

Fermi National Accelerator Laboratory



DZERO Central Tracking Upgrade VME Readout Buffer Controller - VRBC

A. Collantes, D. Mendoza

--PRELIMINARY--

8/25/99

Last revised 1/14/00

D0 Engineering Note # 3823-112-EN-330224

DRAFT

1	GENERAL INFORMATION	1
1.1	System Introduction (Times New Roman 12 for subtitles).....	1
1.1.1	Additional Subtitles (Times New Roman 12 for sub-subtitles)	1
1.2	Description Of Component & How It Fits Into The System	1
1.3	List Of Component Requirements	1
2	THEORY OF OPERATION AND OPERATING MODES	2
2.1	Basic Features & Operation (Including Block Diagram)	2
2.2	Diagnostic Features	2
3	EMBEDDED & DIAGNOSTIC/DEVELOPMENT SOFTWARE	3
3.1	Embedded Software.....	3
3.1.1	Software Tools & Methodologies	3
3.1.2	Description Of Embedded Software	3
3.1.3	Additional Subsections.....	3
3.2	Development & Diagnostic Software	3
3.2.1	Description Of Hardware Test Platform	3
3.2.2	Description Of Software Test Platform	3
3.2.3	Software Tools & Methodologies	3
3.2.4	Test Features	3
3.2.5	User Interface Examples	3
3.2.6	Additional Subsections.....	3
4	INTERFACE SPECIFICATIONS	4
4.1	VMEbus (Or Other Bus) Interface	4
4.1.1	Addressing Modes	4
4.1.2	Data Cycles Types	4
4.1.3	Register Descriptions	4
4.1.4	Additional Subsections.....	4
4.2	Front Panel I/O, Test & Monitoring	4
4.2.1	First Connector.....	4
4.2.1.1	Connector Pin Configurations.....	4
4.2.1.2	Signal Descriptions.....	4
4.2.1.3	Protocols.....	4
4.2.1.4	Additional Subsections	4
4.2.2	Other Connectors	4
4.3	Rear Auxiliary Connector Interface	4
4.3.1	Connector Pin Configurations	4
4.3.2	Signal Descriptions	5
4.3.3	Protocols	5
4.4	Daughterboard Interface	5
4.4.1	Connector Pin Configurations	5
4.4.2	Signal Descriptions	5
4.4.3	Protocols	5
4.5	Other Interfaces (e.g., RS-232).....	5
5	ELECTRICAL & MECHANICAL SPECIFICATIONS.....	6
5.1	Packaging & Physical Size	6
5.2	PC Board Construction.....	6
5.3	Power Requirements.....	6

5.4 Cooling Requirements..... 6
5.5 Additional Subsections..... 6
6 SAFETY FEATURES & QUALITY ASSURANCE PROCEDURES..... 7
6.1 Module Fusing & Transient Supression..... 7
6.2 Other Safety & Quality Assurance Subsections 7
7 EXAMPLE OF COMPONENT OPERATION WITHIN THE SYSTEM..... 8

A APPENDICES..... 9

 A1 List Of Component Documentation 9

 A2 Schematics..... 9

 A3 PAL, FPGA Equations..... 9

 A4 Timing Diagrams 9

 A5 Parts List 9

 A6 Additional Appendices..... 9

1 GENERAL INFORMATION

(Once this document is completed, there will be a formal design review of this project to insure that it properly integrates into the system that it has been built with safety in mind, etc.)

1.1 System Introduction

1.2 Description of Component & How it Fits Into The System

The VME Readout Buffer Controller **VRBC**, is a module designed to remove control traffic from the VME bus during data acquisition; this function is performed by instructing the VRB modules¹ via a custom J3 backplane in a standard VME VIPA crate. In addition, the VRBC allows for a centralized interface to the Trigger System so that Event Data can be readout from the VRB's in the correct order. Trigger information is received through the SCL² receiver mezzanine card, which carries all the information required by the VRBs to be able to operate in a DZERO Run II environment. Figure 1, Shows the location of this module within the overall Central Tracking readout electronics.

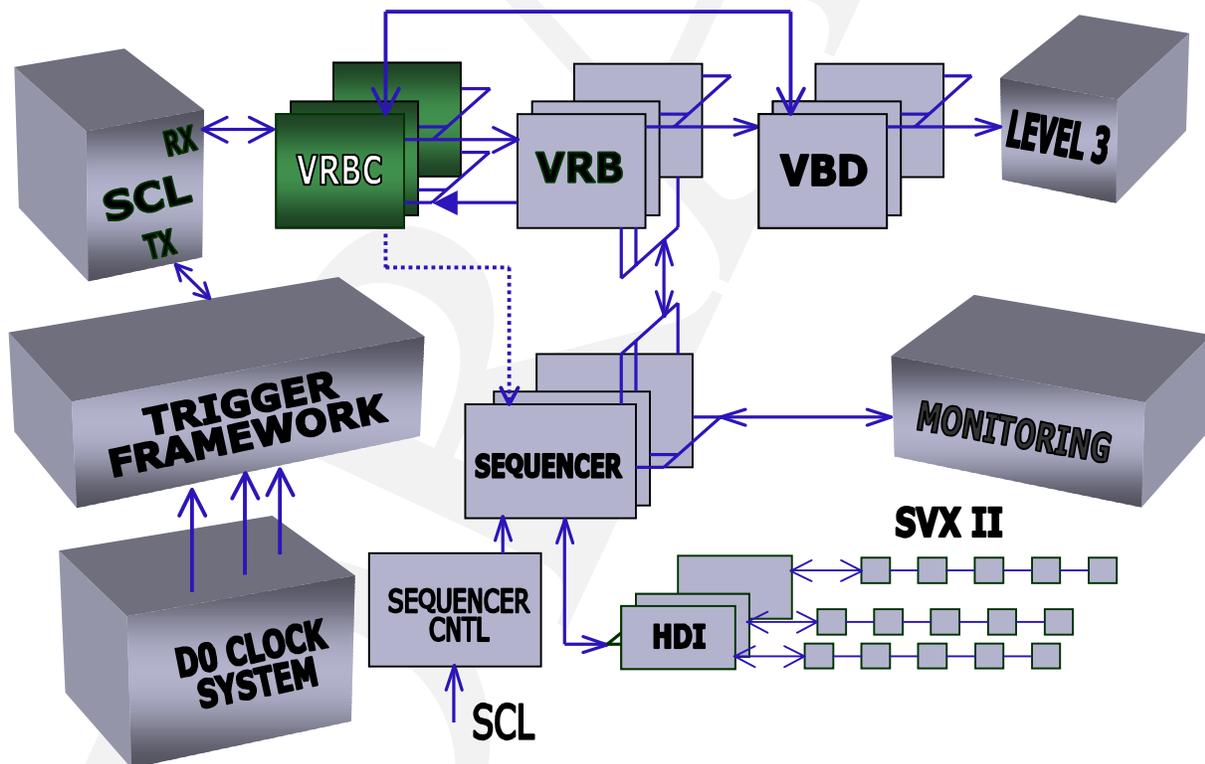


Figure 1. The VRBC in the Central Tracking Readout Electronics

¹ VME Readout Buffer. Description available at <http://www-ese.fnal.gov/eseproj/svx/vrb/vrb.htm>

² SCL, Serial Command Link. Description available at <http://www.pa.msu.edu/hep/d0/ftp/scl> description

1.3 Features and Functions

- ☛ 9U x 400mm VME board
- ☛ VME Slave capabilities. Modes: A32/A24/A16 // D32/D16/D8 read and write cycles
- ☛ Controls up to 14 VRBs in a single VME crate through the J5/6 backplane connector
- ☛ Internal 53.104 MHz. Clock (used in “Autotest” mode)
- ☛ *In-Situ* re-programmability
- ☛ Provides a Crate ID number
- ☛ Operates in Auto-test mode if the SCL is not present
- ☛ Operates as a VME, level programmable, interrupt requester
- ☛ Records relevant real time event information for diagnostic purposes when certain types of errors occur.
- ☛ Front panel connectors available for the various signals for diagnostic and monitoring purposes.
- ☛ Front Panel LEDs indicate the status of the System.

1.4 Block Diagram

Figure #2 depicts a block diagram of the VRBC module and a cursory description follows.

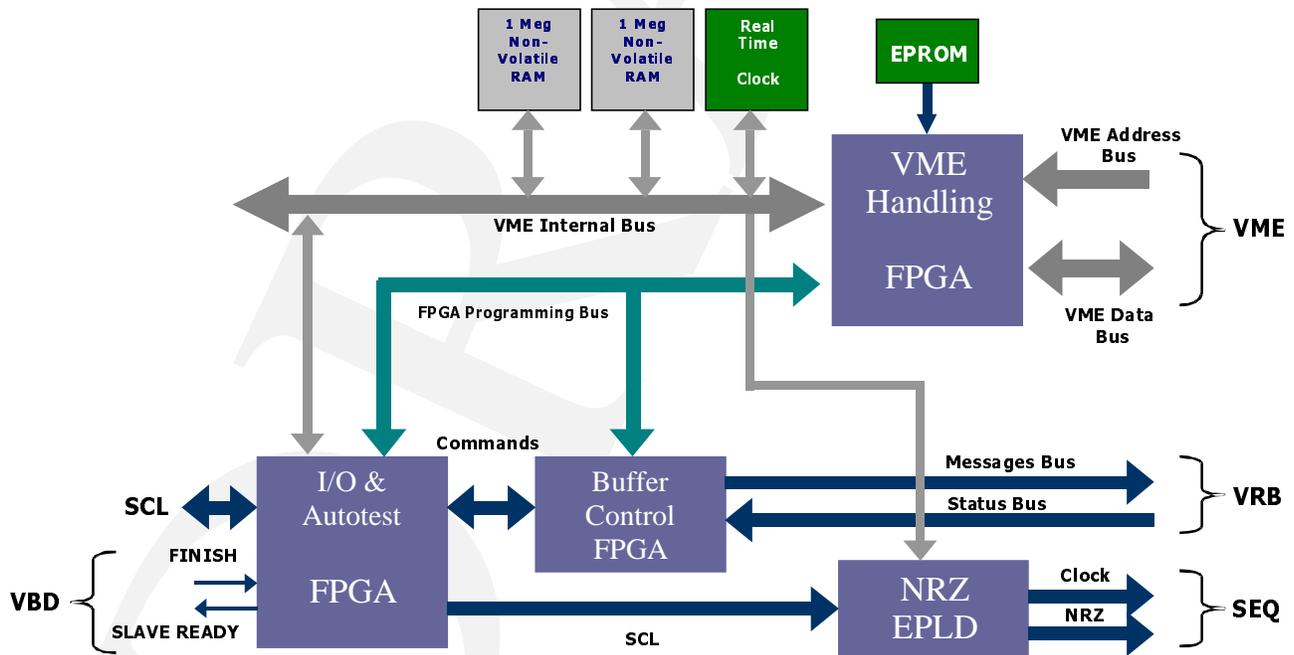


Figure # 2. VRBC Block diagram

The VRBC acts as the interface between the VRB modules and the D0 Clock System and Trigger Manager System. Signals received from these systems are decoded and distributed to the VRBs in the crate. The transport of the commands from VRBC to VRB is done via a custom J3 backplane.

Internally, the VRBC consists of four main functional blocks, shown in blue color in Figure#2.

- ☛ VME Handling,
- ☛ Autotest and I/O
- ☛ Buffer Control
- ☛ Non- Return to Zero Code Generator (NRZ)

Each one of these blocks is implemented in a Field Programmable Field Array (FPGA) device, except the NRZ generator, which fits into an Erasable Programmable Logic Device (EPLD).

Being the FPGAs SARAM-based devices, an EPROM configuration device (Altera part # EPC1) with an on-chip oscillator and its own length counter, allows the *VME Handling* FPGA to be either, reprogrammed whenever the design changes, or initialized at power up. At power-up the other two FPGAs are reconfigured, over an internal programming bus. This is accomplished by transferring into them programming data previously stored in non-volatile memories.

Alternatively, all these devices support the JTAG (Joint Test Action Group) programming method, which is IEEE Std. 1149.1-1 compliant. In this case the FPGAs lose their programming information whenever a power shutdown takes place.

1.5 Addressing Modes and Memory Map

The VRBC responds to Short I/O, Standard and Extended addressing modes and data transfers are achieved in 8,16 and 32-bit formats. Table #1 shows the assignment of VME space and the different addressing modes to access it.

Addressing Mode	Address Range (Hex)
Short I/O	AB00 – ABFF
Standard I/O	480000 – 48FFFF
Extended I/O	80480000 – 8048FFFF

Table #1. VRBC Addressing Modes

A total of 64K bytes are assigned to each module in a given crate in the DZERO environment. The distribution of this memory space assignment, is carried out according to the following memory map:

Address	Description	R/W	Size
80480000	Non volatile RAM 0 *	R/W	Byte
80480002	Non volatile RAM 1 *	R/W	Byte
80480004	Stamp Time *	R	Byte
80480006	Non volatile RAM (Start/Stop) *	R	Byte
80480008	Crate ID Number	R	Word
8048000A	Reset *	R	Word
8048000C	Interrupt Status/ID Register (vector address)	R/W	Byte
8048000E	Interrupt Control Register	R/W	Byte
80480090	Time Set – seconds	W	Word
80480092	Time Set – minutes	W	Word
80480094	Time Set – hour	W	Word
80480096	Time Set – day	W	Word
80480098	Time Set – date	W	Word
8048009A	Time Set – month	W	Word
8048009C	Time Set – year	W	Word
8048009E	Time Set – century	W	Word
804800A0	Program FPGA s *	R	Word
8048800A	Write commands in Autotest mode *	W	Word
8048800C	Autotest mode 1 *	R	Word
8048800E	Autotest mode 2 *	R	Word
80488010	Autotest mode 3 *	R	Word
80488012	Normal Operation *	R	Word

Address	Description	R/W	Size
----------------	--------------------	------------	-------------

80488014	L3 transfer Number	R	Word
804800A4	Write Command to NRZ	W	Word
804800B0	Modulus Counter - Interrupts frequency	R/W	Word
804800B2	Path Control – Data Path setting	R/W	Byte
804800B4	Serial Number	R	Byte
80480010-8048008E	Event error & diagnostic information (non-volatile Time-Keeping RAM)	R/W	Word
(*) These memory locations are not registered on the board; they act as a memory locations to issue commands. For instance, reading the memory location 8048000A causes a RESET signal to be generated. This may be interpreted as a command.			

Table #2. VRBC Memory Map

Internally, the address bus is 16 bits wide and the data bus is 32 bits wide.

1.6. Interfaces

The VRBC Module communicates with the outside world according to the diagram shown in Figure #2. The systems that communicate with this board, uni-directionally or bi-directionally are:

- ☛ VME Processors (VME)
- ☛ Serial Command Link (SCL)
- ☛ VME Readout Buffer (VRB)
- ☛ VME Buffer Driver (VBD)
- ☛ SVX Sequencer

1.6.1. VME

The VRBC can be accessed from any VME processor in the crate capable of performing the following addressing modes and data cycles:

Address Modifier	Description	Data Cycles	Add. Range
3D	Standard Supervisory Data Access	D8/D16/D32	480000–48FFFF
39	Standard Non-privileged Data Access	D8/D16/D32	480000–48FFFF
0D	Extended Supervisory Data Access	D8/D16/D32	80480000–048FFFF
09	Extended Non-privileged Data Access	D8/D16/D32	80480000–048FFFF
2D	Short Supervisory Data Access	D8/D16/D32	AB00 – ABFF
29	Short Non-privileged Data Access	D8/D16/D32	AB00 – ABFF

All the required signals for accessing VRBC's internal registers and memory locations are available through the standard J1 & J2 VIPA compliant connectors.

1.6.2 SCL

Communication with the Serial Command Link is carried out via the proposed PC-MIP Type II Receiver mezzanine card³. Signals and their description are shown in Table #4.

Signal	Description
SCL_READY	Serial Command Link Ready Status
SCL_SYNC_ERROR	Serial Command Link Synchronization Error
SCL_DATAERROR *	Serial Command Link Data Error
SCL_ACK	Acknowledge & Clear SCL Error Flags
CLK_53	53 MHz Clock
CLK_7	7.5 MHz Clock
CURRENT_TURN[15..0]	Current Turn Number
CURRENT_BX[7..0]	Current BX Number in this Turn
FIRST_PERIOD *	First Period in a Turn Marker
BEAM_PERIOD *	Period with Beam Marker
SYNC_GAP	Sync Gap Marker (no L1 Accepts)
COSMIC_GAP *	Cosmic Gap Marker (only Cosmic L1 Accepts)
SPARE_PERIOD *	Spare Period Marker
L1_PERIOD *	Period with L1 Accept Issued
L1_ACCEPT	L1 Accept to this Geo Section
L1_TURN[15..0]	Level 1 Turn Number
L1_BX[7..0]	Level 1 BX Number in this Turn
L1_QUAL[15..0]	L1 Accept Qualifiers
L2_PERIOD *	Period with L2 Decision Issued
L2_REJECT	This Geo Section L2 Reject
L2_ACCEPT	This Geo Section L2 Accept

³ Specifications available in <http://www-ese.fnal.gov/d0trig/sclrev.pdf>

INIT_SECTION	Initialize Geographic Section Flag
L1_BUSY	Busy L1 Status
L2_BUSY	Busy L2 Status
L1_ERROR	Error L1 Flag
L2_ERROR	Error L2 Flag
INIT_ACK	Init Acknowledge Signal to Hub-End
SYNC_LOST *	SCL Receiver Synchronization Lost
SPARE_STATUS[1..0]	Spare Status Signals to Hub-End

Table 4. SCL – Signals as seen by the VRBC

(*) Signal no used in the present version [as of 10/22/99]

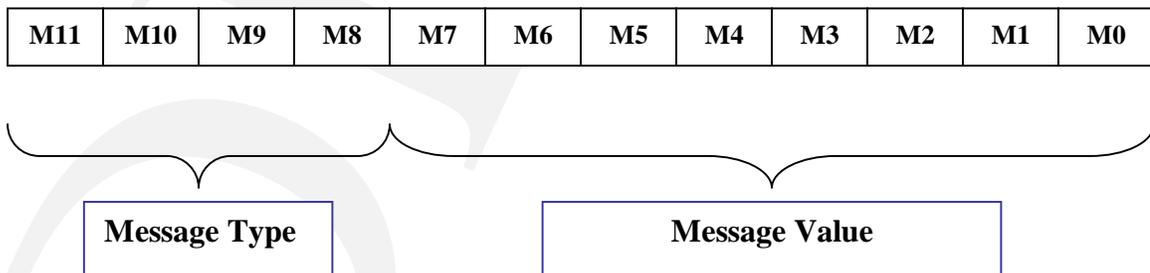
All these signals are fed into the Autotest and I/O FPGA for subsequent use and processing.

1.6.2. VRB

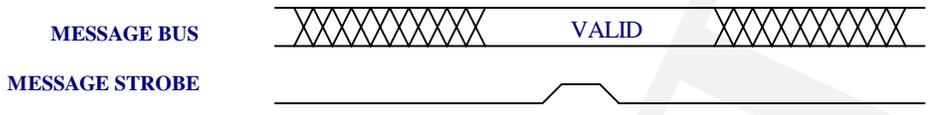
Communications with the VME Readout Module is performed via a Message Bus and a Status Bus across the J5/6 backplane connector. The primary function of this interface to the VRB boards is to provide control on how the buffers on the VRB are selected (for reading and writing) during SVX data acquisition and VME readout. Upon reception of a L1 Accept signal, the VRBC provides the VRB with a buffer number for storing data. Upon reception of a L2 Accept signal, the VRBC provides the VRB with the buffer number whose data will be transferred to the VBD module over VME.

A well-defined protocol allows communication between the VRBC and the VRB boards. A 12-bit Message Bus transports information to the VRB, which, in turn, reports its status through a 10-bit Status Bus to the VRBC, so that the proper actions are taken in normal data taking or for diagnostic purposes.

The *Message Bus* consists of a 4-bit field indicating the type of message and an 8-bit field with the message value. The bit assignments follows:



Each transmission over the Message Bus is qualified by the rising edge of a MESSAGE STROBE signal as shown below.



The following table contains a cursory description of these signals, the reader is encouraged to read reference ¹ for a more detailed explanation.

Message Type		Message Value Description
#	Name	
1	Readout Buffer Number	Contains the value of the next buffer number for VRB Input. Value ranges from 0 to 63. Received After L1 Accept
2	Not used	N/A
3	Bunch Xing Number	Contains the Bunch Crossing Number, which forms the lower 8 bits of the 24-bit Geographic Section Beam Xing Number. Received After L1 Accept
4	Scan Buffer Number	Contains the value of the next buffer number for VRB Output. Value ranges from 0 to 63. Received After L2 Accept.
5	Event Number	Contains the Event Number to be inserted into the data stream. Received After L2 Accept.
6-12	Not used	N/A
13	Clear Errors	Clears any error condition on the Status Bus
14	Reset / Re-Start	Value Reset: Resets the input FIFOS on the VRB. Value Re-Start: Reinitializes the entire VRB.

As for the **Status Bus**, each bit indicates a status signal from the VRBC. These signals are ALL active low and are latched at the VRBC. Their description follows:

Bit #	Name	Description
0	Readout Busy	Kept active when the active channels on the VRB are receiving data. Released when ALL the channels have received data.
1	Scan Busy	Kept active when transferring data to the VBD. Released when all the event data has been transferred over VME
2	Sync Error	Becomes active when a G-Link sync. Error has been detected. Kept asserted while the error condition exists
3	Frame error	Becomes active when an invalid data word has been detected on an active channel. Remains asserted until the next Message Type 1.
4	Identifier Error	Becomes active when an invalid event identifier has been detected. Remains asserted until the next Message Type 1
5	Format Error	Becomes active when a data format error has been detected. Remains asserted until the next Message Type 1.
6	Controller Error	Becomes active when an invalid message has been sent to the VRB. Remains asserted until a Message Type 13 is received.
7	VRB Error	Becomes active when the VRB was unable to process an event correctly. Remains asserted until a Message Type 1 is received.
8,9	Reserved	N/A

1.6.3. VBD

The VBD module moves event data over VME from the VRBs after the Trigger Framework has issued a L2 Accept. DMA transfers are used to move data to one of the two 256 Kbytes buffers on the VBD.

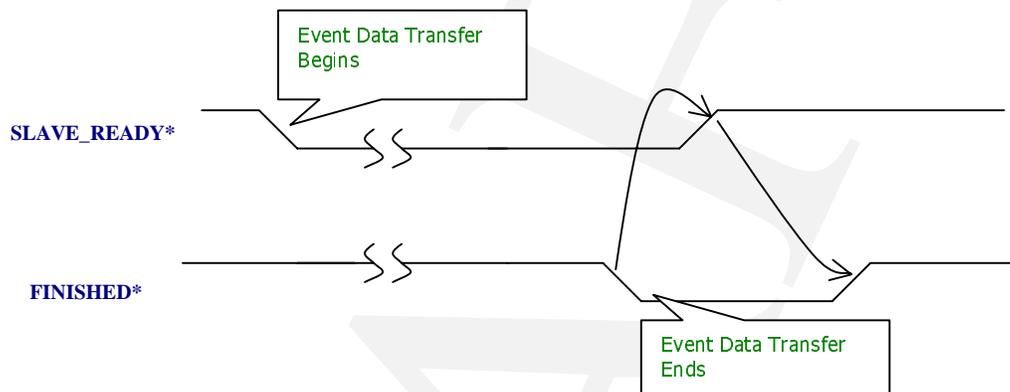
Two signals are required to execute the handshaking between the VME Buffer Driver module and the VRBC: SLAVE READY* and FINISHED*. They are both assumed to be open collector and negative true logic. Their description follows:

SLAVE READY*: An active LOW signal that informs the VBD that the VRB buffers are full and ready to be readout.

FINISHED*: An active LOW signal that informs the VRBC that the VBD has finished reading the VRB buffers.

The acquisition cycle begins when the VRBC announces that a new event data transfer is about to occur after a L2 Accept, this is done by asserting its SLAVE READY signal; the VBD then arbitrates for the bus. Once the mastership of the bus is granted, it is not relinquished until the entire event has been transferred; at that time, FINISHED is asserted and SLAVE READY becomes inactive allowing a subsequent event data transfer.

A timing diagram may help to illustrate this hand-shaking concept:

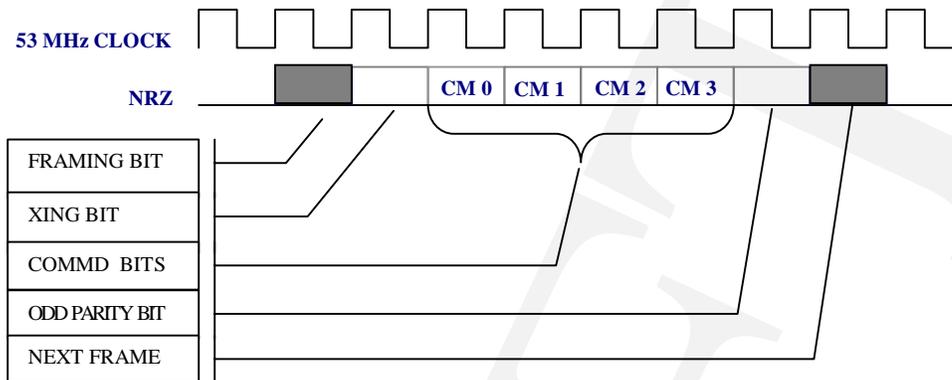


1.6.4. SVX Sequencer⁴

The VRBC provides an alternate path for the Master Clock (53.104 MHz), synchronous to the Tevatron, and the serial 7-bit frame Non-Return-to-Zero (NRZ) encoded signal. These signals are required by the SVX Sequencer board to control the SVX chips for data acquisition. These signals are ECL compatible and available on the front panel twinax connectors. No phase adjustment method is provided on board to compensate for delays due to different length cables or any other factor of skewing. Those adjustments should be provided at the receiver end.

Four bits, out of the seven in the 7-bit NRZ packet are encoded commands. The other three are a framing bit, a parity bit and a crossing bit. Their location within the 7-bit frame is shown below.

⁴ SVX Sequencer Board. Description available at <http://d0server1/users/utes/default.htm>



The decoding of the 4-bit commands is as follows:

Command Bits				Hex	Description
Binary					
CM3	CM2	CM1	CM0		
0	0	0	0	0	Idle
0	0	0	1	1	Acquire
0	0	1	1	3	Trigger
1	0	1	0	A	Ramp
0	0	1	0	2	Digitize
0	1	1	0	6	Readout
0	1	0	1	5	Reset_Preamp
0	1	1	1	7	Cal_Inject
1	0	0	0	8	Reserved
1	0	0	1	9	Reserved
0	1	1	0	4	Reserved
1	0	1	1	B	Reserved
1	1	0	0	C	Reserved
1	1	0	1	D	Reserved
1	1	1	0	E	Reserved
1	1	1	1	F	Reserved

1.7 Interrupts

The VRBC is capable of generating interrupt request on the VME bus allowing synchronization with the embedded processor residing in the VME crate. This feature may allow moving data over the network whenever the L3 System becomes inactive.

Release Mechanism

In the DZERO environment, the code for interrupts uses the computational model RORA, (Release On Register Access) as the release mechanism. With this approach, the interrupt generator (VRBC) only releases its request when the INTERRUPT HANDLER, within the interrupt service routine, accesses a specific register. In the present version of the VRBC, this register is named “INTERRUPT CONTROL REGISTER”.

Phases

Three phases are required to accomplish a successful RORA interrupt operation:

- i. Phase 1. Interrupter awaits service
- ii. Phase 2: Interrupt Handler reads Status/ID from the Interrupter:
- iii. Phase 3: Execution of Interrupt Service Routine

Status/ID (Vector Address) Register

This is an 8-bit R/W register that contains the VRBC’s identification required by the Interrupt Handler to determine which module in the crate requested service to an interrupt. The ID must be placed by the processor into this register before any attempt to issue an interrupt request transaction.

Interrupt Control Register

This is an 8-bit R/W register that, when accessed, is used by the Interrupt handler to release the interrupt request line according to the RORA scheme. It also provides three (3) signals that may be utilized by the handler in the Interrupt Service Routine. These signals are:

- ***Interrupt Enable/Disable***⁵: Bit 0

This signal, allows/disallows the generations of interrupt requests when a new data event was received or when an interrupt request was forced by the Interrupt Reset signal. The power-up value of this bit is LOW (disable)

- ***Interrupt Reset Signal***⁵. Bit 1.

When read indicates the status of the interrupt request:

- 0 – No Interrupt Requested
- 1 – Interrupt Request in Progress

When written:

- 0 - The interrupt request line is dropped.
- 1 – Generates an interrupt request transaction.

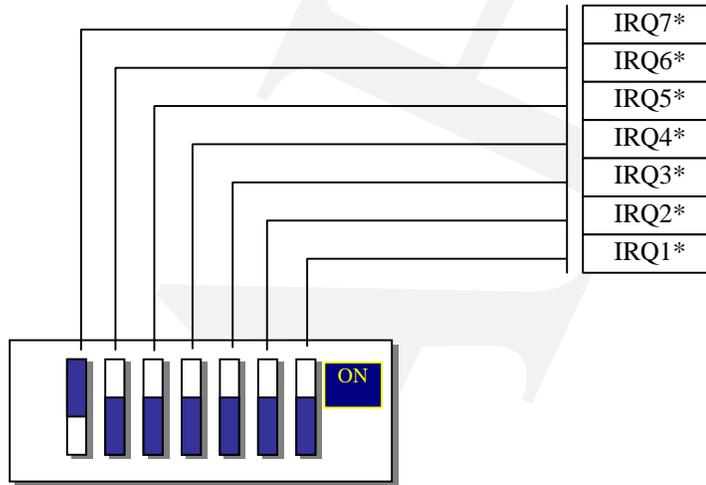
The power-up value of this bit is LOW.

⁵ By document written by Fritz Bartlett, JUL/99

- **Veto**⁵. Bit 2

This signal, allows/disallows the processing of further events by the VRBC. The power-up value of this signal is 0. (Processing allowed)

Interrupt Levels. Seven priority levels are used on the VRBC, IRQ1* through IRQ7*. All of them are active LOW signals, being IRQ7* the one with highest priority. Only one of them can be used at a time. (IRQ1* OR IRQ2* OR...OR IRQ7*). This is accomplished by setting the dip- switch S1 to the required level as show below.



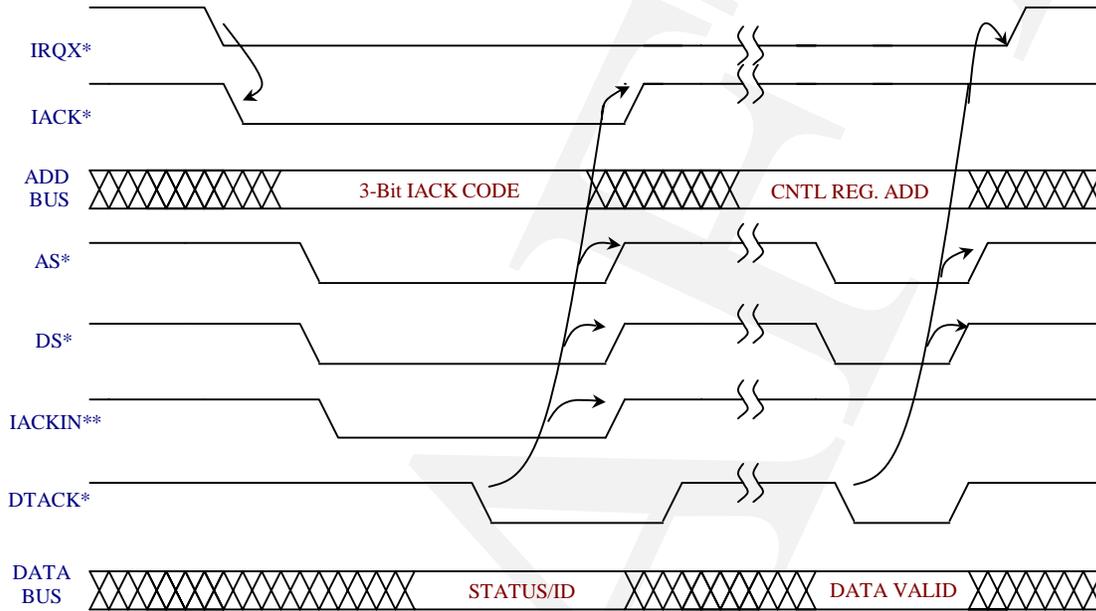
Dip-switch S1, Interrupt Level Setting (IRQ7* as an example)

Sources of Interrupt Requests

Either software or hardware can generate an interrupt request. The former is referred to the Interrupt Control Register explained earlier in this section, and the latter is referred to the front panel LEMO connector input. This input is TTL compatible, 50-ohm terminated and buffered inside the board. When the Secondary Data Path is enabled (discussed later in this document), the arrival of an event can cause an Interrupt as well, upon selecting the appropriate mode of operation; in this case the frequency of the interrupts programmed through VME.

Timing Diagram

A timing diagram is presented next to illustrate how an interrupt request is processed. The relevant signals are presented.



Register Addresses

The applicable VME addresses used to process an interrupt request are:

8048000C	Interrupt Status/ID Register (vector address)	R/W	Byte
8048000E	Interrupt Control Register	R/W	Byte
804800B0	Counter for Interrupt frequency Setting	R/W	Word

The meaning of each register will be explained in 2.5 in this document.

2 THEORY OF OPERATION AND OPERATING MODES

This section will discuss in some detail several of the VRBC components as shown in the Block Diagram in Figure #2. We will begin with the functionality of each FPGA.

2.1 VME FPGA.

This FPGA takes care of the communications with the VME world. Physically it takes the form of a 240-pin Altera device. Part Number EPF10K30-240.

The main sections of its logic design are a VME decoder section, a 10K-device programmer section, a Primary and Secondary data paths manager, a memory management section and the interrupt management section. Details of its implementation can be found in “VRBC VME Readout Buffer Controller, Users Guide V 1.0, June 1999” By A. Collantes.

2.2 AUTOTEST & I/O FPGA

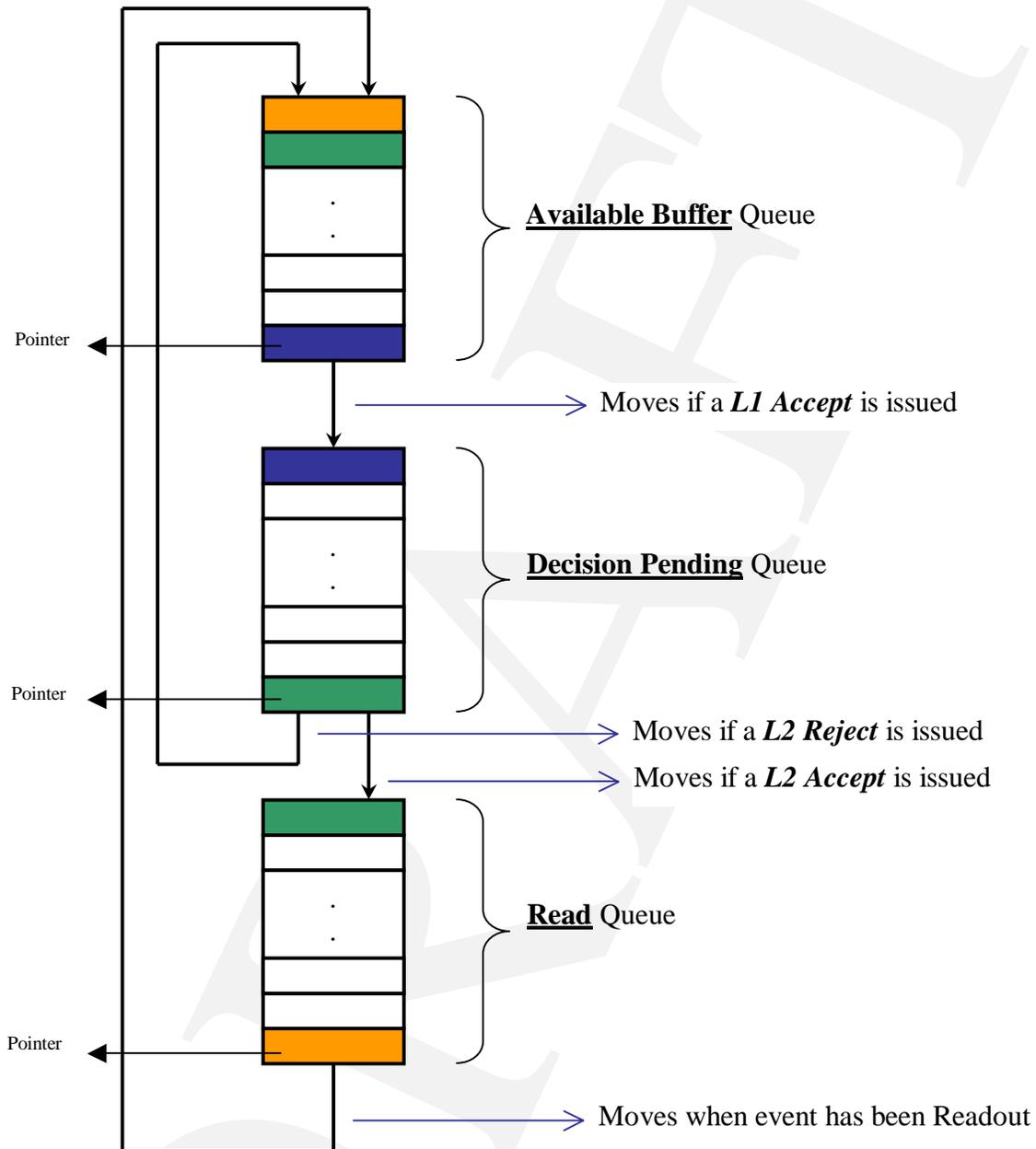
The VRBC is the entity in a Geographic Section (GS) that makes use of the Serial Command Link signals, decoding them, processing them and sending them out to some of the modules of the Silicon Readout System, namely the VRBs, the VBD and the SVX Sequencer. Status information of the GS as well as error conditions is transmitted back to the SCL hub end. The primary function of this FPGA is to decode the signals provided by the Trigger Framework and to translate them in to commands to guide the event data acquisition process according to the D0 Clock System signaling. Physically it takes the form of a 240-pin Altera device. Part Number EPF10K30-240.

The relevant sections of the design implemented in this device are Autotest, Compfif_time, L3TN, Stamp_Time, and Decod. Again, details of its implementation can be found in “VRBC VME Readout Buffer Controller, Users Guide V 1.0, June 1999” By A. Collantes

2.3. BUFFER CONTROL FPGA

This FPGA Physically it takes the form of a 240-pin Altera device. Part Number EPF10K30-144. The logic inside essentially controls which of the VRB 's data buffers will be filled (with data from the SVX chips) and read (by the VBD or any processor over the VME bus).

The VRBC uses the concept of buffer number queues (and their associated buffer pointers) to keep track of which data buffers are available for filling, which data buffers are holding data of an event for which a decision is pending and which data buffers are to be read via VME. The management of these three queues, previously mentioned, is visualized in the next diagram. Any queue behaves as a FIFO system.



At power-up, the queue of *available buffers* will indicate that all the buffers are available, that is, event data will be stored in any of the VRB buffers beginning with the first one. That is where the buffer pointer is pointing.

Upon arrival of a Level One Trigger- (L1) signal, the queue of *available buffers* will hold the buffer number to be filled with SVX data. After a L1 Accept signal, the buffer number pointed by the Available Buffer pointer is passed to the *Decision Pending* queue where, if a L2-Accept signal is issued, this buffer number will be transferred to the bottom of the *Read* queue.

Otherwise (meaning L2 Reject was issued) it is transferred to the Top of the *Available* queue. Once in the *Read* queue, the buffer number sits there until the event is readout by the VBD and then moved to the *available* queue where it can be re-used.

The current design implemented a total of 24 buffer numbers per queue. A “BUSY” flag is raised and sent to the Trigger Manager whenever the pending queue is FULL. Subsequent L1 Trigger signals are suppressed having no effect on the buffer management.

2.4. NRZ EPLD

2.5 Register Descriptions

2.5.1. Non-Volatile RAMs

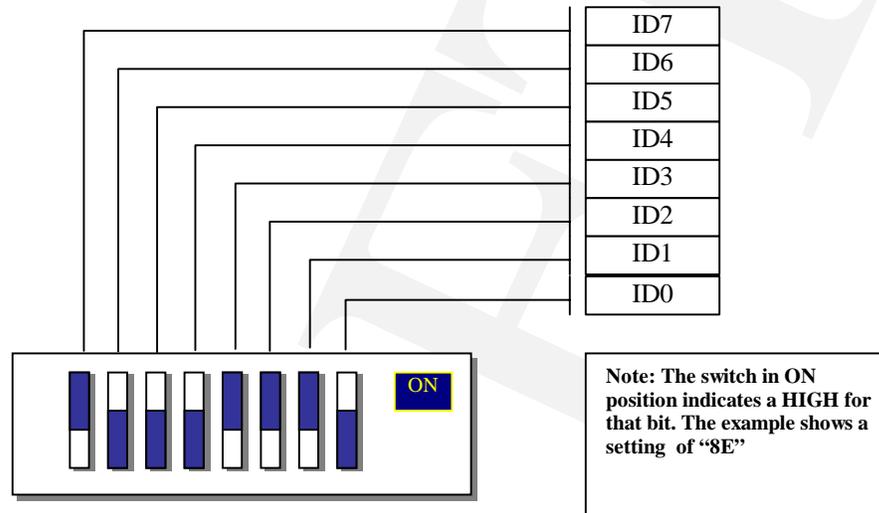
The non-volatile RAMs are used to store the programming data for the AUTOTEST and BUFFER CONTROL. These devices are not mapped onto the VME address space, therefore provisions are taken to write into them by addressing the memory locations indirectly. Three (3) VME addresses are provided to achieve this, they are shown in the table below.

VMEADD	Description	Cycle	Size
80480000	Non volatile RAM 0 *	R/W	Byte
80480002	Non volatile RAM 1 *	R/W	Byte
80480006	Non volatile RAM (Start/Stop) *	R	Byte

. The mechanism is as follows: By reading the memory location 480006 hex, the writing cycle begins for a particular device (RAM0 or RAM1). If, for instance, the RAM0 is selected by a write operation in the memory location 480000 hex, data will be written into the device’s memory location 0 hex. Further accesses to this address will increment the address pointer one by one. The procedure ends when a new reading operation is performed on the memory location 480006 hex; finishing the write cycle for that particular device.

2.5.2. Crate ID Number

This is 16-bit, read-only register. The low order byte indicates the 8-bit crate ID as shown in the front panel selector; the high order byte indicates the 8-bit crate ID set by the on-board dip-switch **S1** as shown in the figure below.



VME ADD	Description	Cycle	Size
80480008	Crate ID Number	R	Word

2.5.3 Reset

A read cycle on this memory location will perform a RESET function to the FPGAs logic

VME ADD	Description	Cycle	Size
8048000A	Reset *	R	Word

2.5.4. Interrupt Status/ID Register

VME ADD	Description	Cycle	Size
8048000C	Interrupt Status/ID Register (vector address)	R/W	Byte

This is an 8-bit R/W register used to identify the module that is causing an interrupt and requires to be serviced by the Interrupt handler. This register must be downloaded prior to the enable of the interrupt requests generated by the VRBC.

2.5.5. Interrupt Control Register

VME ADD	Description	Cycle	Size
8048000E	Interrupt Control Register	R/W	Byte

This is an 8-bit R/W register that, when accessed, is used by the Interrupt handler to release the interrupt request line according to the RORA scheme. It also provides three (3) signals that may be utilized by the handler in the Interrupt Service Routine. These signals are:

- **Interrupt Enable/Disable⁷**: Bit 0

This signal, allows/disallows the generations of interrupt requests when a new data event was received or when an interrupt request was forced by the Interrupt Reset signal. The power-up value of this bit is LOW (disable)

- **Interrupt Reset Signal⁵**. Bit 1 .When read indicates the status of the interrupt request:

0 – No Interrupt Requested

1 – Interrupt Request in Progress

When written:

0 - The interrupt request line is dropped.

1 – Generates an interrupt request transaction.

The power-up value of this bit is LOW.

- **Veto⁵**. Bit 2

This signal, allows/disallows the processing of further events by the VRBC. The power-up value of this signal is 0. (Processing allowed)

2.5.6. Stamp Time

VME ADD	Description	Cycle	Size
80480004	Stamp Time *	R	Byte

A read cycle on this memory location will retrieve the Real Time Clock information from the Time-keeping RAM when an event error occurs and will place it into the appropriate memory locations as shown in Table # 2 of this document.

⁷ By document written by Fritz Bartlett, JUL/99

2.5.7. Real Time Clock Settings

A group of write-only memory locations (from 480090 through 48009E) allows programming the settings of the Real Time Clock embedded into the Non-volatile Time-keeping RAM. The VME addresses and their functions are shown in the table below.

VME ADD	Description	Cycle	Size
80480090	Time Set – seconds	W	Word
80480092	Time Set – minutes	W	Word
80480094	Time Set – hour	W	Word
80480096	Time Set – day	W	Word
80480098	Time Set – date	W	Word
8048009A	Time Set – month	W	Word
8048009C	Time Set – year	W	Word
8048009E	Time Set – century	W	Word

These addresses are basically a translation of the device's memory locations to the VME address space. For completeness the reader is encouraged to get more detailed information and parameters entry format at the Web site

<http://www.dalsemi.com/DocControl/PDFs/1743p.pdf>

2.5.7. FPGAs Programming

VME ADD	Description	Cycle	Size
804800A0	Program FPGA s *	R	Word

A READ cycle on this memory location will perform the re-programming of the AUTOTEST & IO FPGA and the BUFFER CONTROL FPGA with the contents of the programming data previously stored in the non-volatile RAMs. The process is identical to that done at power-up.

2.5.8. AUTOTEST Modes

In the absence of the Serial Command Link, the VRBC can operate in AUTOTEST mode, meaning that some of the SCL signals can be emulated and external input signals can be used to

test the VRBC functionality and its communication with the outside world. When working in any of the *Autotest modes*, a local crystal oscillator provides the 53.104 MHz clock required by the logic to properly emulate the SCL link signals. The setting of any of these modes will override the clock signal originated at the Trigger Framework.

There is provision for three different Autotest modes:

Autotest mode 1: In this mode, some SCL signals can be emulated using VME commands. These commands are stored sequentially into an internal Dual Port Memory (DPM) and executed one at the time every 132 ns. Up to 16 commands can be stored in the DPM and their definition follows:

Value	Description
1	Init
2	L1 Accept
3	L2 Accept
4	L2 Reject
5	Sync Gap
6	Finished
7-15	Spare

The “Value” (1-15) specified in the first column is the number to be written into the VME address specified below in order to emulate the signal specified in the second column.

VME ADD	Description	Cycle	Size
8048800A	Write commands in Autotest mode *	W	Word

To execute the sequence of signals previously written into the DPM, a VME READ transaction should be performed on the following VME address:

VME ADD	Description	Cycle	Size
8048800C	Autotest mode 1 *	R	Word

Autotest mode 2: In this mode an external Test Beam L1 signal (see front panel diagram for its location) is required to launch the sequence of signals required to process an event in the absence of the SCL. The associated VME address to issue this command is:

VME ADD	Description	Cycle	Size
8048800E	Autotest mode 2 *	R	Word

Autotest mode 3: Same as above except that it requires an external Laser L1 Trigger and the resulting trigger signal is delayed by 396 ns. The associated VME address to issue this command is:

VME ADD	Description	Cycle	Size
80488010	Autotest mode 3 *	R	Word

2.5.9. Normal Operation Mode

This is the mode of operation of the VRBC by default. In this mode, all the signals required to perform a DAQ cycle come directly from the SCL Mezzanine Card. After the selection of any of the Autotest modes, a Normal Operation can be resumed by reading the VME memory location shown below.

VME ADD	Description	Cycle	Size
80488012	Normal Operation *	R	Word

2.5.10. L3 Transfer Number

VME ADD	Description	Cycle	Size
80488014	L3 transfer Number	R	Word

This is a 16-bit read-only register whose contents is the Geographic Section Level 3 Transfer Number that is being 16 assigned to the event that just received an L2 Confirm during a 132 nsec period.

2.5.11. VME NRZ coding

VME ADD	Description	Cycle	Size
---------	-------------	-------	------

804800A4	Write Command to NRZ	W	Byte
-----------------	-----------------------------	----------	-------------

By writing an encoded byte into this memory location, a sequence of NRZ commands is generated depending upon the current state of the NRZ finite state machine (FSM). The latter ensures that the SVX II chip timing requirements are met. The following diagram of the FSM illustrates the different sequences of commands that can be obtained to instruct the SVX II.

VME ADD	Description	Cycle	Size
804800B0	Modulus Counter - Interrupts frequency	R/W	Word
804800B2	Path Control	R/W	Byte

The VRBC is capable of controlling the path that event data will follow during data taking. To understand this concept, let's define two possible paths the data may take:

Primary Data Acquisition Path (PDAQ)

This is the normal path the data follows during data taking, That is, data is transferred from all the VRB modules to the VBD and subsequently moved to the Level-3 Trigger System.

Secondary Data Acquisition Path (SDAQ)

For run II, it is likely to have a SDAQ path for all of the DZERO sub-detectors. The SDAQ is intended for use during commissioning, calibration runs and whenever the Level-3 Trigger System and its attendant high-speed data path become unusable. It provides a mechanism to move data over the network from the embedded processor in the VME crate to the host servers of the on-line systems. When this path is adopted, the VRBC will generate an interrupt request (whose frequency is programmable) to the embedded processor which, in turn, will respond issuing a veto and so inhibiting further event acquisitions.

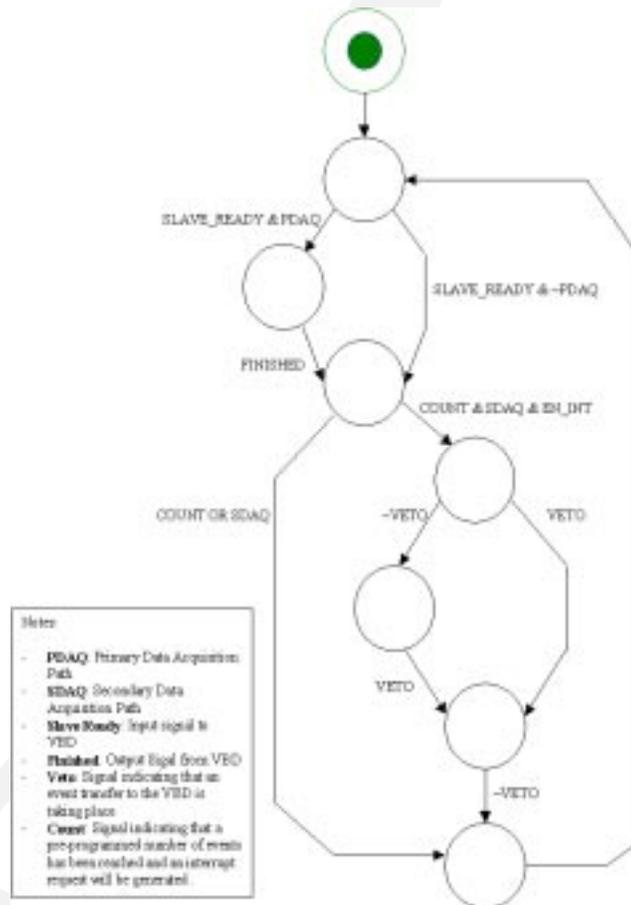
In order to accomplish a data path control, two VME registers are supplied on the VRBC board. A path control register which allows to program the path that the data will follow during data taking and a Modulus Counter Register whose contents defines the number of events preceding an interrupt (modulus counter).

The registers' bit assignment follows:

REGISTER	Description	Value(hex)
Path Control	None of the Paths is used	00
Path Control	PDAQ Path is used*	01
Path Control	SDAQ Path is used	02
Path Control	Both Paths are used	03
Modulus Counter	The Value defines the number of events prior to generating an interrupt	XXXX

* Default Value

Once, the mode has been set, the data acquisition process complies with the FSM shown below:



2.5.13. Serial Number

VME ADD	Description	Cycle	Size
804800B4	Serial Number	R	Byte

Each one of the VRBC modules has a Serial Number readable through VME as a means to be identified for monitoring and diagnostic purposes. The setting of the bits is hardwired on the board.

EMBEDDED & DIAGNOSTIC/DEVELOPMENT SOFTWARE

(This section is not written until later in the development phase, not too much later though.)

3.1 Embedded Software

3.1.1 Software Tools & Methodologies

3.1.2 Description of Embedded Software

3.1.3 Additional Subsections

3.2 Development & Diagnostic Software

3.2.1 Description Of Hardware Test Platform

3.2.2 Description Of Software Test Platform

3.2.3 Software Tools & Methodologies

3.2.4 Test Features

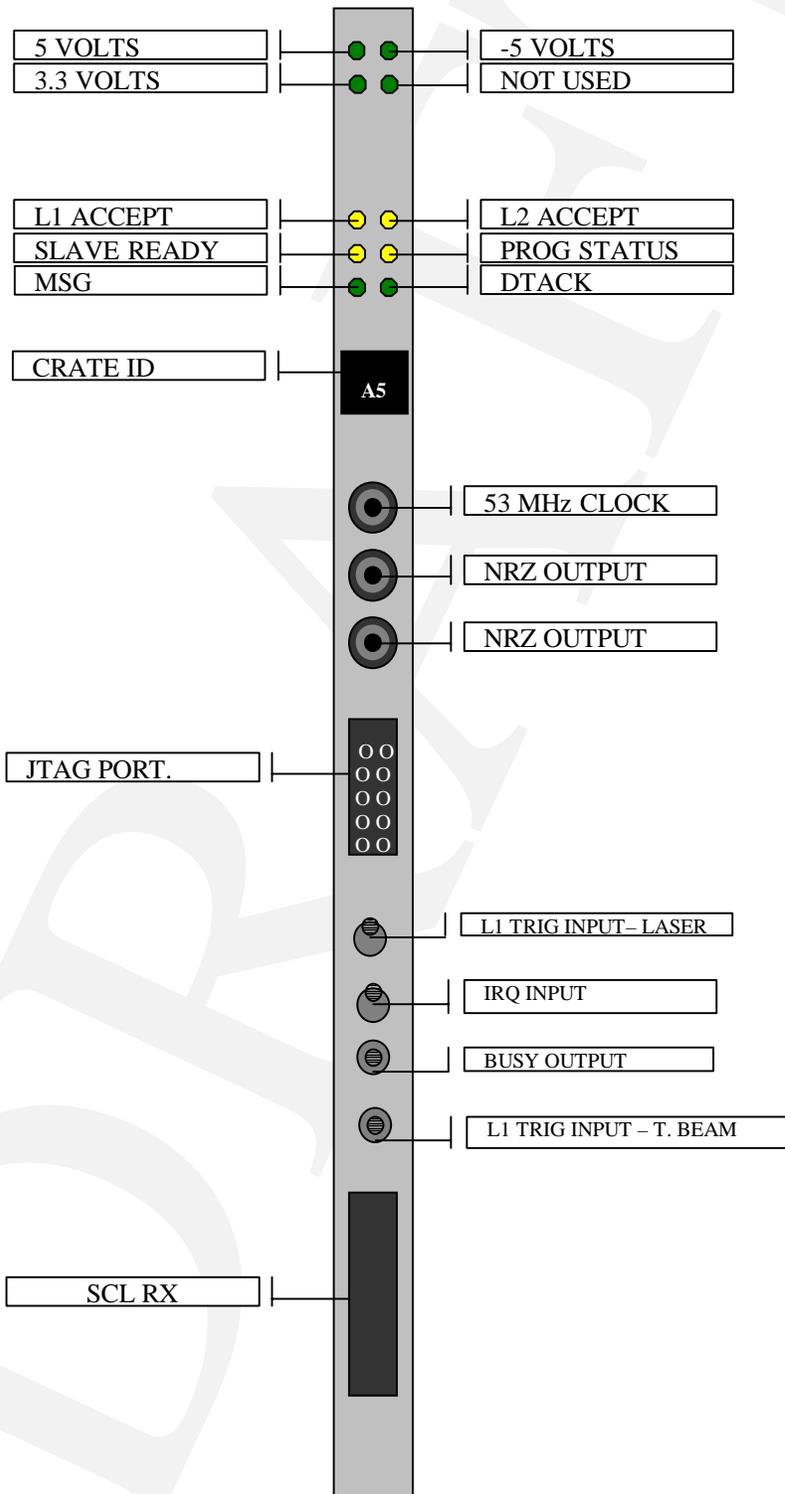
3.4.1 (These paragraphs list and describe the tests, which can be performed with the software.)

3.2.5 User Interface Examples

(These paragraphs give examples of how to run these tests.)

3.2.6 Additional Subsections

VRBC Front Panel



LABEL	FORM	DESCRIPTION
5 Volts	GREEN LED	5 Volts present on the board
-5 Volts	GREEN LED	- 5 volts present on the board
3.3 Volts	GREEN LED	3.3 volts present on the board
L1A	YELLOW LED	When it flashes, a L1 Accept signal was issued
L2A	YELLOW LED	When it flashes, a L2 Accept signal was issued
SLVRDY	YELLOW LED	When it flashes a Slave Ready Signal was asserted by the VRBC
PROG	YELLOW LED	When blinking, the programming of the On-board FPGAs was successful
MSG	GREEN LED	When it flashes, a message was sent to the VRB via J3 backplane conn.
DTK	GREEN LED	When it flashes A VME transaction took place
CRATE ID	SELECTOR	Lower portion (8 bits) of the Crate ID
NRZ	TWINAX CONN.	ECL output of the NRZ signal
CLK	TWINAX CONN.	ECL output of the 53 MHz Clock
NRZ_CFT	TWINAX CONN.	ECL Output of the NRZ signal (to the Central Fiber Tracker if needed)
L1 LASER	LEMO CONN	TTL Input - L1 Trigger signal for Laser scan test
IRQ	LEMO CONN	50-0hm terminated TTL Input - External Interrupt Request signal
BUSY	LEMO CONN	TTL Output VRBC Busy signal. Active HIGH when an event data transfer takes place. The Secondary Data Path must be enabled
L1 T.BEAM	LEMO CONN	TTL Input - L1 Trigger signal for Test Beam test
SCL	NA	Opening to bring in the LM R-200 Coax cable from the Serial Link fan-out modules (Trigger System)

4 INTERFACE SPECIFICATIONS

(These paragraphs give details of all I/O of the subject of this document including bus interfaces, front panel I/O and monitoring points {e.g., LEDs}, auxiliary card interface and/or daughtercard interface. If the subject of this document requires an auxiliary and/or daughter card, separate documents should be written for each of these cards as well. Figures of the front-panel and its connectors, etc. as well as auxiliary and/or daughter card connectors, etc. are almost always included.)

4.1 VMEbus (Or Other Bus) Interface

4.1.1 Addressing Modes

4.1.2 Data Cycles Types

4.1.3 Register Descriptions

4.1.4 Additional Subsections

4.2 Front Panel I/O, Test & Monitoring

4.2.1 First Connector

4.2.1.1 Connector Pin Configurations

4.2.1.2 Signal Descriptions

4.2.1.3 Protocols

4.2.1.4 Additional Subsections

4.2.2 Other Connectors

4.3 Rear Auxiliary Connector Interface

(Add sections as required to describe the operation of the interface including signal descriptions, operations performed, timing requirements, etc.)

4.3.1 Connector Pin Configurations

4.3.2 Signal Descriptions

4.3.3 Protocols

4.4 Daughter-board Interface

(Add sections as required to describe the operation of the interface including signal descriptions, operations performed, timing requirements, physical size, mounting requirements, etc.)

4.4.1 Connector Pin Configurations

4.4.2 Signal Descriptions

4.4.3 Protocols

4.5 Other Interfaces (e.g., RS-232)

5 ELECTRICAL & MECHANICAL SPECIFICATIONS

5.1 Packaging & Physical Size

5.2 PC Board Construction

5.3 Power Requirements

5.4 Cooling Requirements

5.5 Additional Subsections

6 SAFETY FEATURES & QUALITY ASSURANCE PROCEDURES

(Details of all component safety features such as module fusing as well as methods to assure high quality of this component such as environmental testing should be included in this section. Note that this is not a request but is a requirement of Fermilab that modules are built in a safe manner. We will not be able to install and use any equipment that is not built to operate safely. In fact, this system and all its components will be thoroughly reviewed by official lab safety personnel to certify its safety)

6.1 Module Fusing & Transient Suppression

(Printed circuit board layouts should include fusing and transient suppression for all incoming voltages. Fuses and transient suppression diodes should be placed as close as possible to the voltage entry point on the board. Fuses should be sized such that the board is protected against a power short on the board. A single fuse per input voltage should be used. Using fuses in parallel is not good engineering practice and should not be implemented. Add fusing and transient protection to this section)

6.2 Other Safety & Quality Assurance Subsections

7 EXAMPLE OF COMPONENT OPERATION WITHIN THE SYSTEM

A APPENDICES

A1 List Of Component Documentation

A2 Schematics

A3 PAL, FPGA Equations

A4 Timing Diagrams

A5 Parts List

A6 Additional Appendices