VME-4SIP

6U Quad Industry Pack Carrier

REFERENCE MANUAL

Revision 1.3
February 2005

ALPHI TECHNOLOGY CORPORATION
6202 S. Maple Avenue #120
Tempe, AZ 85283 USA
Tel : (480) 838-2428
Fax: (480) 838-4477

NOTICE

The information in this document has been carefully checked and is believed to be entirely reliable. While all reasonable efforts to ensure accuracy have been taken in the preparation of this manual, ALPHI TECHNOLOGY assumes no responsibility resulting from omissions or errors in this manual, or from the use of information contain herein.

ALPHI TECHNOLOGY reserves the right to make any changes, without notice, to this or any of ALPHI TECHNOLOGY's products to improve reliability, performance, function or design.

ALPHI TECHNOLOGY does not assume any liability arising out of the application or use of any product or circuit described herein; nor does ALPHI TECHNOLOGY convey any license under its patent rights or the rights of others.

1. Introduction

The VME-4SIP from ALPHI Technology is a Quad Industry Pack carrier for the VME Bus and will hold one to four single or two double INDUSTRY PACK modules. Each INDUSTRY PACK may be operated at 8 or 32 MHz clock frequencies by writing to a register. The VME-4SIP supports I/O, memory, and interrupt functions. The input-output spaces and ID spaces for the INDUSTRY PACKs are accessed in A16:D16 mode from the VME Bus, selected by jumpers at offsets which are multiples of hex $0800. Memory access may be defined in A24 space or in A32 space. The address access mode and size are software controlled (by writing to a specific register). In the A32 mode, each INDUSTRY PACK is allocated 8 Mbytes of memory while in the A24 mode the allocated memory space of the VME Bus may be set to a minimum of 128 Kbytes per INDUSTRY PACK to a maximum of 2 Mbytes per INDUSTRY PACK. This is achieved by setting a register to select the desired memory mode of operation. The default value of the register is set to disable memory. Consequently, it is very important that this register be initialized before a memory access. A simple, yet very powerful interrupt routing scheme provides up to 3 interrupt lines on the VME Bus to be driven. Each group can accept a total of 8 interrupt lines (either of the two interrupt lines from each INDUSTRY PACK). A rotating priority scheme (round robin) is used within each group for the 4 possible inputs (A/B/C/D). The output from each group is set to drive any of the 7 VME Bus interrupt lines by means of a software-set register. This is cleared by reset, so disabling interrupts. An interrupt acknowledge vector (STATUS/ID) register is included, and may be set by software to return a vector in cases where an INDUSTRY PACK does not have the capability to do so.

2. Summary of features

- 6U VME Bus Industry Pack carrier board
- Accepts 4 INDUSTRY PACK modules or 2 double wide modules.
- Complies with the VITA 4 INDUSTRY PACK Specification
- Conforms to VME Bus specification C.1
- Each INDUSTRY PACK can be operated at 8MHz or 32 MHz
- I/O and ID access mode A16/D16
- A16 base address defined by 5 jumpers
- Memory access in A24/D16 or A32/D16 modes
- A24 or A32 memory space access base address
- All 22 address bits can be controlled in A24 access modes
- Read-modify-write cycles supported
- INDUSTRY PACK interrupts directed to any 3 of the 7 VME Bus interrupt lines
- I/O connectors on front panel, Rear panel I/O is optional
  - LED indicators on front panel show INDUSTRY PACK access

3. Block diagram

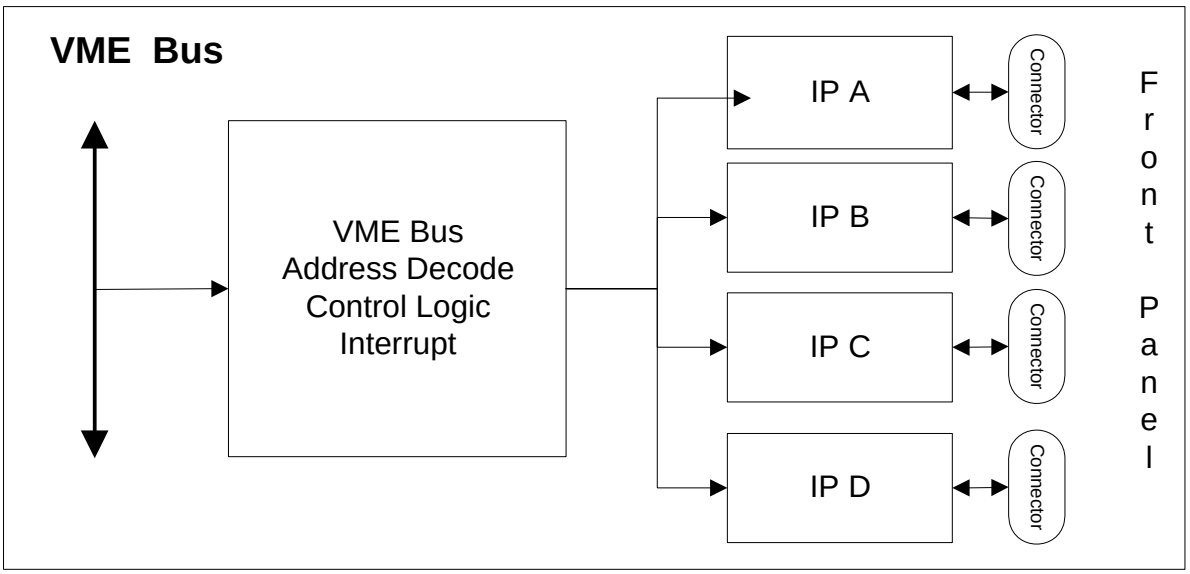

FIG 1: Block Diagram of the VME-4SIP

4. Description

4.1 Input-Output and ID Addressing

I/O space is accessed using A16:D16 mode. The base address of the VME-4SIP in the A16:D16 mode is selected by 5 jumpers. These offsets are at multiples of $0800 - hexadecimal. The default setting of the jumpers is to a base address of $6000. The selection of each INDUSTRY PACK, lettered A to D, is done by address lines A09 and A08. Thus the I/O space for INDUSTRY PACK A starts at $6000, that for INDUSTRY PACK B at $6100 etc. The ID space is accessed by setting A07 to 1, so that the ID space for INDUSTRY PACK 0 starts at $6080. Table 1 shows how these spaces are arranged.

IP module	Space	Offset
A	I/O	Base + $0000
A	ID	Base + $0080
B	I/O	Base + $0100
B	ID	Base + $0180
C	I/O	Base + $0200
C	ID	Base + $0280
D	I/O	Base + $0300
D	ID	Base + $0380
Other	Register	Base + $0400

Table 1: I/O Space Mapping

A few configuration registers are located at higher offsets: with A10 set to 1. The module occupies 16 bit address space extending to offsets of $047F above the base address. Other modules in the VME Bus can be allowed to use addresses such as $6500 if need be. The base address settings are achieved by setting jumpers 1 to 5 to correspond to address lines A11 to A15 respectively. A jumper bridging a pin to the ground row will define a logical 0 for its address bit. Thus, to take the example of a base address of $6000, address bits A11, A12 and A15 must be 0 and A13 and A14 at 1, so jumpers 1, 2 and 5 must be inserted to link to pins 2, 4 and 10. Table 2 lists the hexadecimal values set by each jumper NOT inserted.

Jumper pin	Address line not linked	Offset
9-10	A15	$8000
7-8	A14	$4000
5-6	A13	$2000
3-4	A12	$1000
1-2	A11	$0800

Table 2: A16 Base Address

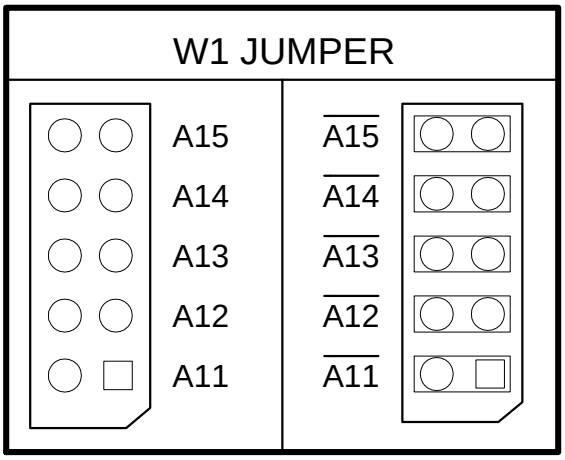

Fig 2: Diagram of the W1 Jumper

Jumpers W6,P3 and P4 are for factory use only and should not be changed.
Jumpers W2-W5 are used to synchronize multiple IP's if they are using IPstrobe (pin 47 on the IP) as trigger-in or trigger-out. For example, if IP A is used as a source to synchronize the other IP's then pins 1 & 2 need to be connected by a jumper on W2 while pins 2 & 3 need to be connected by a jumper on W3, W4, and W5.
Note that in many cases the INDUSTRY PACK registers will be of 8 bits, using data lines D0..D7. Access to the registers is done in 16 or 8 bits. The upper 8 bits are ignored.

4.2 Configuration Registers

The following table shows the address offsets of the configuration registers from the base address of the modules, in A16 space. They are all 8-bit registers, at even offsets in the usual environment. All other addresses are reserved.

Register	Function	R/W	Offset (hex)
IVR	Interrupt vector Reg.	R/W	$0400
MAM	Memory access mode	R/W	$0402
HAR	High address	R/W	$0404
FIPC	Clock frequency set	R/W	$0406
IVRE	IVR Enable	R/W	$0408
ISR	Interrupt Status	R/W	$040A
MAR	Memory base address	R/W	$040D
GP1	Group 1 mask	R/W	$0414
GD1	Group 1 drive	R/W	$0416
GP2	Group 2 mask	R/W	$0424
GD2	Group 2 drive	R/W	$0426
GP3	Group 3 mask	R/W	$0434
GD3	Group 3 drive	R/W	$0436

Table 3: Configuration Registers

4.3 Memory Space Access

IP memory space can be accessed using A32 or A24 modes. Selecting the appropriate mode is done by writing to MAM (memory address mode) register. On reset, this register is reset to zero, disabling memory access. Consequently, before proceeding it is very important to initialize MAM. Mode 7 is used in A32 memory accesses, and mode 1-5 is used in A24 memory accesses.

4.3.1 A32 Memory Space Access

To enable A32 mode (A31-A1)[1] memory access, the first step is to write $07 to the register MAM which is found on address $0402. This permits the operation in mode 7. The most significant byte is defined by the MAR (MAR[7..1] correspond to lines A31 to A25 respectively). For example using MVME 162 requires the loading of MAR with $F1 (the most significant byte) because A32:D16 on the MVME 162 is from $F1000000 to F1FFFFFF. Lines A23 and A24 are used to select the IP (A/B/C/D). This leaves 23 lines (A22 and below) for the IP or 8 MB of memory space for each IP.

Offsets for the four modules are:

IP module	MEM space Offset
IP A	Base address + $ 0000 0000
IP B	Base address + $ 0080 0000
IP C	Base address + $ 0100 0000
IP D	Base address + $ 0180 0000

Table 4: IP module offsets

4.3.2 A24 Memory Space Access

Memory space available is 16 Mbytes total.
Allocation of space size for the VME_4SIP is made using MAM register.
MAM = 5 ($402) allocated 8 Mbytes total for the board or 2 Mbytes by IP space.
MAM = 4 ($402) allocated 4 Mbytes total for the board or 1 Mbytes by IP space.
MAM = 3 ($402) allocated 2 Mbytes total for the board or 512 Kbytes by IP space.
MAM = 2 ($402) allocated 1 Mbytes total for the board or 256 Kbytes by IP space.
MAM = 1 ($402) allocated 512 Kbytes total for the board or 128Kbytes by IP space.

The MAR register ($40C) is used to allocate the memory size requested by MAM within the 16 Mbytes (A24). For example, with a A24 base address of $F000 0000 and a request of 8Mbytes of memory for the VME_4SIP (MAM = 5), there are only two possibilities. MAR[7] = 0 : base address => $F000 0000

IPA: $F000 0000-$F01F FFFF
IPB: $F020 0000-$F03F FFFF
IPC: $F040 0000-$F05F FFFF
IPD: $F060 0000-$F07F FFFF

MAR[7] = 1 : base address => $F800 0000

IPA: $F080 0000-$F09F FFFF
IPB: $F0A0 0000-$F0BF FFFF
IPC: $F0C0 0000-$F0DF FFFF
IPD: $F0E0 0000-$F0FF FFFF

For example, with a A24 base address of $F000 0000 and a request of 512Kbytes of memory for the VME_4SIP (MAM = 1), there are 32 possibilities.

MAR[7..3] = 0 : base address => $F000 0000

IPA: $F000 0000-$F001 FFFF
IPB: $F002 0000-$F003 FFFF
IPC: $F004 0000-$F005 FFFF
IPD: $F006 0000-$F007 FFFF

MAR[7..3] = 1 : base address => $F008 0000

IPA: $F008 0000-$F009 FFFF
IPB: $F00A 0000-$F00B FFFF
IPC: $F00C 0000-$F00D FFFF
IPD: $F00E 0000-$F00F FFFF

The IPx memory space can still have up to 8 Mbytes of memory but only a portion will be seen by the VME bus. For example in the mode 1 MAM[ ] =1 that allocated only 128Kbytes by IP, (512Kbytes for the VME_4SIP) if the module on IP_A has 1 Mbytes, HAR[ ] ($404) will be used to select a bank of 128 Kbytes within the 1 Mbytes located on IPA.

Access of the full 1 Mbytes of memory located on IPA will be made in two step:

- Set the HAR[6..1] to select the block of 128k within the 1M byte.
- Access the IPA memory *always* on $F000 0000 to $F001 FFFF.

Note: HAR[6..1] register has only bit # 1 to bit # 6 active not bit# 0 to bit # 7.

When addressing the IP, there is no A0. Instead there are Byteselect0 (BS0) and Byteselect1 (BS1)

MODE 5 : Write $05 at $0402 (MAM).

Lines A20 and below are passed to the IP (2^{21}=2MB). A22 and A21 are used to select the IP. HAR[6] and HAR[5] are used to select a 2MB block within the IP. MAR[7] is used to decode a 8MB space inside the VME space (16MB). It must be noted that although the user can address up to full 8MB per IP, the VME BUS sees ONLY 2MB per IP.

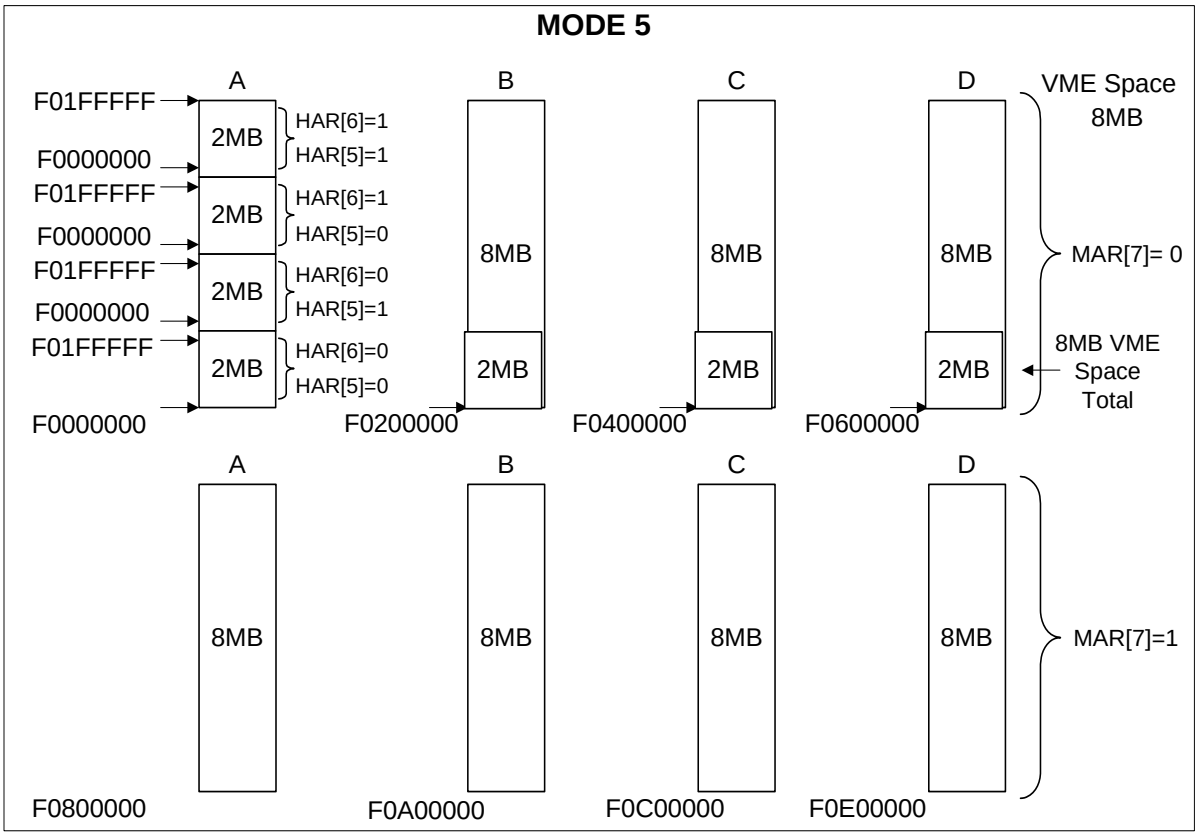

Fig 3: Mode 5 A24 addressing

MODE 4 : write $04 at $0402 (MAM)

Lines A19 and below are passed to the IP (2^{20}=1MB). A21 and A20 are used to select the IP. HAR[6..4] are used to select a 1MB block within the IP. MAR[7] and MAR[6] is used to decode a 4MB space inside the VME space (8MB). It must be noted that although the user can address up to full 8MB per IP, the VME BUS sees ONLY 1MB per IP.

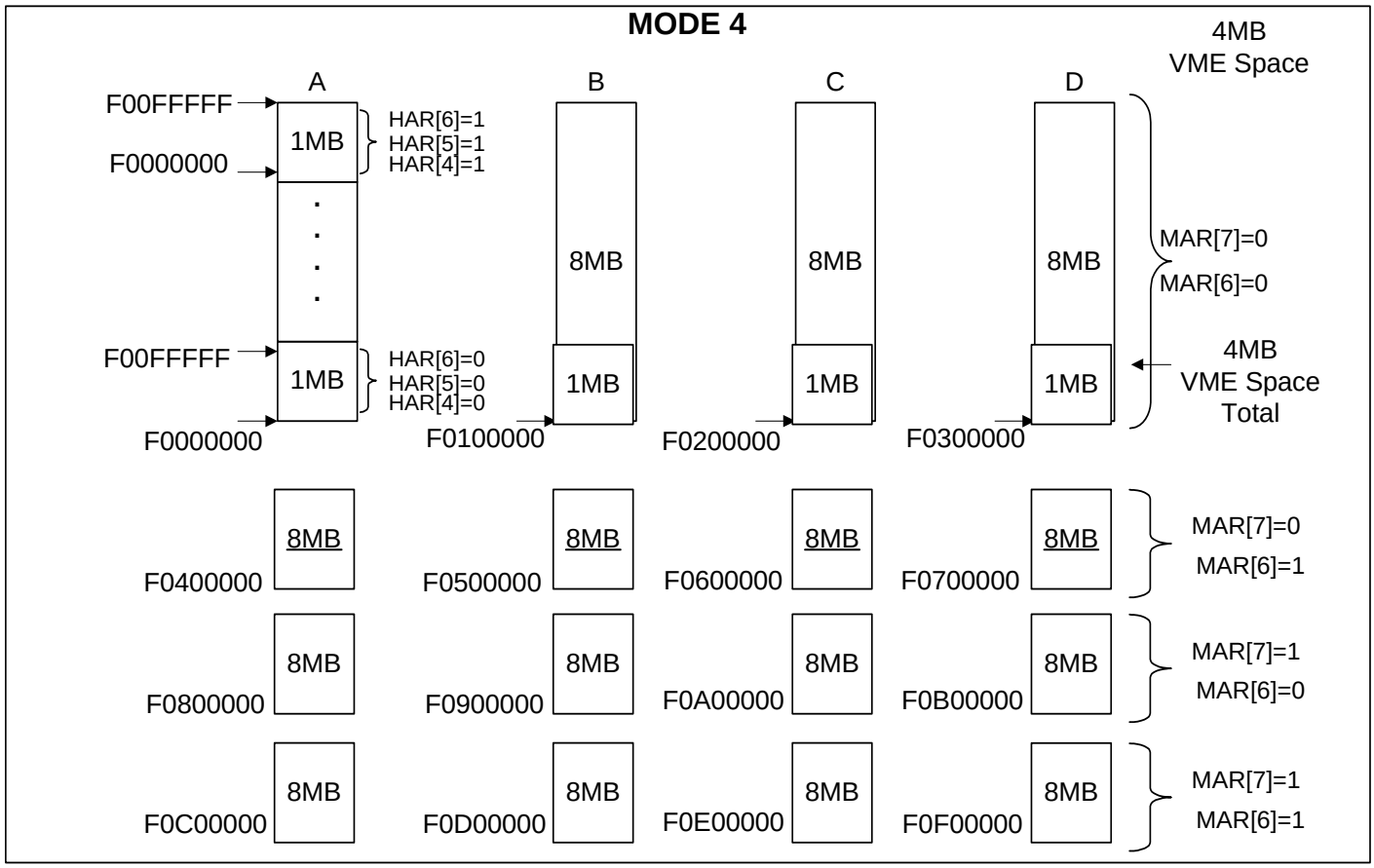

Fig 4: Mode 4 A24 addressing

MODE 3 : write $03 at $0402 (MAM)

Lines A18 and below are passed to the IP (2^{19}=512KB). A20 and A19 are used to select the IP. HAR[6..3] are used to select a 512KB block within the IP. MAR[7..5] are used to decode a 2MB space inside the VME space (4MB). It must be noted that although the user can address up to full 8MB per IP, the VME BUS sees ONLY 512KB per IP.

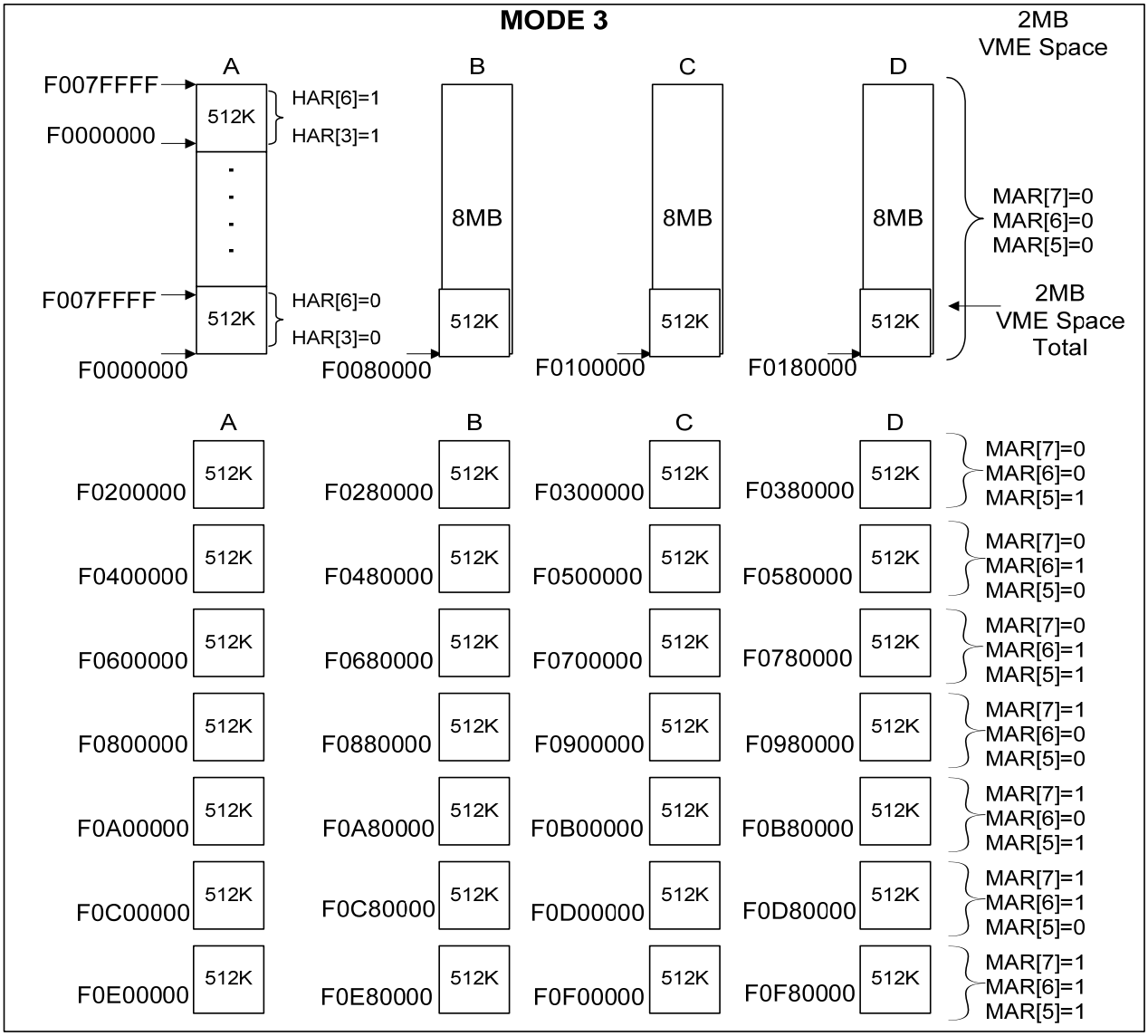

Fig 5: Mode 3 A24 addressing

MODE 2 : write $02 at $0402 (MAM)

Lines A17 and below are passed to the IP (2^{18}=256KB). A19 and A18 are used to select the IP. HAR[6..2] are used to select a 256KB block within the IP. MAR[7..4] are used to decode a 1MB space inside the VME space (2MB). It must be noted that although the user can address up to full 8MB per IP, the VME BUS sees ONLY 256KB per IP.

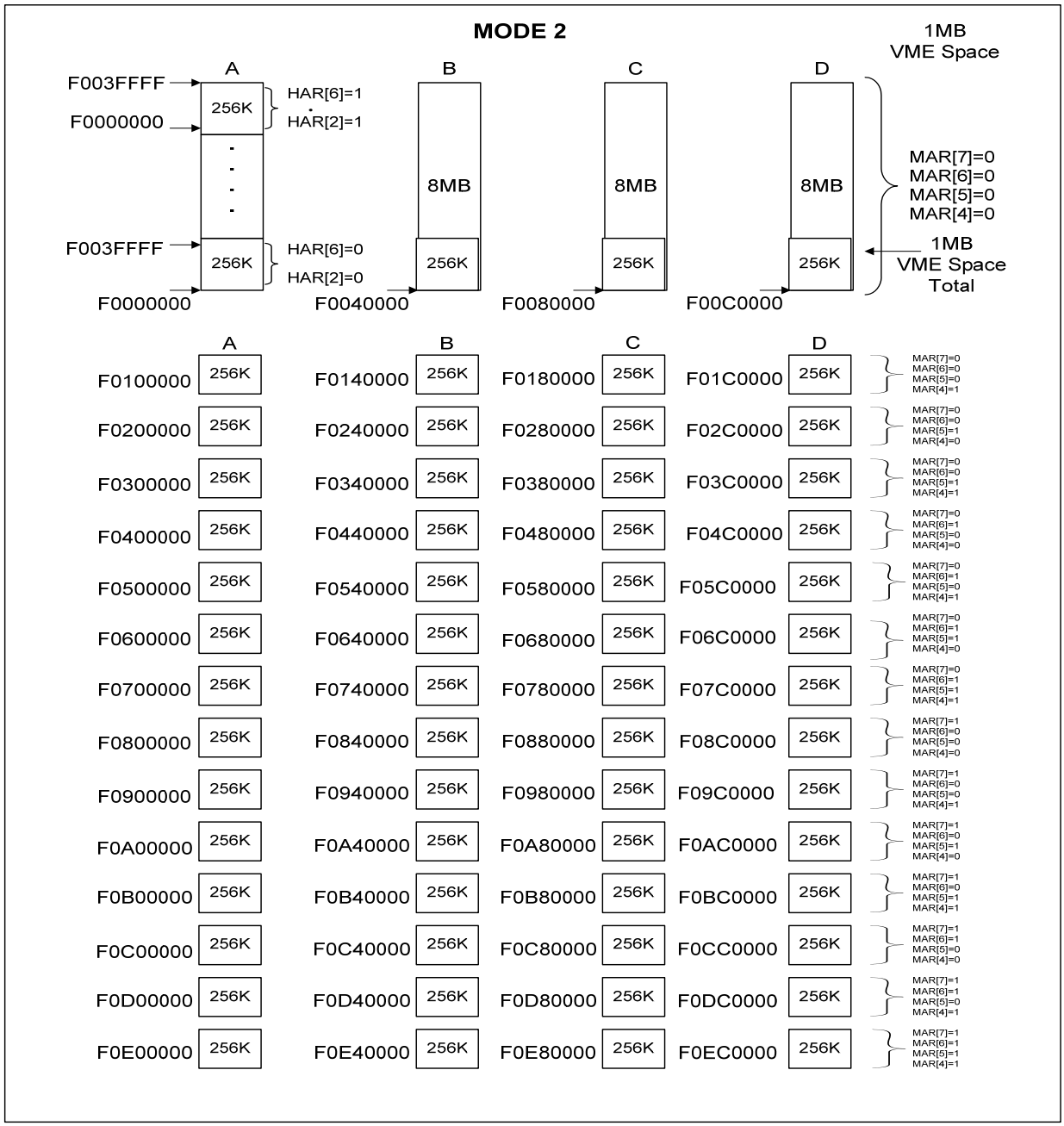

Fig 6: Mode 2 A24 addressing

MODE 1 : write $01 at $0402 (MAM)
Lines A16 and below are passed to the IP (2^{17}=128KB). A18 and A17 are used to select the IP. HAR[6..1] are used to select a 128KB block within the IP. MAR[7..3] are used to decode a 512KB space inside the VME space (2MB). It must be noted that although the user can address up to full 8MB per IP, the VME BUS sees ONLY 128KB per IP.

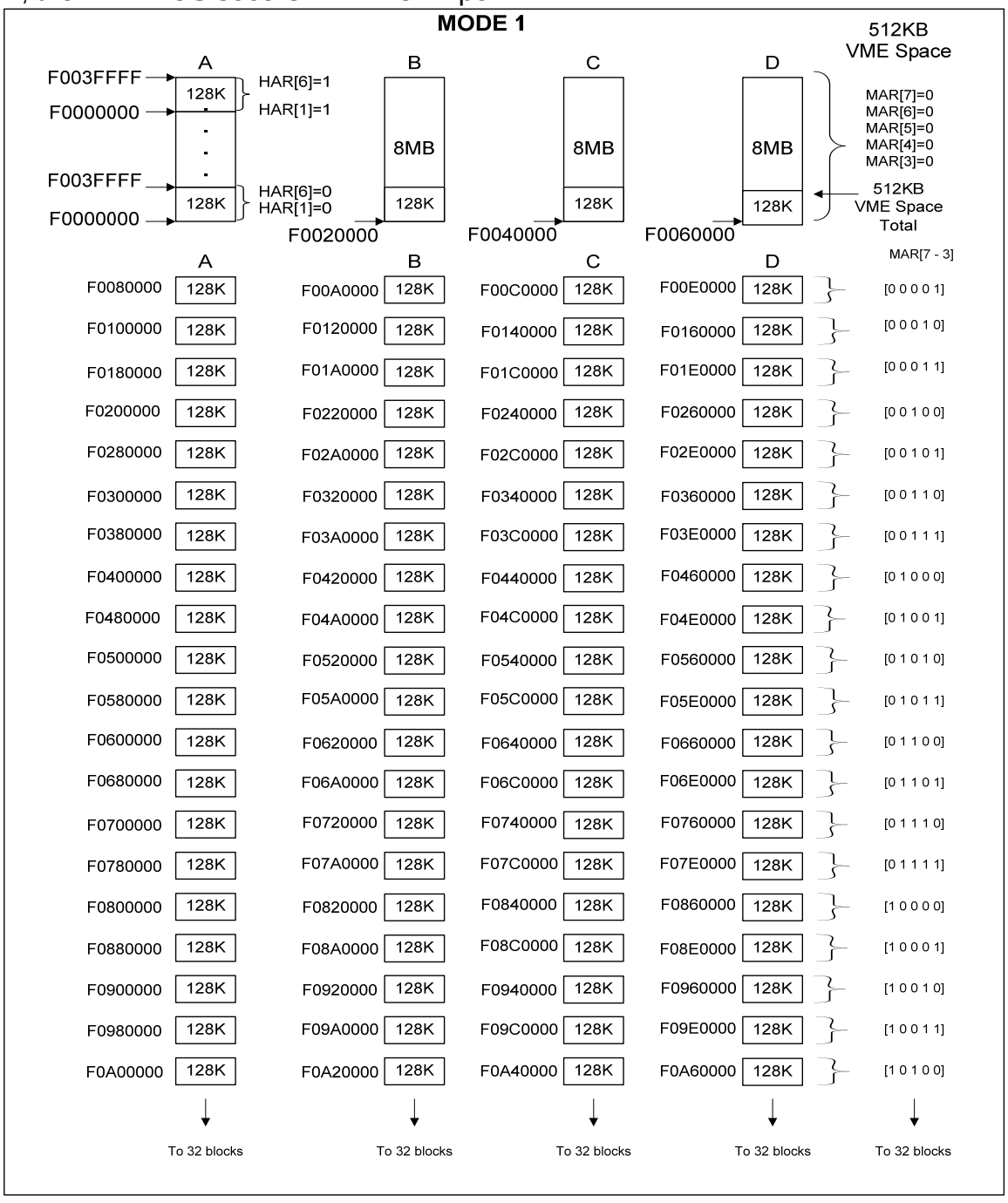

Fig 7: Mode 1 A24 addressing

4.4 Interrupts

The VME4SIP has 4 INDUSTRY PACK sites. Each IP slot contains 2 interrupts commonly known as Intreq0 and IntReq1. The interrupt handling scheme of the VME-4SIP is split up into 3 interrupt mask registers. Each group is able to drive any of the 7 interrupt request lines on the VME Bus. Each group will be labeled as follows **GP1**, **GP2** and **GP3**. The table below shows the address and function of the 8 bits of each group.

$414 - GP1 Bits	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Interrupt line	1	1	1	1	0	0	0	0
IP Location	D	C	B	A	D	C	B	A

$424 – GP2 Bits	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Interrupt line	1	1	1	1	0	0	0	0
IP Location	D	C	B	A	D	C	B	A

$434 – GP3 Bits	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Interrupt line	1	1	1	1	0	0	0	0
IP Location	D	C	B	A	D	C	B	A

NOTE: Interrupts generated from the same IP are not differentiated by the group. It is up to the user to make the distinction between the 2 interrupts.

4.4.1 VME Enable Interrupt Levels

There are three, 3-bit registers which correspond with the 3 Interrupt Mask groups. Each register defines which VME Bus interrupt request line will be enabled. The 3 registers **GD1**, **GD2** and **GD3** are all cleared upon reset. The following chart will show each value.

NOTE: that each of these registers must be set to a different IRQ if non-zero. The tables below show the address and function of each register.

$416 GD1 Bits	Data	IRQ
001	1	IRQ1
010	2	IRQ2
011	3	IRQ3
100	4	IRQ4
101	5	IRQ5
110	6	IRQ6
111	7	IRQ7

$426 GD2 Bits	Data	IRQ
001	1	IRQ1
010	2	IRQ2
011	3	IRQ3
100	4	IRQ4
101	5	IRQ5
110	6	IRQ6
111	7	IRQ7

$436 GD3 Bits	Data	IRQ
001	1	IRQ1
010	2	IRQ2
011	3	IRQ3
100	4	IRQ4
101	5	IRQ5
110	6	IRQ6
111	7	IRQ7

4.4.2 Registers

For INDUSTRIAL PACK modules that don't provide an Interrupt Vector or Status/ID code the VME4SIP has provided an interrupt vector register for this purpose. See explanation below for further detail.

4.4.2.1 Interrupt Vector Register

The IVR accessed at address $400 has 8 bits, but only the most significant 6 bits are read - write. The two lower bits, when returned in the interrupt acknowledge cycle are used to identify the module. The VME4SIP translates the IVR - A thru D based on how the IVR Enable is set. Each IP has its own IVR so if IVR Enable is set for IP_B any writes or reads of the IVR will be for IP_B. Take notice that the IVR Enable must be separately set first before there can be access to that IP's interrupt vector register. See tables below for detail.

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
IVR A	IVR7	IVR6	IVR5	IVR4	IVR3	IVR2	IP[1]	IP[0]

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
IVR B	IVR7	IVR6	IVR5	IVR4	IVR3	IVR2	IP[1]	IP[0]

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
IVR C	IVR7	IVR6	IVR5	IVR4	IVR3	IVR2	IP[1]	IP[0]

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
IVR D	IVR7	IVR6	IVR5	IVR4	IVR3	IVR2	IP[1]	IP[0]

4.4.2.2 Interrupt Vector Register Enable

This register controls which Interrupt vector register A, B, C, or D is going to be accessed. Before a read or write of the IVR can be made there must be a write to address $408. Only after an IVR is selected can there be a write or read to that IVR. If the next write or read is to a different IP- (IVR) you must first write to $408 the location of the IVR needed, then proceed with a write or read to that IVR. See table below for correct settings.

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
IVRE	-	-	-	-	IVRE-D	IVRE-C	IVRE-B	IVRE-A

Note that the IVRE bits shown will enable both interrupt lines from each INDUSTRIAL PACK.

4.4.3 Interrupt status Register

This 8 bit register is located at address $40A. A read of this register will show if an interrupt line (Intreq0 or Intreq1) from IP_A, B, C or D is active (If it is low). When all interrupts are disabled the user should read $FF. Example, if IntReq-A1 were enabled (Low) a read at $40A would read $FD. See table below for details.

Data Bus	D7	D6	D5	D4	D3	D2	D1	D0
Data	7F	BF	DF	EF	F7	FB	FD	FE
IntReq	D1	D0	C1	C0	B1	B0	A1	A0

4.4.4 Interrupt Vector Register Program Example

The following example will show in detail how to program the VME4SIP Interrupt Vector Register. All 3 three groups will be used along with explanation of each step. Conform address to short I/O of VME master being used.

1. Group 1 (GP1).
   - Setup IVR for IntReq-A0 and VME bus IRQ1.
     - $414 write $01 , This enables IntReq-A0 for GP1.
     - $416 write $01 , This enables VME IRQ1 active low for GP1.
     - $408 write $01 , The IVRE is set for writing to IVR-A.
     - $400 write $AA , Write data to IVR-A, when read back of data you will notice $A8 this is because bits 1 & 2 of the IVR are for ID.
     - $408 write $00 , To disable any write to IVR-A, IVR-B, IVR-C or IVR-D and to allow access to the other IVR registers.
2. Group 2 (GP2).
   - Setup IVR for Intreq-B0 and Intreq-C0 and VME bus IRQ3.
     - $424 write $06 , This enables IntReq-B0 and IntReq-C0 for GP2.
     - $426 write $03 , This enables VME IRQ3 active low for GP2.
     - $408 write $02 , The IVRE is set for writing to IVR-B.
     - $400 write $55 , Write data to IVR-B, when read back of data you will $55.
     - $408 write $04 , The IVRE is set for writing to IVR-C.
     - $400 write $41 , Write data to IVR-C, when read back of data you will notice $42 this is because bits 1 & 2 of the IVR are for ID and for IP_C is $02.
     - $408 write $00 , To disable any write to IVR-A, IVR-B, IVR-C, IVR-D and to allow access to the other registers.

     .

After the 2 groups have been programmed the user must re-enable the interrupt register by a write to the IVRE register to enable IVR-A, IVR-B, IVR-C. Based on what IP locations were used, in this case all 3, the IVRE needs to be set to $07.

It is important that the user writes to $408 to enable IVR-A, IVR-B, IVR-C, or IVR-D before accessing address $400 and writing the value of the IVR. Failure to do so will not enable the user to write to address $400 correctly.

4.4.5 BLOCK DIAGRAM

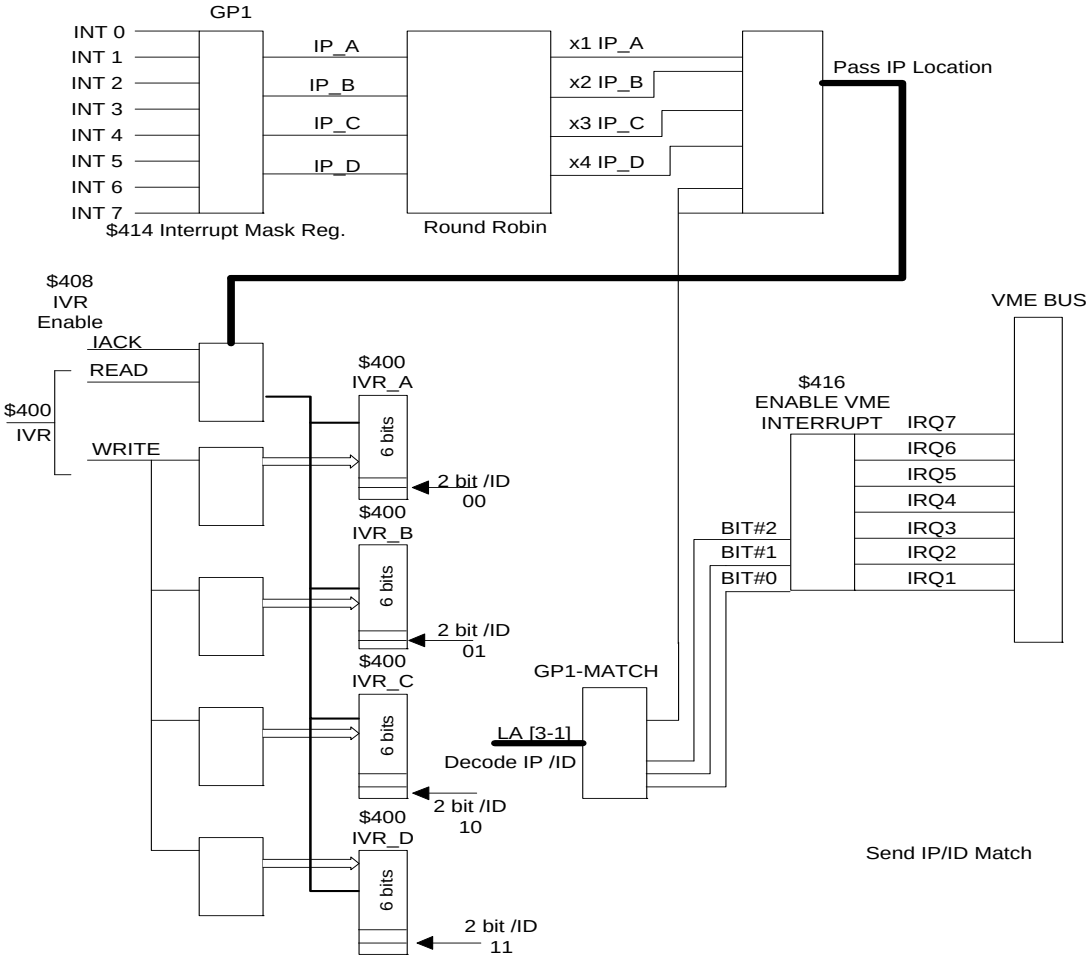

Fig 8: Block diagram of the Interrupt service routine.

4.5 Clock Frequency

The VME4SIP is capable of providing an IP clock of 8Mhz or 32Mhz. Upon reset the VME4SIP resets all 4 INDUSTRY PACK sites at 8Mhz. In the event that 32Mhz is required for an INDUSTRY PACK location the VME4SIP provides a programmable register to do so. At the address $406 it is possible to set IP_A, B, C or D to 32Mhz. The following table below will show the settings necessary to setup each location. Setting D0 will set IP_A to 32Mhz.

Data bus	D7	D6	D5	D4	D3	D2	D1	D0
CLK_IP	-	-	-	-	CLK_D	CLK_C	CLK_B	CLK_A

4.6 Other Features

4.6.1 Fuses

There are 3 fuses for each IP installed which adds to a total of 12 fuses on the board. The 3 fuse are for the +12V, -12V, and +5V lines.

4.6.2 LED Indicators

There are Six LED indicators visible at the front of the board. They are they are labeled on the PCB as L1 to L6 where L1 is at the top of the card.
The LED's have the following meanings:

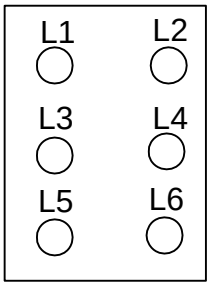

Fig 9: LED layout

LED	LEGEND	DESCRIPTION
L1	IP-D	HOST is accessing IP-D.
L2	IP-C	HOST is accessing IP-C.
L3	VME	VME BUS is being accessed.
L4	BERR	BUS ERROR is generated.
L5	IP-B	HOST is accessing IP-B.
L6	IP-A	HOST is accessing IP-A.

Table 5: LED Legend

5. Jumper Settings

JUMPER	FACTORY SETTING	DESCRIPTION
W1	1-2, 3-4, 9-10	ADDRESS SETUP, DEFAULT $6000
W2	2-3	IP-STROBE for IP-A, DEFAULT DISABLED
W3	2-3	IP-STROBE for IP-B, DEFAULT DISABLED
W4	2-3	IP-STROBE for IP-C, DEFAULT DISABLED
W5	2-3	IP-STROBE for IP-D, DEFAULT DISABLED
W6	1-2	BOARD CLOCK SETTING, FACTORY SET

6. IP Interface

The VITA 4-1995 standard specification was the latest reference for the IP interface at the time this manual was prepared. For a quick reference, the table below shows the IP pin assignment.

PIN NUMBER	SIGNAL	PIN NUMBER	SIGNAL
1	GND	26	GND
2	CLK	27	+ 5V
3	RESET*	28	R/W*
4	D0	29	IDSel*
5	D1	30	DMAReq0*
6	D2	31	MemSel*
7	D3	32	DMAReq1*
8	D4	33	IntSel*
9	D5	34	DMAck*
10	D6	35	IOSel*
11	D7	36	Reserved
12	D8	37	A1
13	D9	38	DMAEnd*
14	D10	39	A2
15	D11	40	Error*
16	D12	41	A3
17	D13	42	IntReq0*
18	D14	43	A4
19	D15	44	IntReq1*
20	BS0*	45	A5
21	BS1*	46	Strobe*
22	- 12V	47	A6
23	+12V	48	Ack*
24	+ 5V	49	Reserved
25	GND	50	GND

Table 6: IP pin assignment