbitration. Addressing is generated in 16-bit (short), 24-bit (standard), or 32-bit (extended) addresses on the VMEbus. The VMEbus is implemented with pinouts on rows A, B, and C on P1 and on row B on P2 as shown in Figures 5-3 and 5-4.

Pin Number	Row A Signal Mnemonic	Row B Signal Mnemonic	Row C Signal Mnemonic
1	D00	BBSY*	D08
2	D01	BCLR*	D09
3	D02	ACFAIL*	D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK	BG3IN*	SYSFAIL*
11	GND	BG3OUT*	BERR*
12	DS1*	BR0*	SYSRESET*
13	DS0*	BR1*	LWORD*
14	WRITE*	BR2*	AM5
15	GND	BR3*	A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20	IACK*	GND	A18
21	IACKIN*	SERCLK	A17
22	IACKOUT*	SERDAT	A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	IRQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31	-12V	+5V STDBY	+12V
32	+5V	+5V	+5V

Figure 5-3. Connector P1 Pin Assignments

NOTE

An asterisk (*) following the signal name denotes that the signal is asserted when low. An asterisk following the signal name for signals that are *edge* significant denotes that the actions initiated by that signal occur on a high to low transition.

5.6.3 PVSB Bus Signals on P2

Figure 5-4 shows pin assignments and signal mnemonics for the P2 connector.

Pin Number	Row A Signal Mnemonic	Row B Signal Mnemonic	Row C Signal Mnemonic
1	AD00	+5V	AD01
2	AD02	GND	AD03
3	AD04	RESERVED	AD05
4	AD06	VA 24	AD07
5	AD08	VA 25	AD09
6	AD10	VA 26	AD11
7	AD12	VA 27	AD13
8	AD14	VA 28	AD15
9	AD16	VA 29	AD17
10	AD18	VA 30	AD19
11	AD20	VA 31	AD21
12	AD22	GND	AD23
13	AD24	+5V	AD25
14	AD26	VD 16	AD27
15	AD28	VD 17	AD29
16	AD30	VD 18	AD31
17	GND	VD 19	GND
18	IRQ*	VD 20	GND
19	DS*	VD 21	GND
20	WR*	VD 22	GND
21	SPACE0	VD 23	SIZE0
22	SPACE1	GND	PAS*
23	LOCK	VD 24	SIZE1*
24	HSERR*	VD 25	GND
25	GND	VD 26	ACK
26	GND	VD 27	AC
27	GND	VD 28	ASACK1*
28	GA0	VD 29	ASACK0*
29	GA1	VD 30	CACHE*
30	GA2	VD 31	WAIT*
31	BGIN*	GND	BUSY*
32	BREQ	+5V	BGOUT*

Figure 5-4. Connector P2 Pin Assignments

NOTE

Since the VME-68K30 supports 64 Mbytes of physical address space, the six high-order address bits are always set to 0 on 32-bit addresses.

5.6.4 PVSB Bus Signal Definitions

PVSB signals are a subset of standard VBS signals. The ISI implementation of which are described in Section 2.

5.6.5 Serial Port Connectors

The card edge connector J1 carries control and data signals from the eight port OCTART to external devices. Pin assignments are given in Figures 5-5 and 5-6.

J1			
Pin Number	Signal Mnemonic	Pin Number	Signal Mnemonic
1	DTR 0	26	GND
2	DCD 0	27	TxD 2
3	DTR 1	28	
4	DCD 1	29	RxD 2
5	DTR 2	30	GND
6	DCD 2	31	TxD 3
7	DTR 3	32	
8	DCD 3	33	RxD 3
9	DTR 4	34	GND
10	DCD 4	35	TxD 4
11	DTR 5	36	
12	DCD 5	37	RxD 4
13	DTR 6	38	GND
14	DCD 6	39	TxD 5
15	DTR 7	40	
16	DCD 7	41	RxD 5
17	GND	42	GND
18	GND	43	TxD 6
19	TxD 0	44	
20		45	RxD 6
21	RxD 0	46	GND
22	GND	47	TxD 7
23	TxD 1	48	
24		49	RxD 7
25	RxD 1	50	GND

Figure 5-5. Connector J1 Pin Assignments

Encoding for Figure 5-5 is as follows:

DTR Data Transmit Ready
DCD Data Carrier Detect
RxD Receive Data
TxD Transmit Data
GND Ground

The remaining signals for the serial ports are carried on the J2 connector with the pin assignments presented in Figure 5-6.

J2	
Pin Number	Signal Mnemonic
1	TxD0
2	RxD0
3	—
4	DCD1
5	GND
6	GND
7	TxD 1
8	RxD 1
9	DTR 1

Figure 5-6. Connector J2 Pin Assignments

5.7 VMEbus Specification

Master Data Transfer Options

A16/A24/A32 D16/D32

Arbiter Options

PRI BG3-0 Set by jumper

The MC68030 initiates an access and times out in 50 micro seconds if no response is received.

TOUT = 50 micro seconds, fixed

Release Options

RWD or ROR (set by jumper)

Interrupt Handler Options

Interrupt levels 1-7 (Static, but enabled by software at all levels)

Interrupter

Auto vectored

5.8 Environmental Specifications

The environmental requirements for the VME-68K30 are as follows:

Operating temperature (ambient)	0° to 35°C -32° to 95°F
Storage temperature	-40° to 65°C -104° to 149°F
Relative humidity	20 percent to 80 percent (non-condensing)

Section 6: Monitor and Diagnostics

6.1 Introduction

This section describes the diagnostic capabilities and user interface to the VME-68K30.

6.2 Capabilities

The VME-68K30 CPU card provides the following basic capabilities:

1. Power-on/reset initialization
2. User interface for loading and executing software
3. Simple memory read/write/execute functions
4. Embedded diagnostics

6.3 Power-on/Reset Initialization

The MC68030 CPU performs the following power-on diagnostics in normal boot (non-diagnostic) mode:

System initialization
    program OCTART for console port I/O
    size memory and initialize to 0

Power-on diagnostics
    check timer
    check clock
    check VDMA/VATM

Clear all software-defined LEDs

Print a banner on the console port

Enter interactive monitor

During the power up initialization, the LEDs display a code to indicate the progress of various initialization stages. The code remains displayed until the next phase of power up is entered. If a failure occurs, the display on the LEDs indicate where the power up initialization failed. LED DS4 is the SYSFAIL indicator. LEDs DS5, 6, and 7 are encoded for diagnostics. The encoding of the LEDs is presented in Figure 6-1

	LED				Diagnostic	SW1 Setting
	DS4	DS5	DS6	DS7		
1.	on	on	on	on	ROM Diagnostic Debugger	
2.	on	off	off	on	Initialize OCTART	
3.	on	off	on	off	Read Board Status Register	
						if-> SW1-5 SW1-4*-> Monitor
4.	on	off	on	on	Initialize Keyboard	
5.	on	on	off	off	Probe VSB Memory	
						if-> SW1-5 SW1-4 -> Monitor
6.	on	on	off	on	Initialize Memory	
7.	on	on	on	off	Relocate PROM	
8.	off	off	off	on	Run from Memory	
						if-> SW1-5* SW1-4*-> Monitor
9.	off	off	off	off	Boot Finished	

Figure 6-1. Interpretation of LEDs

The setting of SW1-4 and SW1-5 determines how much power up code is run before the system returns control to the ROM diagnostic debugger. The switch settings and the point where control is returned are also presented in Figure 6-1.

The nature of the failure directs the PROM to the appropriate embedded diagnostics, which will be executed in an infinite loop until some corrective action is taken. If all tests pass, the system enters the interactive monitor in normal operation.

6.4 Initialization

When you first power the system up, a short PROM message is displayed on the screen:

Probe and Configure PVSB Memory
A0000000 - 0000 - 0001
Initialize all memory .....
8:00 M
Server Node

The first line tells you that the CPU is self-configuring memory, which means it is looking for memory cards and keeping a cumulative count of the cards it finds and the amount of memory resident on each card.

The next line describes memory card by card. A system with three cards would have three lines:

Probe and configure PVSB memory
A00*N* 0000 0000 000*Y*
A00*N*+*1* 0000 0000 000*Y*
A00*N*+*2* 0000 0000 000*Y*

The slot number is designated *N* in the display. The processor looks for 16 Mbyte cards first, 8 Mbyte cards next, and 4 Mbyte cards last. Since cards can be placed in any slot position, the first 16 Mbyte card found may not correspond to slot one, but it will be the first slot displayed. The total memory found on each card is displayed in *Y*. Memory is counted in 4 Mbyte units so a 16 Mbyte card would display 3 (0-3 units) at *Y*. In the illustration, three slots have been found with memory, so three lines are displayed.

The total amount of memory found in the system is printed followed by an **M** indicating megabytes. The system then declares whether it is a cluster or server.

The initialization message concludes with a line that indicates memory is initialized. A dot is displayed for each 4 Mbyte unit located.

6.5 ISI Banner Display

After memory initialization, the ISI logo appears with a display of VME-68K30 board characteristics. The display provides you with information concerning how your board is configured and the type of PROM set you are running.

Integrated Solutions and NBI Company
Copyright © 1987, 1988
VMEbus 68030 5.0 boot 11/4/88
Board version: FF 25 MHz PVSB
:

The first two lines are self-explanatory.

The third line displayed tells you that you are running on a VEMbus with a VME-68K30 processor board. You have the ISI 5.0 version of UNIX. It was released on the date indicated.

The fourth line displays the revision level of the board in your system. The hex numbers FF represent that level in this mock display. You would use this revision level for field upgrades and for matching PROMs to the board you have. The speed of the CPU is also printed in line four (in this case 25 MHz) as is the memory bus type. It will either be a performance VSB (PVSB), which is proprietary to ISI, or a standard (VSB). A PVSB bus supports only PVSB memory boards. A standard VSB supports boards from third party vendors.

The last line of this display is a promt (:). You can autoboot from the PROM prompt or go to the monitor. To autoboot, simply type in @ (the "at" sign) and <RETURN>. If the system does not boot, it cannot find a boot file and one must be loaded and identified by pathname from a distribution tape.

6.6 Loading A Program from The Distribution Tape

When loading a program from the distribution tape, it is necessary to know the position of the program on the tape.

For a TS11 compatible nine-track tape or cartridge unit, insert the tape in the drive and enter the following:

ts(0,x)

where **x** is the position of the file on the tape.

The message:

Loading at 0xwww: xxx+yyy+zzz
Type <RETURN> to start at 0xwww

will appear. Enter a <RETURN> to begin execution.

Load the full diagnostic suite from tape with the same command. The diagnostics that you are allowed to run are listed at the end of this section. Refer to the *System Administrator Guide (SMM:1)* for more information.

6.7 Monitor Commands at the PROM Prompt

Besides the ability to load and execute programs for peripheral devices, the bootstrap PROMs open and modify memory and I/O locations. The address to be opened or modified is specified in hex.

CAUTION

Since these commands may do system level damage, only qualified administrators should use the monitoring feature of the PROMs. If you are uncertain, obtain directions from Integrated Solutions Customer Support. For instance, writing into the control registers of disk controllers can result in reformatting the disk and losing all data.

If UNIX is not running, establish communication with the PROM by pressing the RESET switch on the front of the system. The PROM prompt will appear if the hardware is operative.

If UNIX is running and users are logged in, bring down the system with the *shutdown*(8) command. This command notifies all logged-in users that the system is coming down, kills all processes, and goes into single-user mode. After issuing the *shutdown*(8) command, the file systems should be unmounted with *umount*(8) and the superblock updated with *sync*(8) and *reboot*(8).

When the PROM prompt appears, type an **h** to display a help menu with options that test simple read, write, and execute instructions in memory. The menu is reproduced as follows:

h	Displays the command list
# <address> [value]	Displays/modifies the long word at <address>, if [value] is present it is modified
$ <address> [value]	Displays/modifies the word at <address>
% <address> [value]	Displays/modifies the byte at <address>
*** <address> <length> [value]**	Display/modify memory starting at <address> for <length> long words, if [values] are present they are modified
= <address> <length> [value]	Display/modify memory starting at <address> for <length> words
+ <address> <length> [value]	Display/modify memory starting at <address> for <length> bytes,
|	PROM diagnostic routines (Goes to the embedded diagnostics user interface)
^	Toggles internal cache on/off
!	Toggles interrupts enable
< dev (unit , paratition) file (args)	Load an image (Provide a path to a device with the desired image.)
> <address>	Jump to location (default 0x8000) Begins execution at the specified address.
?	List directory (Requests system to list all device drivers that could contain the boot image.)
~	Puts CPU in transparent mode (used for *da* driver down load)
~.	Exits transparent mode
.	Repeats the last command line
&	Goes to Non-volatile RAM (NVRAM) utility
@	Autoboots the system
:	To boot the system, the user must enter the location of the boot file: dev(unit, paratition)file(args)

Selected monitor commands are discussed in more detail in the following:

% address [value]	Opens/modifies location as a byte where *address* is the hexadecimal address for that memory location and *[value]* is the optional new hexadecimal value to write at address.
$ *address [value]*	Opens/modifies location as a word where *address* is the hexadecimal address for that memory location and *[value]*

is the optional new hexadecimal value to write at address.

 # address [value] Opens/modifies location as a long word where *address* is the hexadecimal address for that memory location and *[value]* is the optional new hexadecimal value to write at address.

These commands can be used in two ways:

by entering the address in hex

$fff520

or by entering the address in hex with a new value for that address

$fff520 1224

If you specify the value, the location is written to without ever being read.

When you type % and the hexadecimal address of a memory location, the system appends a vertical bar (|) followed by the current byte value at that address and waits for you to enter the new hexadecimal data for the location.

When you type $ and the hexadecimal address of a memory location, the system appends a colon (:) followed by the current word value at that address and waits for you to enter the new hexadecimal data for the location.

When you type # and the hexadecimal address of a memory location, the system appends a semi colon (;) followed by the current long word value at that address and waits for you to enter the new hexadecimal data for the location.

For example, entering

200

might display

200:68AC

where " " represents the cursor location.

If you do not type in a new value and you press <RETURN>, the location remains unmodified.

If you type in a new hex value following the displayed hex value

200:68AC 68AF

and press <RETURN>, the new value is written to the current location, and the colon (:) prompt appears.

If a location is closed with a <LINE FEED> rather than a <RETURN>, the next location is opened rather than returning to the monitor prompt (:).

If a location is closed with back slash (\), rather than a <RETURN>, the previous location is opened rather than returning to the monitor prompt (:).

NOTE

Opening a non existent memory location with interrupts enabled will cause the system to continuously attempt to access the non existent location, causing trap errors.

+ address length [value] — Essentially the same as the % command, except a block with a starting address and specified length is tested.

= address length [value] — Essentially the same as the $ command, except a block with a starting address and specified length is tested.

** address length [value]* — Essentially the same as the # command, except a block with a starting address and specified length is tested.

> address — Begins execution at a hex address.

If you enter the greater than sign (>) and an address in hex, you will immediately begin execution at that address. For example, entering

>68AC

will jump to 68AC.

! — Toggles interrupt enabling. Entering *!* enables interrupts; entering *!* again disables them. All ISI toggles in the monitor work in this way.

? or h — Lists a directory or help menu of commands available from the monitor. Entering an "h" does the same.

(dev , dev) — Lists the root directory of whatever major and minor device are identified in decimal in the parenthesis.

/ — Enter PROM diagnostics. A menu is displayed for diagnostics with the following options:

1. cache test
2. FPU test
3. NVRAM test
4. software interrupt test
5. timer test
6. VDMA test
7. ?

The last item on the menu is a command to print the menu again.

dev **(M, m)** *pathname*

Loads and starts execution of a file on a device, where:

dev
is the device name (e.g., **sd, sm, ts, da, vb, ex, nw**)

M
is the device major device number (e.g., **0** for the first such device,**1** for the second)

m
is the device "minor" number, usually

- for disks, the partition number, or 100b (block 100 on the device)
- for tapes, the file number on the tape
- otherwise, 0

pathname

is the path in a 4.3 system to the file you want to execute on disk.

For example, you could issue these commands:

ts(0,8)
sd(0,6)stand/bad144

to execute the eighth file on a tape or to execute the file *stand/bad144* on the *g* (sixth) partition of an **sd** disk. If you want to go directly to a block instead of a partition, you can specify some number followed by "b". For example:

sd(0,10b)

In some applications, it may be desirable to load a program without automatically beginning its execution. Preceding the load string with "<" causes the file to be loaded and a starting address to be printed. Then control is returned to the PROM monitor. The program can then be started by typing a ">" followed by the specified starting address.

~

Puts the CPU into transparent mode. Characters from the console are transmitted to the second serial port and characters from the second serial port are directed to the console port.

~.

Exits transparent mode to the monitor prompt.

&

A short menu of options that are set in non-volatile RAM is displayed. These options include setting and displaying dates and times. The RAM is written by selecting one of the options.

6.8 PROM Diagnostics

If you enter **&** while in the monitor menu, a diagnostic menu will appear on screen. The menu indicates that you are in the diagnostic mode and any of the displayed letters, if entered on the command line, will cause the diagnostic to be run.

Some global commands are embedded in the diagnostic that will allow you to move around in memory and exit the diagnostic program. These are:

1. Typing an "S" on the command line will give you the value of the stack pointer.
2. Typing an "x" puts you in a scratch pad if you are at the beginning of a command line, but if entered at the end of a command line or prompt line it loads the scratch pad with the value of the command line.
3. A <LINE FEED> will go to the next address in memory.
4. Typing a "\" moves you to the previous line in memory.
5. Entering <RETURN> gets you back to the main menu from where ever you are.

The debugger menu has the following options:

a	set source/destination function codes *This shows the current function code and allows you to set a new value.*
b	set byte/word/long
c	repeatedly r/w at fixed location
d	look at memory locations interactively *You will be prompted for an address and allowed to read it and write into it.*
e	simple memory test
f	full memory test
g	repeatedly read/write/verify at a fixed location
h	run read/modify/write cycles
i	Alternate space I/O word transfer *This command writes to ISI defined space as described in Section 4.*
k	memory fill
m	memory move
p	set PC *Sets the program counter.*
g	toggle I-cache *Toggles the internal instruction cache on and off*
s	set SR *Allows you to set the status register.*

u	*set CACR* *The cache access control register is set with this command.*
x	eXit cold boot *You leave the monitor and the system is cold booted*
l	memory dump
n	memory compare *Writes data to specified memory location and compares it on a read*
r	toggle D-cache *Toggles the internal data cache on and off*
t	tas test *Tests and sets a semaphore and echos the result*
v	set VBR

6.9 Tape Diagnostics

ISI's diagnostics suite is delivered only on tape with provisions made for disk installation if so desired. The diagnostics support the concept of "USER" and "EXPERT" level diagnostics and hide the expert (in some cases destructive) diagnostics from the average user. The diagnostics capabilities (test/utility) supported by the VME-68K30 stand alone diagnostics are:

6.10 Brief Description of Available Diagnostics

BAD144: Bad sector mapping

BOOT: Same as boot EPROMs

CACHE: Cache test

CAT: Copy device contents to screen

CHKTXT: Compare memory version of text with a file

CLUSTER: tests cluster processor interface

CMP: Compare two devices

COPY: Copy a file from one device (FROM:) to (TO:) another

CPU: CPU performance test

DCOPY: Copy one disk partition to another

DISKFORMAT: Disk (sm, sd, el, gd) formatting

DISKREAD: Disk read test

DISKTEST: Write then read specific patterns on disk

DISKWRITE: Disk write test (destructive)

DMA: Simple dma test (destructive), requires working disk drive

DMAX: More flexible dma test, requires working disk drive

DPERFORM: Disk performance test

FLOATING POINT: tests floating point coprocessor interface

ICP: ISI ICP8/16 test

INTERRUPT: Test interrupt processing

MEM: Memory test (patterns, walking, ping-pong, execution, uniqueness)

MEMTEST: Memory test (byte oriented, less user interaction)

OCTART: VME-68K30 on-board OCTART serial I/O test

OD: Convert device contents to readable format and copy to screen

STRESS: a system-level test involving mmu/memory/dma/cache

VME: VDMA/VATM test

6.11 ISI Stand Alone Diagnostics Executive (sadie)

To run diagnostics at tape, boot your system and insert an ISI supplied SADE tape in your drive. Several banners will appear as described in the subsection "Initialization' in this section.

At the boot prompt (:) type:

ts@

The system will display the menu that is appropriate for your CPU and PROM version. There are two levels of tape diagnostic available: User and Expert. The difference between them is that Expert level diagnostics contain destructive tests, and should only be run by system administrators or others responsible for system operation.

As soon as the tape is loaded, the user level menu is displayed with the message:

"Stand Alone Diagnostics Executive"

6.11.1 User Level sadie Menu

a) copy b) bad144 c) diskformat d) mmu

e) mem f)dma g) gdbad h) dmax

Enter letter of desired test [a h]:

6.11.2 Expert Level sadie Menu

CAUTION

If you are a system administrator or otherwise fully responsible for system operation, use the expert level diagnostics with the warning that some may be destructive. If you are unsure about the effects of any of the following tests, consult the system administrator before proceding.

To load the expert level SADE commands, enter "@ instead of any of the letter options at the user level prompt. The following menu will be displayed:

) copy	**b) bad144**	**c) diskformat**	**d) mmu**
e) mem	**f) dma**	**g) gdbad**	**h) install**
i) memtest	**j) boot**	**k) dmax**	**l) cat**
m) od	**n) dperform**	**o) diskread**	**p) diskwrite**
q) disktest	**r) dcopy**	**s)cpu**	**t) cmp**
u) clock	**v) icp**	**w) interupt**	**x) gip**
y)mathtest	**z) fault**	**{)prmmu**	**|)chktst**

Enter letter of desired test [a |]:

Integrated Solutions

DOCUMENTATION COMMENTS

Please take a minute to comment on the accuracy and completeness of this manual. Your assistance will help us to better identify and respond to specific documentation issues. If necessary, you may attach an additional page with comments. Thank you in advance for your cooperation.

Manual Title: *VME-68K30 Hardware Reference Manual*	Part Number: *490247 Rev. A*

Name: _____ Title: _____

Company: _____ Phone: ( ) _____

Address: _____

City: _____ State: _____ Zip Code: _____

1. Please rate this manual for the following:

	Poor	Fair	Good	Excellent
Clarity	☐	☐	☐	☐
Completeness	☐	☐	☐	☐
Organization	☐	☐	☐	☐
Technical Content/Accuracy	☐	☐	☐	☐
Readability	☐	☐	☐	☐

Please comment: _____

_____

2. Does this manual contain enough examples and figures?

Yes ☐ No ☐

Please comment: _____

_____

3. Is any information missing from this manual?

Yes ☐ No ☐

Please comment: _____

_____

4. Is this manual adequate for your purposes?

Yes ☐ No ☐

Please comment on how this manual can be improved: _____

_____

_____

_____

_____

_____

_____

Fold Down First Fold Down

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

First-Class Mail Permit No. 7628 San Jose, California 95131

Postage will be paid by addressee

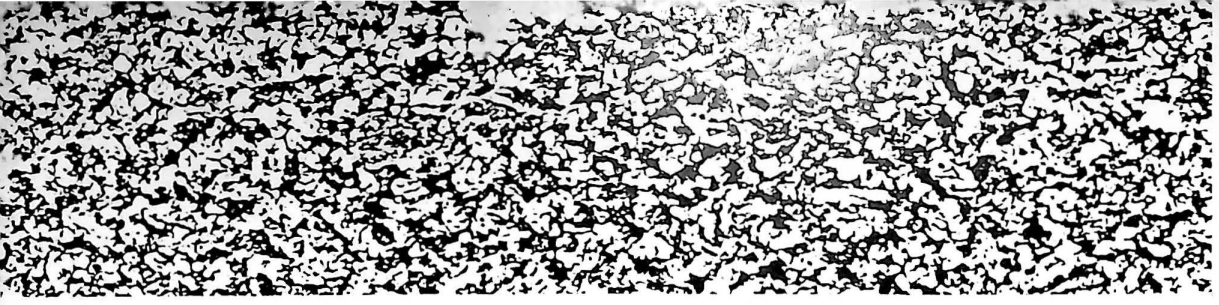
Integrated Solutions

An NBI Company

ATTN: Technical Publications Manager
1140 Ringwood Court
San Jose, CA 95131

Fold Up Second Fold Up

Staple Here



Integrated Solutions
1140 Ringwood Court
San Jose, CA 95131
408 943-1902
Telex 499 6929