The STAR DAQ Sector Crate Address Space Architecture and Interrupt Assignment

Tonko Ljubicic
Date: **03/10/98 11:05 AM**
Version: **2.0** (rev. 10)

Introduction

The purpose of this document is to
a) assign and fix all the necessary windows in all of the address spaces in the STAR DAQ sector crates.
b) assign and fix all the interrupt lines, levels and vectors for all of the systems in the sector crate.

This is necessary so that different devices which share a common address space (i.e. VME) do not step on each others windows (or interrupt levels). This was done taking into account
a) the software/hardware requirements of the STAR DAQ system,
b) the intrinsic limitations/possibilities of the various busses (i.e. different cycles, number of available interrupt levels etc.),
c) variety, complexity and possible limitations of the bridge chips (i.e. Tundra Universe, Raven/Falcon etc.) and/or of the systems themselves (i.e. Motorola MVME2604 VME processors).

The intended audience are the software developers who should be familiar with the system anyway so this is the reason why this document is terse.

Address Spaces

1. VME crate space

1.a A32 Address Space

FFFF.FFFF 7000.0000	**Unused**	
5000.0000	**SCI image of the Global Crate**	512 MB
3000.0000	**Sector L3 RAM (reserved)**	512 MB
2F00.0000	**Rosebud**	16 MB
28C0.0000	**Unused**	
28B0.0000	**Receiver Board 12 - CNTRL**	1 MB
	…	
2800.0000	**Receiver Board 1 - CNTRL****	1 MB→maps to 1200.0000 PCI for each RB
2600.0000	**Receiver Board 12 - RAM**	32 MB
	…	
1000.0000	**Receiver Board 1 - RAM***	→ maps to 1000.0000 PCI for each RB 32 MB
0000.0000	**Sector Broker RAM**	256 MB

** Receiver Board RAM VME Image (Addresses are set for the example of CHER 1)*

1200.0000 1180.0000	**Mezzanine C RAM**	8 MB
1100.0000	**Mezzanine B RAM**	8 MB
1080.0000	**Mezzanine A RAM**	8 MB
1000.0000	**CHER SDRAM**	8 MB → maps to 1000.0000 PCI

***Receiver board Control VME image (addresses are for CHER 1)*

2810.0000 280A.0000	**Unused**		→ doesn't map to PCI
  2809.0000	**CHER PCI Configuration Space**	64 kB	→ maps to Configuration Cycles on the PCI bus
2808.0000	**Universe Regs.**	64 kB	
2807.0000	**Mezzanine 03 PLX Regs.**	64 kB	
2806.0000	**Mezzanine 02 PLX Regs.**	64 kB	
2805.0000	**Mezzanine 01 PLX Regs.**	64 kB	
2804.0000	**CHER PLX Regs.**	64 kB	
2803.0000	**Mezzanine C Cntrl.**	64 kB	
2802.0000	**Mezzanine B Cntrl.**	64 kB	
2801.0000	**Mezzanine A Cntrl.**	64 kB	
2800.0000	**CHER VRAM & Cntrl.**	64 kB	→ maps to 1200.0000 PCI

1.b A24 Address Space

FF.FFFF 01.0000	**Unused**
00.FFFF 00.0000	**MVME2604 Location Monitor**

1.c A16 Address Space

FFFF F000	**Rosebud**		
0000	**Xycom I/O Board**	??? kB	← used during tests only

2. Receiver board PCI space

PCI Memory Space

3000.0000	**Unused**		
2F00.0000	**VME A32 (Rosebud)**	16 MB	→ maps to 2F00.0000 VME A32
1301.0000	**Unused**		
1300.0000	**VME A16 (Rosebud)**	64 kB	→ maps to 0000 VME A16
1280.0000	**CHER SDRAM (big-endian)**	8 MB	
120A.0000	**Unused**		
1209.0000	**Reserved for VME→PCI configuration cycles**	64 kB	
1208.0000	**Universe Regs.**	64 kB	
1207.0000	**Mezzanine C PLX Regs.**	64 kB	
1206.0000	**Mezzanine B PLX Regs.**	64 kB	
1205.0000	**Mezzanine A PLX Regs.**	64 kB	
1204.0000	**CHER PLX Regs.**	64 kB	
1203.0000	**Mezzanine C Cntrl.**	64 kB	
1202.0000	**Mezzanine B Cntrl.**	64 kB	
1201.0000	**Mezzanine A Cntrl.**	64 kB	
1200.0000	**CHER VRAM & Cntrl.**	64 kB	← seen at 2800.0000 VME
1180.0000	**Mezzanine C RAM**	8 MB	
1100.0000	**Mezzanine B RAM**	8 MB	
1080.0000	**Mezzanine A RAM**	8 MB	
1000.0000	**CHER SDRAM**	8 MB	← seen at 1000.0000 VME
0000.0000	**VME A32 image (Sector Broker's Memory)**	256 MB	→ maps to 0000.0000 VME

The shaded areas represent the regions visible from the VME bus

PCI Configuration Cycles

- Universe
- CHER SDRAM/VRAM PLX
- Slot 1
- Slot 2
- Slot 3

Device	Device type	PCI ID	Address	IDSEL AD pin
Universe	Universe II	0000.10E3	xxxx.0800	12
CHER VRAM	PLX9080	0000.0000	xxxx.1000	13
Slot 1	PLX9080	0000.0001*	xxxx.1800	14
Slot 2	PLX9080	0000.0002*	xxxx.2000	15
Slot 3	PLX9080	0000.0003*	xxxx.2800	16

* settable by the local side CPU. 0x0000.0000 after reboot.

3. Mezzanine local space as seen from the I960 CPU

	CPU		PCI***			VME*	
VRAM (cachable, little endian)	E000.0000-E03F.FFFF	←	A: 1080.0000-10C0.0000 B: 1100.0000-1140.0000 C: 1180.0000-11C0.0000	mem	←	A: 1080.0000-10C0.0000 B: 1100.0000-1140.0000 C: 1180.0000-11C0.0000	A32
SDRAM (cachable, little endian)	E040.0000-E07F.FFFF	←	A: 10C0.0000-1100.0000 B: 1140.0000-1180.0000 C: 11C0.0000-1200.0000	mem	←	A: 10C0.0000-1100.0000 B: 1140.0000-1180.0000 C: 11C0.0000-1200.0000	A32
VRAM (cachable, big endian)	E080.0000-E0BF.FFFF	x	x	x	x	x	x
SDRAM (cachable, big endian)	E0C0.0000-E0FF.FFFF	x	x	x	x	x	x
VRAM (non-cachable, little endian)	F000.0000-F03F.FFFF	x	x	x	x	x	x
SDRAM (non-cachable, little endian)	F040.0000-F07F.FFFF	x	x	x	x	x	x
VRAM (non-cachable, big endian)	F080.0000-F0BF.FFFF	x	x		x	x	x
SDRAM (non-cachable, big endian	F0C0.0000-F0FF.FFFF	x	x	x	x	x	x
Control registers	C000.0000-C001.0000	←	A: 1201.0000-1202.0000 B: 1202.0000-1203.0000 C: 1203.0000-1204.0000	mem	←	A: 2801.0000-2802.0000 B: 2802.0000-2803.0000 C: 2803.0000-2804.0000	A32
PLX registers	A000.0000-A000.1000	x	A: 1205.0000-1206.0000 B: 1206.0000-1207.0000 C: 1207.0000-1208.0000	mem	←	A: 2805.0000-2806.0000 B: 2806.0000-2807.0000 C: 2807.0000-2808.0000	A32
PCI Memory	0000.0000-9000.0000	→	0000.0000-9000.0000	mem	-	*some PCI addresses map to VME*	-
CPU on-chip RAM**	0000.0000-0000.07FF	x	x	x	x	x	x

* The VME Address is an example for Receiver Board 0. Appropriate offsets must be added for the rest of the Receiver Boards

** The on-chip CPU RAM takes priority over the PCI image so that the first 2 kB of PCI memory are not accessible by the CPU

*** A, B & C denote the 3 Mezzanines.

Mezzanine Control Registers

	Local		PCI*			VME*	
Registers	C000.0000-C001.0000	←	1201.0000-1202.0000	mem	←	2801.0000-2802.0000	A32
ASIC 0 regs.	0000-0040						
ASIC 1 regs.	0040-0080						
…	…						
ASIC 5 regs.	0140-0180						
ASIC_BUFFER	0200						
ASIC_INT	0204						
VRAM_WPBM	0400						
SER_CLEAR	0404						

All registers are byte wide but reside on a 4 byte boundary.

*Appropriate base address for the Mezzanine/Receiver Board should be calculated. RB0-MZ0 shown in the example.

4. Receiver Board RAM and Control Registers

	Local		PCI			VME	
SDRAM	0000.0000-0080.0000	←	1000.0000-1080.0000	mem	←	1000.0000-1080.0000	A32
VRAM	0080.0000-0080.8000	←	1200.0000-1200.8000	mem	←	2800.0000-2800.8000	A32
Registers	0080.8000-0081.0000	←	1200.8000-1201.0000	mem	←	2800.8000-2801.0000	A32
CLR_BSY_A	0000 (RW)						
CLR_BSY_B	0004 (RW)						
CLR_BSY_C	0008 (RW)						
LVL_BSY_A	0010 (RW)						
LVL_BSY_B	0014 (RW)						
LVL_BSY_C	0018 (RW)						
CLR_FEND	0020 (WO)						
FEND_STAT	0024 (RO)						
FEND_ERR	0028 (RO)						
SVT_MODE	002C (RW)						
VRAM_WPBM	0030 (WO)						
BSY_ENABLE	0034 (RW)						

All registers are one byte wide only.
The VRAM is also one byte wide only but resides on a 4 byte boundary. The length is 256 bytes.

5. Address space of the Sector Broker (Motorola MVME2604)

➔ This map corresponds to a modified EXTENDED_VME map as defined in the Motorola documentation.

	CPU		PCI			VME	
DRAM	0000.0000-1000.0000	←	0000.0000-1000.0000	mem	←	0000.0000-1000.0000	A32
VMEA32	1000.0000-5000.0000	→	1000.0000-5000.0000	mem	→	1000.0000-5000.0000	A32
VMEA24	FB00.0000-FBFF.FFFF	→	FB00.0000-FBFF.FFFF	mem	→	00.0000-FF.FFFF	A24
VMEA16	F9FF.0000-FA00.0000	→	F9FF.0000-FA00.0000	mem	→	0000-FFFF	A16
VMECSR	FA00.0000-FAFF.FFFF	→	FA00.0000-FAFF.FFFF	mem	→	00.0000-FF.FFFF	CSR
SCI	5000.0000-7000.0000	→	5000.0000-7000.0000	mem	←	5000.0000-7000.0000	A32
PCI Memory	FD00.0000-FD10.0000	→	FD00.0000-FD10.0000	mem	x	x	x
SCSI Reg.	FD01.0000-FD01.FFFF	→	FD01.0000-FD01.FFFF	mem	x	x	x
SCSI Mem.	FD02.0000-FD02.FFFF	→	FD02.0000-FD02.FFFF	mem	x	x	x
Universe	FD05.0000-FD05.FFFF	→	FD05.0000-FD05.FFFF	mem	x	x	x
PCI I/O	FE00.0000-FE10.0000	→	0000.0000-0010.0000	i/o	x	x	x
ISA	FE00.0000-FE00.FFFF	→	0000.0000-0000.FFFF	i/o	←	00.0000-00.FFFF	A24
SCSI Reg.	FE01.0000-FE01.FFFF	→	0001.0000-0001.FFFF	i/o	x	x	
ENET	FE02.0000-FE02.FFFF	→	0002.0000-0002.FFFF	i/o	x	x	
VGA	FE03.0000-FE03.FFFF	→	0003.0000-0003.FFFF	i/o	x	x	
PMC	FE04.0000-FE04.FFFF	→	0004.0000-0004.FFFF	i/o	x	x	
REGISTERS					x	x	x
Raven MPIC	FC00.0000	←	FC00.0000	mem	x	x	x
Raven/Falcon	FEF8.0000	←	FEF8.0000	mem	x	x	x
ROM	FF00.0000	x	x	x	x	x	x

Interrupts Allocation

VME

Level	Interrupter	Handler
1	Receiver Board - Universe DMA	Sector Broker
2	Receiver Board - Error VME64 AutoID Receiver Board - CHER PLX DMA	Sector Broker
3	Receiver Board - Mezzanine C	Sector Broker
4	Receiver Board - Mezzanine B	Sector Broker
5	Receiver Board - Mezzanine A	Sector Broker

Each Receiver Board owns two VME interrupt vectors - one even numbered (corresponds to the software initiated interrupt) and one odd (corresponds to the PCI/error interrupts).

Vector - software	Vector - hardware	Receiver Board
0x60 (96 dec)	0x61 (97 dec)	1
0x62 (98 dec)	0x63 (99 dec)	2
…	…	…
0x76 (118 dec)	0x77 (119 dec)	12

Receiver Board

PCI/Universe source	VME level	VME vector
Universe DMA	1	odd
LINT3 - VRAM PLX	2	odd
Universe PCI error	2	odd
Universe Auto ID	2	even
LINT2 - Mezz. C	5	odd
LINT1 - Mezz. B	4	odd
LINT0 - Mezz. A	3	odd
Universe Software Interrupt	1	even

Mezzanine

I960 pin	External pin	Priority*	Source
RESET#	PLX USRo#	-	software reset
NMI#	PLX LRSETo#	32	PCI bus reset
XINT7#	ABORT_EVENT	24	Event Aborted
XINT6#	EVTEND	27	Event Done Timeout
XINT5#	BERR	29	Local Bus Error
XINT4#	FLT	28	Link Error
XINT3#	ASIC_INT	30	ASIC Done
XINT2#	EVTWDT	26	Event Watchdog
XINT1#	LSERR	25	PCI Error
XINT0#	PCI_INT	31	PCI Doorbell

Sector Broker (Motorola MVME2604)

The Motorola MVME2604 uses the Raven MPIC as the interrupt controller. The MPIC has 16 input lines (levels) which it assigns to 16 different software programmable vectors, each with software programmable priorities from 1-15 (15 is the highest; 0 disables the interrupt). All of these 16 sources generate one INTR interrupt to the PowerPC processor. The software on the processor reads the vector from MPIC's registers and thus determines the source.
All the other interrupts (the PPC decrementer/timer , local bus error, etc. have higher priority and are not visible to the programmer - which is fine…)

The mapping is taken from the existing map in the VxWorks BSP with but with modified priorities.

Source	Level	Vector	Priority
ISA interrupter - serial ports, aux. timer	0	0x10	15
Falcon ECC error	1	0x11	13
Ethernet	2	0x12	14
SCSI (not used)	3	0x13	0
VGA (not used)	4	0x14	0
Universe LINT0 → VME bus error	5	0x15	12
Universe LINT1 → VME IRQ 1-5	6	0x16	10
Universe LINT2 → local Universe DMA	7	0x17	10
Universe LINT3 → VME IRQ 6-7	8	0x18	10
PMC INTA → SCI	9	0x19	11
PMC INTB (not used)	10	0x1A	0
PMC INTC (not used)	11	0x1B	0
PMC INTD (not used)	12	0x1C	0
Location Monitor 0 (not yet used)	13	0x1D	5
Location Monitor 1 (not yet used)	14	0x1E	5
Unused	15	0x1F	0
Raven ERR → PCI Master./Target Abort	internal	0x50	15