

Fermi National Accelerator Lab

## BLM Upgrade Controller Card Program Design

Randy Thurman-Keup  
**AD / Instrumentation Department**

August 31, 2006

BEAMS-DOC-2433-V1  
CVS Code Version v08

### *Abstract*

This document details the structure and behavior of the program that runs in the eZ80 on the Controller Card. The main functionality of this program is to start and stop data acquisition and to maintain data buffers that are accessible from VME at any time without disrupting the aborting capability. These data buffers consist of three circular buffers containing summed data at short, medium, and long integration times, two linear buffers housing profile and flash frames, and a single entry buffer for the most recent display frame.

## 1 Introduction

The Controller Card (CC) is an embedded processor (eZ80) board residing in the VME crate of the Beam Loss Monitor (BLM) system [1]. The program's job is to transfer information between the crate processor and the digitizer cards (DC), timing card (TC), and abort card (AC), in a deadtimeless fashion, *i.e.* no interruption in the aborting capability of the system. The program operates with a combination of polling and interrupts.

The Controller Card Program (CCP) responds to 5 types of events.

- **TCLK**  
The TC puts relevant TCLK events into its FIFO. The CCP polls the TCLK FIFO status register to determine if there is data in the FIFO. If there is data, it reads it from the FIFO and handles it.
- **MDAT**  
When the TC receives the relevant MDAT frame, it writes the state information to the MDAT FIFO. The CCP polls the MDAT FIFO status register to determine if there is data in the FIFO. If there is data, it reads it from the FIFO and handles it.
- **DATA\_LATCH (*Interrupt*)**  
This is an interrupt generated by the Timing Card when it is time to latch some flavor of Digitizer Card data (fast, slow, or very slow sums). The CCP must then read the status bytes in the Timing Card to determine which data to latch.
- **Abort\_Service**  
The abort card generates an interrupt when one of three user selectable states occurs.
  - A digitizer card indicates that a channel is not OK
  - A digitizer channel is over one of the thresholds but the multiplicity requirement is not met for an abort
  - An abort has occurred

The CCP does not respond to the interrupt. Instead it periodically polls the AC to determine the above information.

- **Crate Processor**  
When settings need to be updated, the CP writes the settings to the CC and sets an appropriate register in the CC which then responds by loading the settings into the appropriate cards at the appropriate time.

The CCP might need to do intelligent checking of the state of cards and possibly issue a Reset.

## 2 Initialization

At boot time, the CP and CC must handshake to properly bring up the system. This handshaking between CC and CP is documented in Figure 1 below. The initialization procedure includes stopping the CC at the beginning of the procedure. Stopping the CC at boot time is necessary for a number of reasons: first, at crate power up time, the other cards will not have gone through their FPGA programming sequence if the CC immediately starts to access them; second, the waiting gives the CP time to download settings to the CC; and third, if the CC reboots by, e.g. hitting an invalid instruction, it would probably be good to notify the outside world, and save a

snapshot of the CC's debug memory contents for later analysis. The settings that need to be downloaded to the CC are listed in Table 1.

- **Timing Card**

The timing card must be downloaded with the appropriate TCLK events (see Table 3). In addition to the events listed in Table 3, there are several other TCLK and BSCLK events that the CCP does not respond to but which are responded to by the other cards in the system.

- TCLK \$8F – 1 Hz event for updating the TC clock
- TCLK \$5B – TeV Pbar Injection TBT trigger
- TCLK \$5C – TeV Proton Injection TBT trigger
- BSCLK \$AA – Revolution marker used to generate Make\_Meas clock
- BSCLK \$DA – MI TBT trigger

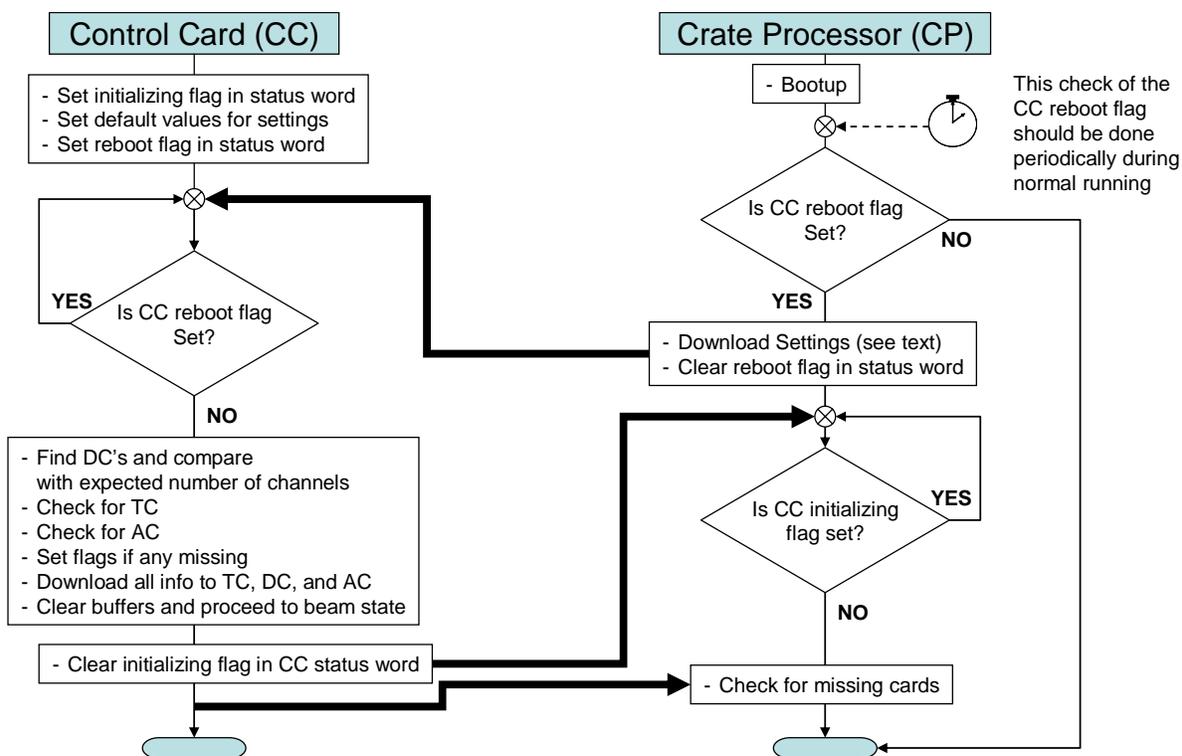
- **Digitizer Card**

The digitizer card must be setup for the correct memory map. The correct map is enabled by setting bit 7 of address 0xFF to 1. All other settings are part of the CP download.

- **Abort Card**

The abort card settings are all downloaded as part of the CP download.

After successfully initializing, the CCP continuously polls for events, periodically interrupted by the Data Latch interrupt.



**Figure 1: Flowchart of handshaking that occurs at boot time. The alarm clock symbol indicates that the CP should periodically check the reboot bit in the CC and if it finds it set, it should probably notify the outside world somehow. See Section 3 for a description of the flags in the status word.**

**Table 1: Settings that need to be set on the CC by the CP. The default value is the value that is used by the CCP if not overwritten by the CP and is a compile time setting in the CCP.**

| Setting              | Description   | Default Value                                | Address(Size) |
|----------------------|---|--|---------------|
| System Time          | Unix time in seconds since ???  | 0  | 0x09800014(4) |
| TC Operation mode    | Value to be written to TC control bus CSR. Controls whether TC uses AA marker or internal oscillator.                                 | 4 for TeV<br>0 for MI                        | 0x0980010E(2) |
| Make Measure Div     | Amount to divide down the clock by (int. osc.)  | 1 for TeV<br>2 for MI                        | 0x09800102(2) |
| Fast Sum Length      | # of measurements to accumulate in DC fast sum.   | ~64  | 0x09800104(2) |
| Slow Sum Length      | # of measurements to accumulate in DC slow sum.   | ~1500  | 0x09800106(2) |
| Very Slow Sum Length | # of measurements to accumulate in DC very slow sum. (In MI this is the integral and is the same as Fast Sum Length)                  | ~50000 in TeV<br>~64 in MI                   | 0x09800108(2) |
| DC Mode Selects      | Digitizer channel mode select word (one word / channel). Controls integration mode and whether squelch is enabled among other things. | 0x0002 for TeV<br>0x000A for MI (no squelch) | 0x09800200(2) |

| Setting                        | Description  | Default Value                                 | Address(Size)   |
|--------------------------------|--|---|-----------------|
| DC Manual Set Value            | The manual setting value if manual setting mode is chosen  | 0x0000  | 0x09800202(2)   |
| # of Channels                  | Expected number of channels  | Whatever is present                           | 0x09800100(2)   |
| Abort Info                     | Thresholds, Masks, and Multiplicities for all States   | Everything 0xFF                               | 0x09900000      |
| Machine                        | Which machine is this (1=TeV, 2=MI, 3=Booster)   | 2   | 0x0980001C(2)   |
| Initial state                  | Initial machine state  | 0   | 0x0980001E(2)   |
| DC FPGA Control Register       | Funny name for something which contains the number of make_meas to skip before doing pedestals divided by 16 ( $N_{\text{skip}} / 16$ )              | 0x1000  | 0x0980010A(2)   |
| IRQ3 Enable Bits               | 3 bits that determine what generates an IRQ3 interrupt on the AC   | 0x0007  | 0x09800112(2)   |
| Squelch Value                  | This must be in the appropriate units which is $X(16\sigma_{\text{raw}}\sqrt{n_{\text{vs}}})$ where $X$ is the # of sigma to place the squelch       | 0x0000 for now since we are not squelching    | 0x09800400(120) |
| Pedestal Length                | Must be 16 * Very Slow Sum Length in MI (doesn't matter in TeV)  | 1024  | 0x09800116(2)   |
| Abort Enable                   | Enables the functioning of the AC  | 1   | 0x09800114(2)   |
| Flash, Display, Profile source | This determines the source of the first half of the frame is (0=fast or 1=slow). Bit 0 is for Flash, bit 1 is for Profile, and bit 2 is for Display. | 0x0006 (uses fast for flash, slow for others) | 0x09800006(2)   |
| End of beam delay              | Delay (in Fast latch periods) after end-of-beam event before asserting AIP   | 18  | 0x09800118(1)   |
| Flash Delay                    | Delay from flash frame clock event until data is grabbed from buffer   | 0   | 0x0980011A(1)   |
| Profile Delay                  | Delay from profile frame clock event until data is grabbed from buffer   | 0   | 0x0980011C(1)   |
| Display Delay                  | Delay from display frame clock event until data is grabbed from buffer   | 0   | 0x0980011E(1)   |
| Input Switch state for peds    | Whether or not to open the input switch while taking pedestals   | 1   | 0x09800120(2)   |

### 3 Program Components

#### 3.1 CC VME Status Register

The status register consists of 16 bits in VME memory. Table 2 shows the bit definitions of the status word.

**Table 2: Bit definitions in the CC Status word.**

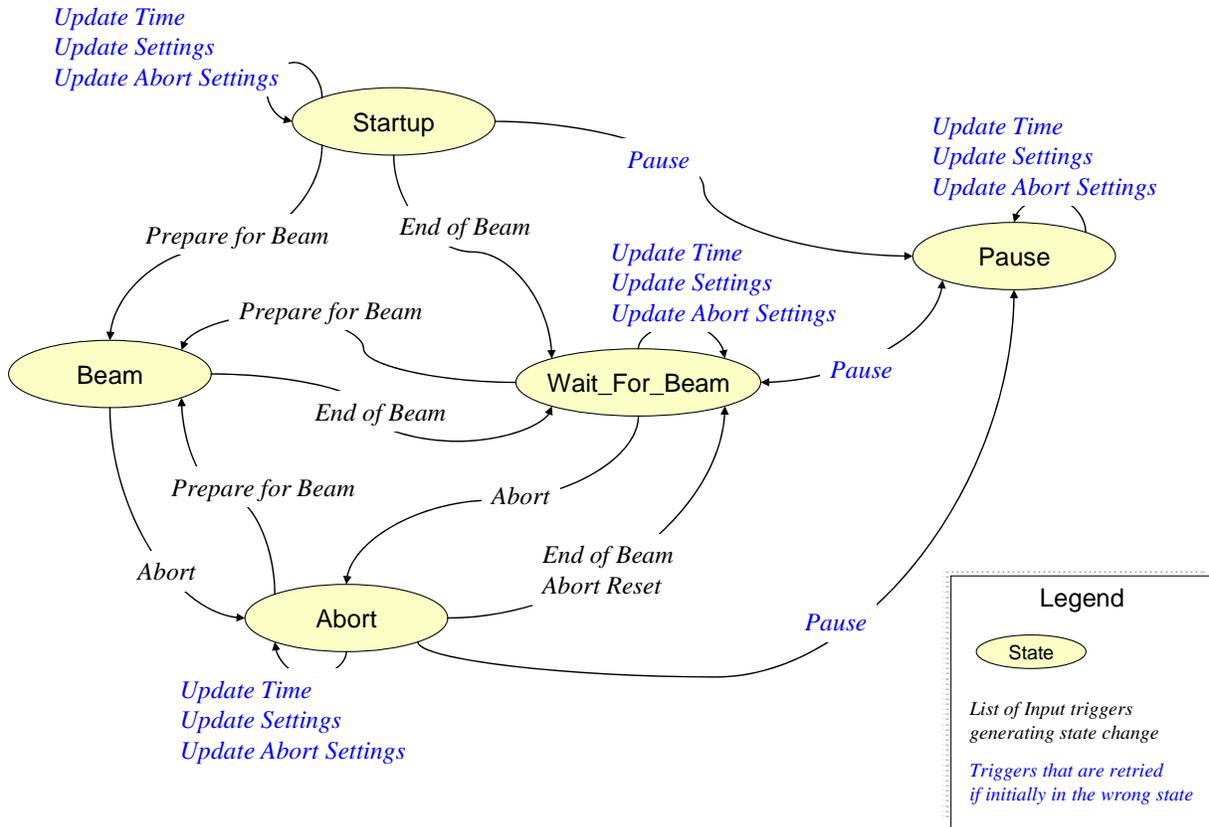
| <b>Bit</b> | <b>Description</b>                                      |
|------------|---|
| 15         | RESERVED  |
| 14         | RESERVED  |
| 13         | RESERVED  |
| 12         | Mismatched raw data pointers in TC, DC, or AC           |
| 11         | Pedestals Valid; Don't read pedestals until this is set |
| 10         | Very Slow Buffer has wrapped                            |
| 9          | Slow Buffer has wrapped                                 |
| 8          | Fast Buffer has wrapped                                 |
| 7          | Wrong number of Digitizer Card channels found           |
| 6          | Abort Card not found                                    |
| 5          | Timing Card not found                                   |
| 4          | Crate has triggered an Abort                            |
| 3          | Some channels are indicating abort                      |
| 2          | Some channels are not OK                                |
| 1          | CCP is in the process of initializing                   |
| 0          | The CC has rebooted; clearing this kicks off the CC     |

### **3.2 VME Accessible Circular Data Buffers**

The data from all the digitizers is stored in VME memory in the form of circular buffers. There are index counters which indicate which frame is the current frame, and flags in the status register to indicate when each buffer has wrapped around. Each data frame header contains a flag byte which indicates a number of things. At end of beam, the last frame contains a 1 in the flag byte. This is nominally to allow the CP to ignore this frame at the beginning of the next beam cycle. The first frame of a cycle has a 2 in the flag byte to allow the CP to correct for a bug in the system whereby the DC waits to start summing, but the TC sends out latches immediately. The result is that the slower sums are incomplete when the first latch is received. The CP can divide by the actual sum length if it knows a frame is the first one of the cycle. All other frames contain 0 in the flag byte.

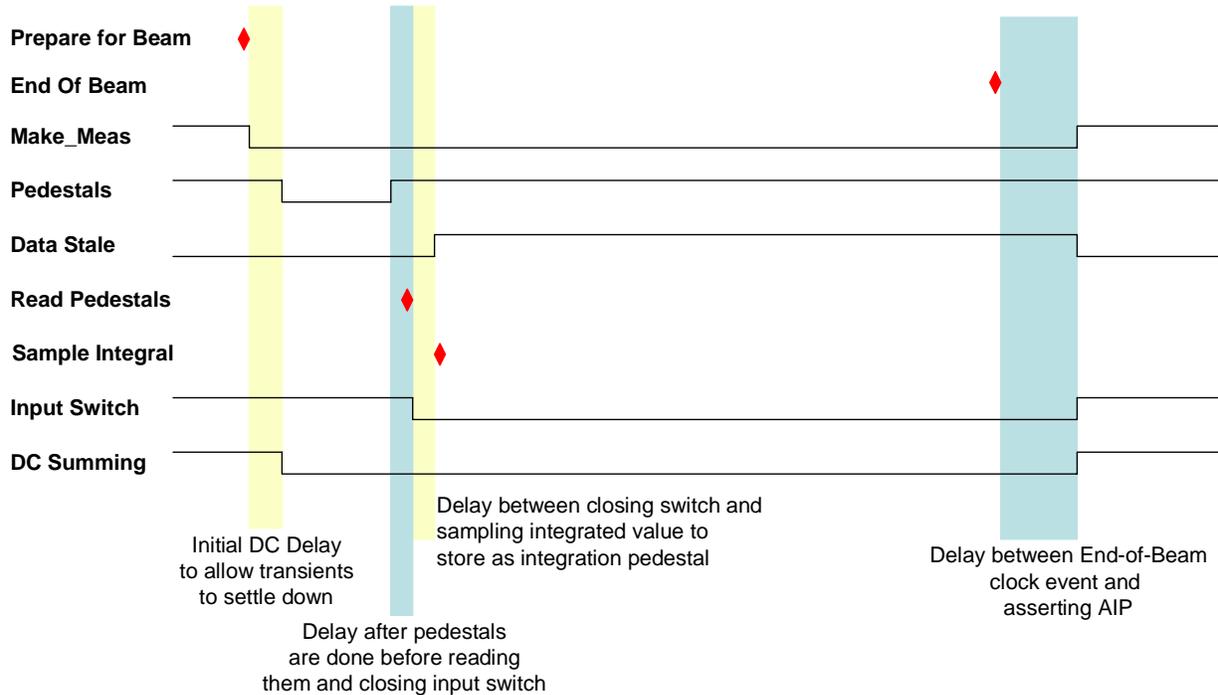
### 3.3 State Machine

Figure 2 illustrates the CCP state machine.



**Figure 2: State Machine diagram showing states and state change triggers. The triggers in blue are persistent triggers in that they hang around if they occur during an invalid state. When a valid state is reached, they are replayed.**

Figure 3 illustrates a typical beam cycle.



**Figure 3: Beam cycle showing the relative timing of various events. The shaded areas are delays between various key events. The integrated value must be sampled after closing the input switch, since there is a glitch upon closing the switch causing the integral to go negative by some amount. The lines are true low.**

### 3.4 Debugging Information

There is a section in VME accessible memory that contains possibly useful information for debugging and error trapping. There are counts of various quantities like the number of TCLKs, the number of each type of data latch, etc... This list is in Appendix A and in the BLM Users' Guide [1]. In addition, there is a CP trigger to dump Control Bus memory to a section of VME memory.

## 4 External Triggers

### 4.1 TCLK Event

The code responds to TCLK events in the form of polling the TC. A supported TCLK event causes the TC to put the TCLK number in the FIFO. The CCP polls the FIFO by checking the TCLK FIFO status register and reading the TCLK event. Table 3 documents the supported TCLK events.

**Table 3: List of CCP supported TCLK events, state inputs resulting from them, and a description of any action taken.**

| TCLK   | State Input             | Action   |
|--|-------------------------|--|
| <b>Tevatron</b>  |                         |  |
| \$77   | <i>None</i>             | Create a flash frame. The current fast sum and very slow sum buffer frames are appended to the flash frame buffer which contains a maximum of 256 flash frames and is cleared at Reset_DC. |
| \$75   | <i>None</i>             | Create a profile frame. The current fast sum and very slow sum buffer frames are appended to the profile frame buffer which contains a maximum of 256 profile frames                       |
| \$76<br>\$78   | <i>None</i>             | Create a display frame. The current fast sum and very slow sum buffer frames are placed in the display frame. Only one display frame is allowed at a time.                                 |
| \$71   | <i>Prepare for Beam</i> | Prepare for beam by issuing a Reset_DC, resetting the circular buffers, and clearing the abort in progress line.   |
| \$4B   | <i>End of Beam</i>      | End of beam. Tell TC to assert Abort In Progress line.   |
| \$47   | <i>Abort</i>            | Abort. Tell TC to assert Abort In Progress line.   |
| \$48   | <i>Abort Reset</i>      | Abort Reset.   |
| \$70   | <i>None</i>             | Reset linear buffers for flash and profile frames  |
| \$73   | <i>None</i>             | Until MDAT12 is functioning, this will trigger a machine state change to state 2.  |
| \$74   | <i>None</i>             | Until MDAT12 is functioning, this will trigger a machine state change to state 1.  |
| <b>Main Injector</b>   |                         |  |
| \$7C   | <i>None</i>             | Create a flash frame.  |
| \$7A   | <i>None</i>             | Create a profile frame.  |
| \$7B   | <i>None</i>             | Create a display frame.  |
| \$79<br>\$20<br>\$21<br>\$23<br>\$29<br>\$2A<br>\$2B<br>\$2D<br>\$2E | <i>Prepare for Beam</i> | Prepare for beam by issuing a Reset_DC, resetting the circular buffers, and clearing the abort in progress line.   |
| \$26   | <i>End of Beam</i>      | End of beam.   |
| \$27   | <i>Abort</i>            | Abort  |
| \$24   | <i>Abort Reset</i>      | Abort Reset  |

## 4.2 MDAT Frame

An MDAT frame forces a change of abort thresholds if the machine state has changed. The TC receives MDAT frames and places the state number in the FIFO. The CCP polls the FIFO status and reads the MDAT frame from the FIFO. It then switches to the proper thresholds page in the DC, copies the abort mask information corresponding to the MDAT machine state from VME memory to the AC, and tells the TC to generate an update abort settings signal. MDAT frame

\$12 contains the Tevatron state, and MDAT frame \$56 contains the Main Injector state. The state number received by the CC is in the range 0-255 with 0-127 being the TeV state, and 128-255 being 128 + the Main Injector state.

### **4.3 Data Latch Interrupt**

Periodically, at each sum period, a corresponding latch is generated by the TC by setting IRQ2 on the control bus. The CC is interrupted and the CCP reads the latch status registers in the TC to determine which sum data to read from the Digitizer Card. It then reads the data and stores it in the circular buffers in VME accessible memory. Finally it clears the corresponding status register which in turn clears IRQ2. Most of the actual memory copies from DC's to circular buffers are handled by assembly language instructions that are modularized in C language #defines. The assembly code makes use of specialized instructions to do multi-byte copies from one memory location to another, and is optimized to do most address manipulations in registers.

### **4.4 Abort Card Service**

The AC is periodically polled by the CCP and if the current status indicates over threshold, or abort, or channel not OK, the CCP status register is modified to reflect the state of the AC.

After an abort, the snapshot of the last frame of the abort card before the abort is copied to VME memory for access by the CP.

### **4.5 Crate Processor**

The CCP must poll various VME registers to respond to new data and/or requests from the front end. When the CP wants to initiate one of these events, it loads the proper settings and then writes a non-zero value to the appropriate register. The italicized entries are triggers that are located in the debug section of VME memory.

- **Update Time**  
The Time value must be copied to the TC and the TC must be told to update its time value.
- **Update Settings**  
The DC and TC global settings (e.g. Fast Sum Length) must be copied to the DC and TC boards. This must be done while the state machines are frozen. So in the main injector where beam cycles happen relatively quickly, it is done between beam cycles. In the Tevatron, it may on occasions be necessary to update immediately, instead of between stores. In that case, writing a 0xA596 to this register forces an immediate update.
- **Update Abort Settings**  
The DC thresholds, and the AC abort settings must be written to the DC and AC boards. Again, this must be done between beam cycles. Additionally, the TC must be told to assert the update settings line in order for the changes to take effect. By default, updates to the abort settings are done only between beam cycles. As with the regular settings, updates to the abort settings can be forced by writing a 0xA596 to this register.
- **Read Pedestals**  
Read the pedestals from the digitizer cards and place them in VME memory.
- **Clear Abort Information in Status Word**  
Tells the CCP to reset the bits in the Status Word that indicate the state of the AC.

- **Read / Write Flash Memory**  
Reading copies the flash memory to the VME memory section reserved for flash downloads/uploads. Writing copies the VME memory section to flash memory. Writing to flash memory involves a protocol and is more elaborate than reading. To initiate a write, the CP must write a 0xA596 to the register.
- **Read/Write Control Bus Memory**  
*Read copies 256 bytes of control bus memory to VME memory starting at the specified 16 bit address. The write copies the data byte to the specified 16 bit control bus address. To do this requires writing 0xA596 to the register to avoid accidentally doing this. These are used for debugging purposes and are not part of normal operation.*
- **Run User Function**  
*This trigger runs a user specified chunk of assembly code that was downloaded to VME memory. It gets copied to normal RAM and the entry point is called. For this to happen, the CP must write 0xA596 to the register (to avoid accidentally running something). This is a debugging feature and not normally used.*
- **Fake Prepare for Beam**  
*Initiates a prepare for beam state input. Exactly the same as if a prepare\_for\_beam clock event was received.*
- **Fake End of Beam**  
*Initiates an end of beam state input. Exactly the same as if an end\_of\_beam clock event was received.*
- **Pause the system**  
*When this is received, the system goes into a paused state as soon as it enters a non-beam state. So in the main injector, it would go here after the end of the current cycle (or immediately if currently between cycles). In this state, the control card effectively ignores clock events. This input acts like a toggle button, so successive triggers take the system in and out of the paused state.*

## 5 Output Control Lines

The control bus Reset State Machine signal can be used to quickly reset the system on the fly in the event that a card gets confused. A Reset is wired to port B pin 2 on the eZ80 which is setup as an output pin. This pin must be pulled low to assert the reset. Currently, the on-the-fly reset of the crate is not used. It would probably be initiated from the CP.

## 6 References

- [1] **BEAMS-DOC-1410** Beam Loss Monitor Upgrade Users' Guide. Documents hardware and functionality of the system.

## A Data Record

The fundamental unit stored in the various data buffers has the following format:

| Byte Offset | Length | Description  |
|-------------|--------|--|
| 0x00        | 1      | Machine state (0-255)  |
| 0x01        | 1      | Measurement Divisor  |
| 0x02        | 2      | Sum Divisor  |
| 0x04        | 1      | Abort Status (currently unused)  |
| 0x05        | 1      | Channel Count  |
| 0x06        | 1      | Data flag to indicate normal (0), last of cycle (1), new cycle (2), or waiting for stable data (3) frame |
| 0x07        | 1      | Unused   |
| 0x08        | 4      | Time stamp: Microseconds since last TCLK 1 Hz Event  |
| 0x0C        | 4      | Time stamp: seconds since T0 ( <i>i.e.</i> 1 January 1970)   |
| 0x10 + n*4  | 240    | Sliding Sum data for channel <i>n</i>  |

The data flag normally cycles through the following progression:

- at start of beam, wait for the digitizer delay, then write a 2 in the first frame collected (for each type of sum)
- write 3 in all the succeeding frames until after the input switch is closed for the specified amount of time
- write 0 in the rest of the frames of the cycle
- at end of beam, write a 1 in the last frame in the circular buffer

In the abort status word, bits 0-3 are the status of the abort from the Immediate, Fast, Slow, and Very Slow measurements, respectively. Bits 4-7 are not used. Loss data are stored as 32-bit long words with data in order from least significant to most significant byte.

Profile, Flash, and Display frames each contain 2 of these data frames. The first one is the user selectable Fast or Slow Sum frame and the second is the Very Slow Sum frame which in the Main Injector contains the integrated value.

## B VME Dual Port Memory Map

VME dual port memory addresses start at 0x09800000. The eZ80 addressing of the dual port memory begins at 0x800000 (24 bit addressing). The following addresses are specified as offsets from the base addresses.

| Address              | Size | Description   |
|----------------------|------|---|
| <b>System Status</b> |      |   |
| 000000               | 2    | Status Word   |
| 000006               | 1    | Bit pattern for selecting Flash/Profile/Display source<br>Bit 0 = flash, bit 1 = profile, bit 2 = display; 0 = fast, 1 = slow |
| 000008               | 1    | Read flash memory   |
| 00000A               | 2    | Load flash memory (requires 0xA596 to be written here)  |
| 00000C               | 2    | Flash memory load status  |

| <b>Address</b>                             | <b>Size</b> | <b>Description</b>  |
|--|-------------|---|
| 00000E                                     | 1           | Clear Abort Info Bits in Status Word  |
| 000010                                     | 1           | Force Pedestal Read   |
| 000012                                     | 1           | Update Time Setting   |
| 000014                                     | 2           | Time Setting LSW  |
| 000016                                     | 2           | Time Setting MSW  |
| 000018                                     | 1           | Update Settings   |
| 00001A                                     | 1           | Update Abort Settings   |
| 00001C                                     | 1           | Machine Type  |
| 00001E                                     | 1           | Machine State   |
| <b>Flash, Profile and Snapshot Indexes</b> |             |   |
| 000020                                     | 2           | Flash Frame Counter   |
| 000022                                     | 2           | Profile Frame Counter   |
| 000024                                     | 2           | Fast Sum Circular Buffer Frame Index 1  |
| 000026                                     | 2           | Fast Sum Circular Buffer Frame Index 2  |
| 000028                                     | 2           | Slow Sum Circular Buffer Frame Index 1  |
| 00002A                                     | 2           | Slow Sum Circular Buffer Frame Index 2  |
| 00002C                                     | 2           | Very Slow Sum Circular Buffer Frame Index 1   |
| 00002E                                     | 2           | Very Slow Sum Circular Buffer Frame Index 2   |
| 000030                                     | 1           | Fast Sum Frame Index Pointer  |
| 000032                                     | 1           | Slow Sum Frame Index Pointer  |
| 000034                                     | 1           | Very Slow Sum Frame Index Pointer   |
| 000036                                     | 2           | TC Raw data pointer   |
| 000038                                     | 2           | DC Raw data pointer   |
| 00003A                                     | 2           | AC Raw data pointer   |
| <b>Abort Info</b>                          |             |   |
| 000080                                     | 8           | Channel Not OK List from Abort Card   |
| <b>Settings</b>                            |             |   |
| 000100                                     | 1           | Channel Count   |
| 000102                                     | 2           | Make_Meas Divisor   |
| 000104                                     | 2           | Fast Sum Length   |
| 000106                                     | 2           | Slow Sum Length   |
| 000108                                     | 2           | Very Slow Sum Length  |
| 00010A                                     | 2           | Digitizer FPGA Control Register ( high byte is number of make_meas to skip before performing pedestal measurements / 16 ) |
| 00010C                                     | unused      | Digitizer Test DAC  |
| 00010E                                     | 1           | Timing Card Control Bus CSR Register  |
| 000110                                     | unused      | Timing Card Control Settings (reserved)   |
| 000112                                     | 1           | What generates an IRQ3 interrupt from the Abort Card?   |
| 000114                                     | 1           | Abort Card enabled  |
| 000116                                     | 2           | Pedestal Sum Length   |
| 000118                                     | 1           | End of beam delay before asserting AIP (in fast latch units)  |
| 00011A                                     | 1           | Delay after Flash clock event before collecting flash frame (in fast latch units)   |

| Address                             | Size | Description   |
|-------------------------------------|------|---|
| 00011C                              | 1    | Delay after Profile clock event before collecting profile frame (in fast latch units) |
| 00011E                              | 1    | Delay after Display clock event before collecting display frame (in fast latch units) |
| 000120                              | 1    | Turn off inputs while doing pedestals   |
| 000122                              | 1    | Optional extra delay after pedestals before re-enabling DC inputs                     |
| 000124                              | 1    | Delay after re-enabling DC inputs before reading data                                 |
| 000200                              | 2    | Channel 0 Mode Select   |
| 000202                              | 2    | Channel 0 MADC Manual Setting   |
| ↓                                   |      |   |
| 0002EC                              | 2    | Channel 59 Mode Select  |
| 0002EE                              | 2    | Channel 59 MADC Manual Setting  |
| <b>Pedestal Storage</b>             |      |   |
| 000300                              | 256  | 32-bit pedestals stored in standard data record                                       |
| <b>Squelch Storage</b>              |      |   |
| 000400                              | 120  | 16-bit squelch values   |
| <b>Abort Snapshot Storage</b>       |      |   |
| 000500                              | 32   | Abort snapshot record from last frame before abort                                    |
| 000520                              | 2    | Abort circular buffer pointer at time of snapshot                                     |
| <b>Integration Pedestal Storage</b> |      |   |
| 000600                              | 256  | 32-bit Integration pedestals  |
| <b>Debug Information</b>            |      |   |
| 010000                              | 48   | Program name and build date   |
| 010030                              | 4    | Test Sequence ( 0x44332211 )  |
| 010034                              | 4    | TCLK count  |
| 010038                              | 4    | MDAT frame count  |
| 01003C                              | 2    | Last TCLK received  |
| 01003E                              | 2    | Last MDAT state received  |
| 010040                              | 4    | Data Interrupt count  |
| 010044                              | 4    | Fast Sum Latch count  |
| 010048                              | 4    | Slow Sum Latch count  |
| 01004C                              | 4    | Very Slow Sum Latch count   |
| 010050                              | 4    | Profile request count   |
| 010054                              | 4    | Flash request count   |
| 010058                              | 4    | Display request count   |
| 01005C                              | 4    | Abort Interrupt count   |
| 010060                              | 4    | Channel not OK interrupt count  |
| 010064                              | 4    | Channel abort interrupt count   |
| 010068                              | 4    | Crate abort interrupt count   |
| 01006C                              | 4    | Missed Fast Latch count   |
| 010070                              | 4    | Missed Slow Latch count   |
| 010074                              | 4    | Missed Very Slow Latch count  |

| Address                      | Size | Description   |
|------------------------------|------|---|
| 010078                       | 2    | Program State   |
| 01007A                       | 2    | Number of fast data latches to wait until pedestals are done  |
| 01007C                       | 4    | Last Fast Latch time microseconds   |
| 010080                       | 4    | Last Fast Latch time seconds  |
| 010084                       | 4    | Last Slow Latch time microseconds   |
| 010088                       | 4    | Last Slow Latch time seconds  |
| 01008C                       | 4    | Last Very Slow Latch time microseconds  |
| 010090                       | 4    | Last Very Slow Latch time seconds   |
| 010094                       | 4    | Stack Pointer (Updated every time through polling loop)   |
| 010098                       | 4    | Stack Integrity (0xA4A3A2A1; also updated every loop)   |
| 01009C                       | 16   | Digitizer Map (addresses of found digitizers; 16 <sup>th</sup> byte is 0)   |
| 0100AC                       | 4    | Timing Card clock error count   |
| 0100B0                       | 4    | Count of times TCLK queue has more than 1 entry   |
| 0100B4                       | 4    | Count of times MDAT queue has more than 1 entry   |
| 0100B8                       | 4    | Number of times a state $\geq 64$ was received  |
| 0100BC                       | 2    | Last state machine input  |
| 0100BE                       | 2    | Force prepare for beam event (requires 0xA596 to be written here)   |
| 0100C0                       | 2    | Force end of beam event (requires 0xA596 to be written here)  |
| 0100C2                       | 2    | Pause the system (requires 0xA596 to be written here). This happens now if waiting for beam, or after next end of beam event. |
| 010100                       | 1024 | TCLK counts for each TCLK type (256 types)  |
| 010500                       | 1024 | MDAT counts for each MDAT state (256 states)  |
| 018000                       | 2    | Request a Control Bus memory dump   |
| 018002                       | 2    | Request a Control Bus write (requires 0xA596 to be written here)  |
| 018004                       | 2    | Control Bus address to dump / write (dumps 256 bytes / write 1 byte)  |
| 018100                       | 256  | Data copied from / written to Control Bus (only 1 byte in case of write)  |
| 019000                       | 2    | Request User Code to be run (requires 0xA596 to be written here)  |
| 01A000                       | 8192 | Buffer for User Code to be placed   |
| <b>Flash Memory Download</b> |      |   |
| 020000                       | 128K | Buffer for new flash memory contents  |
| <b>Flash Frames</b>          |      |   |
| 080000                       | 512  | Flash Frame 0   |
| 080200                       | 512  | Flash Frame 1   |
| ↓                            |      |   |
| 09FE00                       | 512  | Flash Frame 255   |
| <b>Profile Frames</b>        |      |   |
| 0A0000                       | 512  | Profile Frame 0   |
| 0A0200                       | 512  | Profile Frame 1   |
| ↓                            |      |   |
| 0BFE00                       | 512  | Profile Frame 255   |

| Address                     | Size | Description                                     |
|-----------------------------|------|---|
| <b>Display Frame</b>        |      |   |
| 0C0000                      | 512  | Display Frame                                   |
| <b>Abort Machine States</b> |      |   |
| <b>State 0</b>              |      |   |
| 100000                      | 1    | State Number                                    |
| 100002                      | 2    | Channel Masks 0 1 for Immediate Abort           |
| 100004                      | 2    | Channel Masks 2 3 for Immediate Abort           |
| 100006                      | 2    | Channel Masks 4 5 for Immediate Abort           |
| 100008                      | 2    | Channel Masks 6 7 for Immediate Abort           |
| 10000A                      | 2    | Channel Masks 0 1 for Fast Abort                |
| 10000C                      | 2    | Channel Masks 2 3 for Fast Abort                |
| 10000E                      | 2    | Channel Masks 4 5 for Fast Abort                |
| 100010                      | 2    | Channel Masks 6 7 for Fast Abort                |
| 100012                      | 2    | Channel Masks 0 1 for Slow Abort                |
| 100014                      | 2    | Channel Masks 2 3 for Slow Abort                |
| 100016                      | 2    | Channel Masks 4 5 for Slow Abort                |
| 100018                      | 2    | Channel Masks 6 7 for Slow Abort                |
| 10001A                      | 2    | Channel Masks 0 1 for Very Slow Abort           |
| 10001C                      | 2    | Channel Masks 2 3 for Very Slow Abort           |
| 10001E                      | 2    | Channel Masks 4 5 for Very Slow Abort           |
| 100020                      | 2    | Channel Masks 6 7 for Very Slow Abort           |
| 100022                      | 2    | Abort Multiplicity for Immediate and Fast Abort |
| 100024                      | 2    | Abort Multiplicity for Slow and Very Slow Abort |
| 100026                      | 2    | Crate Abort Mask                                |
| 100028                      | 8    | Unused  |
| 100030                      | 2    | Channel 0 Immediate Threshold                   |
| 100032                      | 2    | Channel 1 Immediate Threshold                   |
| ↓                           |      |   |
| 1000A6                      | 2    | Channel 59 Immediate Threshold                  |
| 1000A8                      | 8    | Unused  |
| 1000B0                      | 2    | Channel 0 Fast Threshold LSW                    |
| 1000B2                      | 2    | Channel 0 Fast Threshold MSW                    |
| 1000B4                      | 2    | Channel 1 Fast Threshold LSW                    |
| 1000B6                      | 2    | Channel 1 Fast Threshold MSW                    |
| ↓                           |      |   |
| 10019C                      | 2    | Channel 59 Fast Threshold LSW                   |
| 10019E                      | 2    | Channel 59 Fast Threshold MSW                   |
| 1001A0                      | 16   | Unused  |
| 1001B0                      | 2    | Channel 0 Slow Threshold LSW                    |
| 1001B2                      | 2    | Channel 0 Slow Threshold MSW                    |
| 1001B4                      | 2    | Channel 1 Slow Threshold LSW                    |

| Address  | Size  | Description   |
|--|-------|---|
| 1001B6   | 2     | Channel 1 Slow Threshold MSW  |
| ↓  |       |   |
| 10029C   | 2     | Channel 59 Slow Threshold LSW   |
| 10029E   | 2     | Channel 59 Slow Threshold MSW   |
| 1002A0   | 16    | Unused  |
| 1002B0   | 2     | Channel 0 Very Slow Threshold LSW   |
| 1002B2   | 2     | Channel 0 Very Slow Threshold MSW   |
| 1002B4   | 2     | Channel 1 Very Slow Threshold LSW   |
| 1002B6   | 2     | Channel 1 Very Slow Threshold MSW   |
| ↓  |       |   |
| 10039C   | 2     | Channel 59 Very Slow Threshold LSW  |
| 10039E   | 2     | Channel 59 Very Slow Threshold MSW  |
| 1003A0   | 80    | Unused  |
| <b>States 1 – 255; each with same offsets as State 0</b> |       |   |
| 100400   | 1024  | State 1 Settings  |
| 100800   | 1024  | State 2 Settings  |
| ↓  |       |   |
| 13F800   | 1024  | State 255 Settings  |
| <b>In Use Abort Settings for States 0-255</b>            |       |   |
| 140000   | 256 K | Copy of settings that are in use. These are the settings that are copied to AC and DCs. |
| <b>Circular Buffers</b>                                  |       |   |
| 200000   | 4 Meg | Fast Sum Buffer 16K frames deep   |
| 600000   | 1 Meg | Slow Sum Buffer 4K frames deep  |
| 700000   | 1 Meg | Very Slow Sum Buffer 4K frames deep   |