The STAR DAQ
Receiver Board User Manual and Reference

Version 1.0

C. Consiglio, M. J. LeVine, T. A. Ljubicic, R. Scheetz

November 17, 1998

Document Versions

11/17/1998 **Version 1.0** First Beta draft. Describes Receiver Board Production B.

Table of Contents

General Information

Introduction

The STAR DAQ Receiver Board (RB) is a 9U VME board custom built for the readout of the two main STAR detectors: **TPC** and **SVT**. Each of these detectors uses a number of these boards (144 and 24 respectively) on the receiving end of the fiber-optic data chain which transfers the digital detector data from the detectors on the experimental platform to the STAR DAQ in the DAQ room. The Receiver Boards receive the raw data as well as commands in the event headers. The Receiver Boards interpret these commands as necessary and distribute the data stream for further processing and later data storage and online review.

Board Overview

The Receiver Board is a 9U high VME board which acts as a *Mother* board for two separate and distinct buses: the Local data bus and the PCI bus.
The Local bus interfaces to the optical fiber logic and provides the pathway for the digital data from the detectors to the three *Daughter* Mezzanine boards.
The PCI bus interconnects the three Mezzanine boards and the Readout Board's local memory and control to VME and acts as the pathway for the processed data and software control of the RB via the VME bus.

Each Receiver Board provides 22 status LEDs, a manual reset switch, optical fiber input and a TTL compatible BUSY LEMO output all of which are located at the board's front panel.

Features:

- 3 custom connectors for interfacing with 3 Mezzanine boards
- 3 PCI slots with 2 custom connectors each for the 3 Mezzanines
- 64 bytes of Video RAM (VRAM) for event header storage (1 X IBM025161LG5D-60)
- 8 MB local SDRAM memory for general use (4 X Toshiba TC59S1616AFT)
- PLX PCI 9080 Bus Master Interface Chip for the PCI interface to the memories and control registers
- Tundra Universe II PCI-VME bridge chip
- Methode Optical Receiver
- HP HDMP-1024 Glink receiver
- 6 Lattice programmable ISP CPLDs
    - 3 for the fiber and demultiplexing logic
    - 1 PCI bus arbiter
    - 1 local memory and register controller
- PCI slave-only test connector for debugging operations

Functional Description

Architectural Overview

The general architecture of the board may be subdivided between the data/control buses that are contained in the board: the Local bus, the PCI bus and the VME bus.
The Local bus section contains:

- the fiber optic receiver and demultiplexer
- the data stream demultiplexer, event header storage and control
- 3 dedicated connections to each of the Mezzanines
- BUSY logic

The PCI bus section contains:

- PCI devices on the bus
- PCI arbitration
- local SDRAM and VRAM memories
- control registers
- interrupt routing

The VME section is purely governed by the Universe II VME⇔PCI bus bridge.

Local Bus

Front-end

The Methode optical receiver combined with the HP Glink device provides the optical interface that is needed to read the data from the optical fiber to a 20bit wide TTL bus for further handling. The HP Glink, performs a serial to parallel translation of the data from the Methode optical receiver and passes it to a de-multiplexer implemented in a CPLD where the data is sent to one of the three Mezzanine boards via a 20bit wide Local bus. The multiplexer logic on the Receiver board also removes the 64 event header from the front-end data stream and places it into the sequential port of the VRAM for later use during event analysis.

The front-end interface is implemented in three CPLD devices operating at 60 MHz.

- One CPLD handles the state machine for the front-end Glink and the three data strobe pulses required for each of the three Mezzanine boards on the Local bus.
- The second FPGA checks the header block for the proper size and generates the pedestal strobe on the 16^{th} word when in SVT mode. This device also generates various control signals sent to the Mezzanines to indicate End-Of-Event, Fault, Abort, etc. See the “Signal and I/O” section for further details.
- The last FPGA contains a status and error register which indicates the state of the optical link and the state of the front-end logic in general. See the “Register” section for details.

BUSY Output

The board provides one TTL LEMO BUSY output which is enabled via software through a register (see the “Registers” section). All the outputs of all the boards in the crate will be OR-ed together to provide one BUSY output per sector crate.
The Receiver Board sets the BUSY under at least one of the following conditions:

1. Start of Header Command
2. Link Error
3. Link Not Ready
4. At least one of the LVL_BSY_n bits in the corresponding registers is set
5. At least one of the CLR_BSY_n bits in the corresponding registers is set

and resets it under all of the following:

1. Link Ready
2. No Link Error
3. None of the valid bits in the LVL_BSY or CLR_BSY registers is set
4. Clear bit in the CLR_FEND register is pulsed or Abort CAV is received

When the front-end logic detects a Start of Data command (CAV) on the fiber it informs the local controller which in turn sets the CLR_BSY_n bits (making them go to the BUSY state) on all three registers. It is the responsibility of the Mezzanine's software to reset these bits when there is sufficient storage available for the next event. When the last (third) Mezzanine resets the corresponding CLR_BSY_n bit a pulse is issued to the front-end which clears the state machines and prepares the front-end for the next event.

The LVL_BSY_n bits may be used by the Mezzanines to request a pause for any other reason since they are both settable and resettable under software control.

If a Start of Header is issued while the BUSY is set the Receiver Board issues a "Busy Overrun" fault.

PCI Bus

General

The Receiver Board houses five PCI devices:

- three slots for the three Daughter Mezzanines
- one local device which interfaces to the SDRAM, header VRAM and the control registers
- the Universe PCI⇔VME bridge interface chip itself

The PCI bus is a 33 MHz 32bit PCI bus where all the local devices comply to the PCI 2.1 standard. Each of the 3 daughter cards connect to the PCI bus via 2 *non-standard* connectors with a 6 mm board-to-board spacing when mated. The standard PMC connector could not be used due to height limitations. The pinouts of these connectors adhere to the 32bit *3.3V* implementation of the PCI bus with the exact pin numbers provided in the Appendix.

An auxiliary 6th slot exists for debugging purposes only with a normal PCI edge-type connector. The connector itself will not be populated on the first revision of the board but might be used later if PCI expansion is needed.

PCI Configuration Cycles

The PCI Configuration Mechanism conforms to the PCI 2.1 Standard with the following connections of the IDSEL line to the AD lines of the various devices:

	Decoded Address	AD# line connected to devices IDSEL#
Mezzanine 0	0x1800	14
Mezzanine 1	0x2000	15
Mezzanine 2	0x2800	16
local PLX 9080	0x1000	13
Universe	0x0800	12

PCI Configuration Cycle Mapping

Universe bridge as a PCI master/slave

The Universe is fully compliant to the PCI 2.1 standard while acting as a PCI⇔VME bridge. It may issue normal burst/non-burst 32bit PCI transfers and PCI configuration cycles as a PCI master and may be a slave

to most types of PCI accesses including PCI configuration cycles. Refer to Universe's manual for full details.

Local memory and Control

The Receiver Board provides 8 MB of fast SDRAM memory, 64 bytes of header VRAM memory and various control registers through a PLX PCI9080 Rev. 3 bridge chip.
The PLX is a PCI⇔I960 bus high performance bridge chip which is used to provide a PCI interface to the local memories and control registers. These devices are purely PCI slaves although the PLX provides DMA capabilities for transfers from these memories to the PCI bus and as such the PLX may be configured as a PCI master if needed.

These local slaves are controlled through a programmable CPLD which implements the necessary glue logic as well as control registers. The address selection is hard-coded in the CPLD's code :

Device	Local Address
SDRAM	0x0000.0000
VRAM	0x0080.0000
Registers	0x0080.8000

The PCI image of these devices may be placed anywhere in PCI space through the PLX's programmable slave windows.

VRAM

The 64 bytes of VRAM act as the event header storage area. The 64 bytes of the event header are filled by the fiber front-end control logic through the sequential port of the VRAM after which they become available through the random port.
Although the data size is 64 bytes the data occupies 256 bytes of address space due to the fact that the data is aligned on a 4-byte boundary. The upper 3 bytes are unused and should be masked. The following table illustrates this packing:

Relative Address	Header Byte as D32
0x0000	0xXXXX.XX00
0x0004	0xXXXX.XX01
0x0008	0xXXXX.XX02
…	…

The VRAM supports burst accesses up to the full 64 byte length with the 2-1-1-1 wait state pattern for writes and 3-1-1-1 pattern for reads.
The transfer of the sequential FIFO to the Random part storage is accomplished by the CPLD and is transparent to the user. The user must ensure proper initialization of the WPBM mask during startup which is accomplished through a control register implemented in the CPLD. Refer to the device's data sheet for further details and to the Register section below.

SDRAM

The 8 MB of SDRAM may be used as general purpose storage on the Receiver Board. In the current software scenario the only use is during the calculation of the pedestals and gains as intermediate storage.
The SDRAM supports bursts of up to one page in length (1 kB) with the 1-0-0-0 wait state pattern for writes and 2-0-0-0 pattern for reads.

Registers

The Receiver Board implements these registers:

- 3 Busy Reset Registers
- 3 Soft Busy Set/Reset Registers
- 1 WPBM Mask Register
- 1 Front-end Clear Register
- 1 Enable BUSY Out Register
- 1 TPC/SVT Mode Register
- 1 Fiber Frontend Status Register
- 1 Fiber Frontend Error Register

All of these register apart from the last two are implemented in the controller CPLD. The last two are physically implemented in the Local bus front-end logic with the controller CPLD supplying the address decoder only.

All the registers are at most 8 bits wide but reside on a quad-byte boundary similar to the header VRAM above.

For the details of these registers see the “Registers” section below.

Arbitration

A PCI bus arbiter is implemented in a programmable CPLD. The algorithm is round-robin with bus parking on the last bus master which initiated a transaction. The arbitration is compliant with the PCI 2.1 recommendations.

Interrupt Routing

The only PCI Interrupt Handler on the Receiver Board is the Universe chip which translates all PCI interrupt requests to VME interrupts. The PCI INTA# lines of all 4 devices (3 Mezzanines and the local PLX) are connected in order to the LINT pins of the Universe. The Universe expects these interrupts to be PCI compliant i.e. low-level activated.
The PCI INTB#, INTC# and INTD# lines are connected to the remaining LINT lines but are not currently used and are left for expansion possibilities only. Refer to Table for the details.

Source	Universe LINT# input
Mezzanine 0 INTA#	0
Mezzanine 1 INTA#	1
Mezzanine 2 INTA#	2
local PLX INTA#	3
INTB#	4
INTC#	5
INTD#	6
unconnected/unused	7

Interrupt Routing

VME Bus

The Universe II bridge device is responsible for the VME bus part of the Receiver Board and as such provides full master/slave capabilities conformant to the VME standard as well as some extensions in the newer VME64 standard:

- D64 (MBLT) master/slave interface
- VME64 CSR master/slave interface
- VME64 Auto-Id mechanism.

Some VME attributes (i.e. SYSCON, Auto-Id, slave base address etc.) may be configured with the DIP switches provided on the board. Refer to the Universe documentation for full details about these power-up options and to the Appendices for the DIP switch configuration.

In normal operation the Receiver Board should never provide SYSCON functions and all other options will be configured by software during a VME Auto-Id cycle.

For other VME options (slave and master modes, interrupt capabilities etc.) refer to the "Initialization and Configuration" Section.

Initialization and Configuration

Upon power-up, VME SYSRST* or front-panel reset the Receiver Board enters the default state:

- BUSY Output is disabled
- Mode is TPC
- CLR_BSYs are reset
- LVL_BSYs are reset
- The fiber-frontend is cleared
- All the CPLDs are reset

The status of the VME slave image depends on the position of the DIP switches (see the section below) but under normal circumstances the Universe will enter the VME64 Auto-Id cycle requesting VME interrupt level 2 while providing the vector 0xFE.
Upon a successful IACK cycle from the crate controller the Universe opens its CSR slave image in CSR space 0 after which the crate controller programs all the relevant images, windows and attributes and performs the VRAM_WPBM write cycle (see "Registers" section below). The VRAM_WPBM cycle disables the mask register in the VRAM and is necessary for correct VRAM operation. Due to the fact that the data bus at the time of the write must contain a pattern of all '1's it was impractical to initiate this cycle in programmable logic.

The programming of the slave images and attributes as well as the various configuration registers of both the Universe and the PLX9080 is up to the user. However, due to the very complex nature of these bridge chips as well as some limits in the various slave window sizes, offsets and relative offsets a canonical map is given below.

The map pertains to the case of one single Receiver Board in the system but any other boards should be configured similarly with the same offsets but with different VME base addresses. Refer to the document "The STAR DAQ Crate Controller Memory Map" for a full description of the memory map of the whole crate.

VME Slave Map

	VME	cycle	PCI	cycle	attribute	
Universe Registers	2808.0000	A32	1208.0000	mem		2
Universe Registers	CSR 1	CSR	-	-		3
Memory	1000.0000	A32	1000.0000	mem	write posted	
RB SDRAM	1000.0000	A32	1000.0000	mem	write posted	
MZ 0 RAM	1080.0000	A32	1080.0000	mem	write posted	1
MZ 1 RAM	1100.0000	A32	1110.0000	mem	write posted	1
MZ 2 RAM	1180.0000	A32	1180.0000	mem	write posted	1
Aux	2800.0000	A32	1200.0000	mem	coupled cycles	
Header. VRAM	2800.0000	A32	1200.0000	mem	coupled cycles	
RB Registers	2800.8000	A32	1200.8000	mem	coupled cycles	
Mezz. 0 Aux.	2801.0000	A32	1201.0000	mem	coupled cycles	1
Mezz. 1 Aux.	2802.0000	A32	1202.0000	mem	coupled cycles	1
Mezz. 2 Aux.	2803.0000	A32	1203.0000	mem	coupled cycles	1
RB PLX Registers	2804.0000	A32	1204.0000	mem	coupled cycles	
Mezz. 0 PLX Registers	2805.0000	A32	1205.0000	mem	coupled cycles	1
Mezz. 1 PLX Registers	2806.0000	A32	1206.0000	mem	coupled cycles	1
Mezz. 2 PLX Registers	2807.0000	A32	1207.0000	mem	coupled cycles	1
PCI Cfg Cycles	2809.0000	A32	0000.0000	cfg	coupled cycles	

1) For details of the Mezzanine Board Images refer to the Mezzanine documentation.
2) This image doesn't actually initiate transactions on the PCI bus since the Universe's Registers take priority.
3) This image is a copy of the Universe's Registers used only during the VME64 Auto ID Cycle.

VME Master Map

	PCI	cycle	VME	cycle	attribute	
1 MB Sector Controller Image	01F0.0000	mem	01F0.0000	A32	coupled cycles	1
512 MB VME Image	5000.0000	mem	5000.0000	A32/D64	write posted	2

The Master Map is independent of the number of Receiver Boards in the crate.

1) The 1 MB image points to a region of the Sector Controller's memory used for communication only
2) The larger image is used to transfer data out of the Receiver Board and typically points to an SCI board's slave image of the Event Builder and Sector Level 3 Processor.

Appendix A - Registers

CLR_BSY_0				Offset: 00
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	CBSY2	CBSY1	CBSY0

Name	Type	Function
CBSY0	RW	1=Mezzanine 0 BUSY
CBSY1	RO	1=Mezzanine 1 BUSY
CBSY2	RO	1=Mezzanine 2 BUSY

A write to this register (regardless of data content) resets the BUSY status of Mezzanine 0 *only*. The other two mezzanines return status only.

CLR_BSY_1				Offset: 04
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	CBSY2	CBSY1	CBSY0

Name	Type	Function
CBSY0	RO	1=Mezzanine 0 BUSY
CBSY1	RW	1=Mezzanine 1 BUSY
CBSY2	RO	1=Mezzanine 2 BUSY

A write to this register (regardless of data content) resets the BUSY status of Mezzanine 1 *only*. The other two mezzanines return status only.

CLR_BSY_2				Offset: 08
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	CBSY2	CBSY1	CBSY0

Name	Type	Function
CBSY0	RO	1=Mezzanine 0 BUSY
CBSY1	RO	1=Mezzanine 1 BUSY
CBSY2	RW	1=Mezzanine 2 BUSY

A write to this register (regardless of data content) resets the BUSY status of Mezzanine 2 *only*. The other two mezzanines return status only.

LVL_BSY_0				Offset: 10
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	LBSY2	LBSY1	LBSY0

Name	Type	Function
LBSY0	RW	1=Mezzanine 0 BUSY
LBSY1	RO	1=Mezzanine 1 BUSY
LBSY2	RO	1=Mezzanine 2 BUSY

A write to this register sets/resets the BUSY status of Mezzanine 0 *only*. The other two mezzanines return status only.

LVL_BSY_1				Offset: 14
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	LBSY2	LBSY1	LBSY0

Name	Type	Function
LBSY0	RO	1=Mezzanine 0 BUSY
LBSY1	RW	1=Mezzanine 1 BUSY
LBSY2	RO	1=Mezzanine 2 BUSY

A write to this register sets/resets the BUSY status of Mezzanine 1 *only*. The other two mezzanines return status only.

LVL_BSY_2				Offset: 18
Bits	**Function**			
31-24	Ignored			
23-16	Ignored			
15-08	Ignored			
07-00	Ignored	LBSY2	LBSY1	LBSY0

Name	Type	Function
LBSY0	RO	1=Mezzanine 0 BUSY
LBSY1	RO	1=Mezzanine 1 BUSY
LBSY2	RW	1=Mezzanine 2 BUSY

A write to this register sets/resets the BUSY status of Mezzanine 2 *only*. The other two mezzanines return status on reads only.

CLR_FEND	Offset: 20
Bits	**Function**
31-24	Ignored
23-16	Ignored
15-08	Ignored
07-00	Ignored

Name	Type	Function
any	WO	Reset Fiber Front End

A write to this register resets the fiber front-end thus making it ready for the next event. The same effect is accomplished upon the third CLR_BSY_n reset.

FEND_STAT								Offset: 24
Bits	**Function**							
31-24	Ignored							
23-16	Ignored							
15-08	Ignored							
07-00	RDY	BSY	spare	spare	spare	STAT	DATA	HDR

Name	Type	Function
spare	RO	Reserved - returns 0
RDY	RO	1=Optical Link is Ready
BSY	RO	1=Front-end is BUSY
STAT	RO	1=Full Event Received
DATA	RO	1=Start of Event Seen
HDR	RO	1=Start of Header Seen

FEND_ERR								Offset: 28
Bits	**Function**							
31-24	Ignored							
23-16	Ignored							
15-08	Ignored							
07-00	LNKERR	BSYERR	WDTERR	HDRERR	spare	spare	spare	spare

Name	Type	Function
spare	RO	Reserved - returns 0
LNKERR	RO	1=Link Error
BSYERR	RO	1=New event started while the front-end was busy
WDTERR	RO	1=End of Data not seen
HDRERR	RO	1=Header Length not 64

SVT_MODE		Offset: 2C
Bits	**Function**	
31-24	Ignored	
23-16	Ignored	
15-08	Ignored	
07-00	Ignored	SVT

Name	Type	Function
SVT	RW	1=Set SVT Mode 0=Set TPC Mode; RESET condition

A write to this register sets/resets the mode of the fiber front-end thus enabling the pedestal strobe for SVT events. It is set to 0 (TPC mode) upon reset.

VRAM_WPBM	Offset: 30
Bits	**Function**
31-24	Ignored
23-16	Ignored
15-08	Ignored
07-00	FF

Name	Type	Function
FF	WO	Must write 0xFF for correct VRAM initialization!

A write to this register with a value of 0xFF *must* occur before an attempt to use the VRAM. See the VRAM data sheet for further information.

BSY_ENABLE		Offset: 34
Bits	**Function**	
31-24	Ignored	
23-16	Ignored	
15-08	Ignored	
07-00	Ignored	BSY_EN

Name	Type	Function
BSY_EN	RW	1=Enable the LEMO BUSY Output 0=Disable; RESET condition

Appendix B - DIP Switches

A set of four DIP switches is provided to set the power-up options of the Universe device. Most of the functionality of the switches is used for debugging only. Refer to Universe's Manual for further reference.

The positions/meanings of each bit will be provided in this document in a later version.

Appendix C - Mechanical and Electrical Characteristics

The Receiver Board is a 9U high by 400 mm deep VME board. It's an eight layer double sided printed circuit board comprised of four signal layers with two separate power and two separate ground planes.

It uses 3 VME connectors:

- Standard VME P1 and P2
- Auxiliary VME P0 placed between P1 and P2 which provides additional power to 5 V as well as 3.3 V. The pinouts for the P0 connector are in the Appendices.

Power Consumption:

- +3.3V 3 A
- +5.0 V 10 A
- ±12.0 V not used

The currents correspond to a board with all three Mezzanines running under full load!
The 5 V measurement is an approximation only measured with additional boards in the VME crate.

Appendix D - Signals and I/O

Local Bus Signals and I/O Lines to the Mezzanine Boards

1) ABORTP: (pulse true low, two link clocks wide, can be generated anytime while link is ready)(There is also a pulse at power-up or manual reset time.) This signal is used by the Mezzanine boards to reset after an event abort, and also used to reset state machines used to strobe ASIC's (no reply needed)

2) EOFEVNT: (level-true low) End of Event signal to all three Mezzanine boards. Reset by FE_CLR.

3) M1D[19:0]: Data lines to first Mezzanine board (true high)
M2D[19:0]: Data lines to second Mezanine board
M3D[19:0]: Data lines to third Mezzanine board

4) M1A[3:1]: Three strobe signals for first Mezzanine board (true high)
M2A[3:1]: Three strobe signals for second Mezzanine board
M3A[3:1]: Three strobe signals for third Mezzanine board

5) PEDCLK: A strobe pulse sent to all three Mezzanine boards so header data byte 16 can be used (true high) by the SVT boards to store a 'SCA offset count'. A local switch presents the same data on both halves of the 20 bit word.

6) FLT_IRQ: (level-true low) Signal used to inform the processors on the Mezzanine boards of a Fault condition. All three processors should acknowledge this condition before the fault latches are reset by FE_CLR. This is an "OR" of (busyerr, headerbad, linkerr, watchdogtimer) Currently abort also will clear this latch.

7) EENDWDT: (level-true low) Timer started at end of received event , if FE_CLR doesn't arrive before the timer expires this fault will set. Event end Watchdog Timer.

8) MCPURES_M1 CPU A error status [Red LED]
MCPURES_M2 CPU B error status [Red LED]
MCPURES_M3 CPU C error status [Red LED]

Signals From The Front-end to the VRAM Control Logic

1) SER_EN#: (true-low) Serial enable line on VRAM chip to store header information.

2) SER_CLK: (true high) Serial clock line to VRAM chip to strobe in header data.

3) LHD[9:0]: (true high)Latched header data lines to VRAM.

4) HEADDONE: (level-true high) Set when header ends and reset by FE_CLR . Goes to VRAM control .

5) ABORTP: (pulse two link clocks wide-true low) Abort pulse used by VRAM logic .

Signals From The Local Controller to The Front-end

1) FE_CLR: (true-high) Causes a Front-end reset.

2) FE_OE[1:0]: Gate signals from processors to enable reading the Front-end status registers. (allows up to three eight bit registers)

status = FE_OE [01]
error = FE_OE [10]
spare = FE_OE [11]

3) SFTBSYA#: Soft busy from Mezzanine board "A". [Yellow LED]
SFTBSYB#: Soft busy from Mezzanine board "B". [Yellow LED]
SFTBSYC#: Soft busy from Mezzanine board "C". [Yellow LED]

4) CLRBSY0#: Mezzanine board "A" didn’t clear BUSY [Yellow LED]
CLRBSY1#: Mezzanine board "B" didn’t clear BUSY [Yellow LED]
CLRBSY2#: Mezzanine board "C" didn’t clear BUSY [Yellow LED]

5) TPCMODE: Logic 1 selects TPC mode Logic 0 selects SVT mode

6) BSY_ENABLE: Enables the LEMO BUSY Output

Additional Appendices

The following is a set of additional pages inserted later i.e. schematics, exact pinouts of various connectors etc. in no particular order.