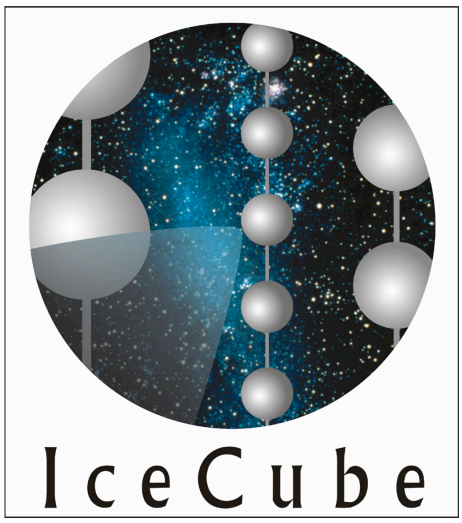
IceCube

DOMAPP CPU FPGA Interface

Version 1.11



Thorsten Stezelberger

TStezelberger@lbl.gov

Lawrence Berkeley National Laboratory

October 17, 2006

Contents

1 Introduction

The DOM Main Board contains an Altera Excalibur chip. This chip combines an ARM 9 CPU and an Altera APEX20KE FPGA. The firmware and software in this chip control the data taking, communication, calibration and the monitoring of the DOM Main Board.

The purpose of this document is to describe the interface between the FPGA part and the CPU part in the Excalibur.

The interface definitions in this document are based on the baseline configuration. The baseline configuration is the EPXA4 member of the Altera Excalibur family and two 16MB Micron SDRAM chips.

1.1 Altera Excalibur Overview

This section is intended to give an overview of the available interfaces between the CPU and the FPGA in the EPXA4. See figure 1 for a block diagram of the interfaces in the EPXA4.

The STRIPE TO PLD bridge enables the CPU to access slave resources in the FPGA. This is used to implement a register interface in the FPGA. The PLD TO STRIPE bridge allows the FPGA to use resources in the CPU section like the SDRAM. This allows for a DMA data transfer into the SDRAM of the CPU. The Dual Ported Memory can be accessed from both, the CPU and the FPGA at the same time. Its size makes it ideal for the communications FIFO. The interrupt interface allows to send interrupts requests to the CPU.

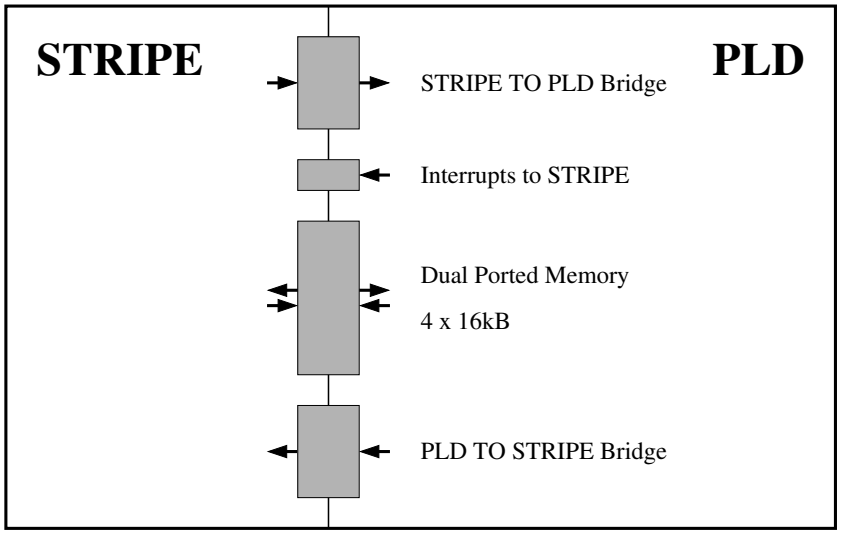


Figure 1: EPXA4 Block Diagram

2 Registers

The registers in the FPGA use the STRIPE TO PLD bridge.

The registers are 32bit wide and respond to 32bit accesses only. All bits in the registers are high active (1=enabled; 0=disabled).

2.1 Register overview

Table 1 gives a list of the registers in the FPGA and their addresses. See the following sections for a detailed description of these registers.

Addr (hex)	Mode	Register	Initial Value
90000000-900001FC	R	Version information	Version
90000400	W	Trigger Source	00000000
90000404	W	Trigger setup	00000000
90000410	W	DAQ	
90000420	W	LBM control	
90000424	R	LBM pointer	NA
90000430	R	DOM status	NA
90000440	R	Systime LSB	NA
90000444	R	Systime MSB	NA
90000450	W	Local Coincidence Control	
90000454	W	LC Cable Length Up	00000000
90000458	W	LC Cable Length Down	00000000
90000460	W	Calibration Source Control	
90000464	W	Calibration Time	00000000
90000468	W	Calibration CPU launch	NA
9000046C	R	Last Calibration Flash Time LSB	NA
90000470	R	Last Calibration Flash Time MSB	NA
90000480	W	Rate Monitor control	
90000484	R	Rate Monitor SPE	NA
90000488	R	Rate Monitor MPE	NA
900004A0	W	Supernova Meter Control	
900004A4	R	Supernova Data	NA
900004C0	W	Interrupt Enable	00000000
900004C4	RW	Interrupt ACK	

continued on next page

continued from previous page

Addr (hex)	Mode	Register	Initial Value
900004E0	W	Flasher Board Control	
900004E4	R	Flasher Board Status	NA
90000500	W	Communication control	
90000504	R	Communication status	NA
90000508	W	Communication (tx_head) tx_dpr_wadr	NA
9000050C	R	Communication (tx_tail) tx_dpr_radr	NA
90000510	W	Communication (rx_tail) rx_dpr_radr	NA
90000514	R	Communication (rx_head) rx_addr	NA
90000518	R	Communication packets in RX buffer	
9000051C	R	Communication error counters	NA
90000520	W	Communication Parameter Levels	NA
90000524	W	Communication Parameter Thresholds and Delays	NA
90000530	W	DOM ID 32 LSB	00000000
90000534	W	DOM ID 16 MSB	00000000
90000540	W	Compression Control	
90000560	W	IceTop Control	
900007F8		PONG	
900007FC		Firmware Debugging	
90000800-90000BFC	R	Supernova Buffer	NA
90000C00-90000FFC	W	R2R Ladder Pattern	ALL 0
90001000-900017FC	W	ATWD A Pedestal Pattern	ALL 0
90001800-90001FFC	W	ATWD B Pedestal Pattern	ALL 0

Table 1: Registers (R=read; W=write; NA=not applicable)

2.2 Register description

2.2.1 Version Information

The Version Information is 16bit wide but is in a 32bit word. The upper 16 bit are not defined. For information about the available features available for a giver version number see ..... .

Ask Arthur for a reference or see CVS dom-fpga/scripts !

Addr (hex)	Function
90000000	FPGA type
4	Build number LSB
8	Build number MSB
C	Communication FIFO
10	Communication DPM
14	DAQ
18	Calibration Sources
1C	Discriminator Rate Meter
20	Local Coincidence
24	Flasher Board Interface
28	Trigger
2C	System Clock
30	Supernova
....	
190	Communications Code Version
....	

Table 2: Version Information

2.2.2 Trigger Source

Each of the bits in the "Trigger Source" register in table 3 if enabled can trigger the ATWD. All the trigger sources are ORed together.

Only one of the two discriminators, SPE and MPE, can be activated at one time. If both, the SPE and the MPE discriminator, are enabled only the SPE discriminator can trigger! This limitation is introduced to ensure the fastest possible trigger time in the FPGA.

Bit	Function
0	SPE discriminator
1	MPE discriminator
2	CPU forced
3	Frontend pulser
4	On board LED
5	Flasher board
6	Frontend R2R
7	ATWD R2R
8	Local coincidence UP
9	Local coincidence DOWN

Table 3: Trigger Source register

2.2.3 Trigger setup

The LBM Control register is for future use. Currently no function is assigned to it. Do not use. I could be used for things like artificial dead time.

2.2.4 DAQ

The "DAQ" register (see Table 4) defines the modes of data acquisition. It doesn't hold the specific information for the modes, it just selects the modes.

Bit	Function
0	Enable Data Taking
1	Enable ATWD A
2	Enable ATWD B
10 - 8	DAQ Mode
14 - 12	ATWD Mode
18 - 16	LC Mode
19	Disable LC Heart Beat Mode
22 - 20	LBM Mode
26 - 24	Compression Mode
28	Enable IceTop Mode

Table 4: DAQ register

LC Heart Beat Mode: Heart Beat Mode doesn't require local coincidence for the calibration triggers ("Dark", Frontend PMT-like Pulser, Onboard LED, Flasher Board, Frontend R2R$\tilde{\text{L}}$adder and ATWD R2R$\tilde{\text{L}}$adder). If the Heart Beat Mode is disabled, the above mentioned trigger require LC if the DAQ LC Mode is other than "OFF".

Sequence to avoid "old" data in the LBM in stop when full mode. When the LBM is full the FPGA internal buffers fill up too. To clear the FPGA internal buffers they have to get flushed into the LBM. To do this disable data taking, then reset the LBM pointer, wait at least $500\mu s$, then reset the LBM pointer again.

Value	Mode
0	ATWD & FADC
1	FADC only
2	Timestamp only
3	TBD

Table 5: DAQ Modes

All Modes need at least one ATWD enabled, otherwise no data is taken. This is because the data flow through the FPGA is linked to the ATWDs, even if no ATWD data is taken. The DAQ Mode "Timestamp only" does not

use the local coincidence system due to the short dead time. Also, "Timestamp only" Mode does not generate a valid Chargestamp.

Mode	Value	Description
Normal	0	Start at channel 0. If there is an overflow at channel 0 go to channel 1. If there is an overflow .....
Testing	1	digitize all 4 channels
Debugging	2	Based on debugging needs
TBD	3	TBD

Table 6: ATWD Modes

The Local Coincidence Modes (see Table 7) are OFF, SOFT and HARD. In HARD mode all events without an incoming local coincidence signal will not get recorded. In SOFT mode only the timestamp will get recorded. In OFF mode all events get recorded.

Value	Mode
0	OFF
1	SOFT
2	HARD
3	FLABBY

Table 7: Local Coincidence Modes

In order to avoid problems, like corrupt data, the LBM pointer must only get reset when data taking os off. The LBM pointer does not reset automatically when data taking gets disabled.

Value	Mode
0	wrap endless
1	stop when buffer full
2	TBD
3	TBD

Table 8: LBM Modes

The "Compression Mode" (see Table 9) selects what data gets stored in the look back memory. Compression "OFF" stores the raw events. Compression "ON" stores the compressed events. Compression "BOTH" creates two data records one for raw data and one for compressed data.

Value	Mode
0	OFF
1	ON
2	BOTH
3	TBD

Table 9: Compression Modes

2.2.5 LBM Control

Bit	Function
0	Reset LBM Pointer

Table 10: LBM Control

A reset of the LBM pointer takes effect for the next 2k data block written to lock back memory. A LBM pointer reset should only occur when data taking is off.

2.2.6 LBM Pointer

The LBM pointer is a byte pointer and points to the next address the DOMAPP firmware will write to. With its 32 bits the LBM pointer is larger than necessary given the available memory size. The extra bits are intended to allow the software to detect if pointer wrapped around.

2.2.7 DOM Status

The DOM Status register (see Table 11) holds monitor and debug information like ATWD A/B acquires, digitizes, readout, is busy, ... TBD.

Bit	Function
0	A busy
1	A FADC busy
2	A Buffer full
8	B busy
9	B FADC busy
10	B Buffer full
16	AHB_Master bus_error
17	AHB_Master slavebuserrint
18	AHB_Master bus_error Latched
19	AHB_Master slavebuserrint Latched
20	Transferring Engineering Data to LBM
21	Transferring Compressed Data to LBM
24	SPE discriminator
25	MPE discriminator
29	toggle bit 5MHz based on 20MHz clock
30	toggle bit 5MHz based on 40MHz clock
31	toggle bit 5MHz based on 80MHz clock

Table 11: DOM Status register

2.2.8 Systime

The Systime register allows the CPU to read the system time. The system time is used in the FPGA for timestamping. The system time has $25ns$ resolution. The system time starts at 0 at power up.

Due to the length of the system time it is necessary to split it into two registers. The Timestamp LSB register holds the lowest 32 bit and the Timestamp MSB register holds the highest 16 bit.

2.2.9 Local Coincidence Control

The Local Coincidence Control register (see Table 12) holds the setup information for the LC like nearest neighbor or far neighbor. Send or not send LC.

For the purpose of Local Coincidence, up and down refer to the labelling on the DOM Main Board.

Bits	Function
0	Enable sending LC up
1	Enable sending LC down
2	Enable receiving LC up
3	Enable receiving LC down
5 - 4	LC length (span across DOMs)
6	Require LC from above and below
7	Discriminator source (0=SPE; 1=MPE)
9 - 8	Self Local Coincidence Mode
15 - 10	Self Local Coincidence Window
21 - 16	Pre LC Window
29 - 24	Post LC Window

Table 12: Local Coincidence Control

The Pre/Post Discriminator LC Window range from $25ns$ to $1.6\mu s$ in $25ns$ increments.

"Self Local Coincidence" gives the possibility to overwrite the normal local coincidence behavior. With "Self Local Coincidence" a DOM can keep certain hits even at the absence of a local coincidence with its neighbors.

"Self Local Coincidence" works as follows; if a high amplitude signal is captured, one wants to store the signal no matter if it is accompanied by an local coincidence.

Self Local Coincidence is done using the discriminators. The "Self Local Coincidence Mode" specifies which discriminator is used to detect the high amplitude signal. The "Self Local Coincidence Window" specifies the time frame after a ADC launch within a high amplitude signal must get detected to satisfy Self Local Coincidence. The "Self Local Coincidence Window" is given in $25ns$ clock cycles. *The "Self Local Coincidence Window" must not longer than the "Post LC Window" plus the longest "LC cable Length" for*

the set "LC length". Otherwise the Self Local Coincidence behavior is not defined.

Value	Function
0	OFF
1	SPE Discriminator
2	MPE Discriminator
3	TBD

Table 13: Self Local Coincidence Mode

2.2.10 LC Cable Length Up

Bits	Function
6 - 0	Neighbor Distance 0
14 - 8	Neighbor Distance 1
22 - 16	Neighbor Distance 2
30 - 24	Neighbor Distance 3

Table 14: LC Cable Length Up

Distance is given in 25ns units. The distance includes electronics delays!

2.2.11 LC Cable Length Down

Bits	Function
6 - 0	Neighbor Distance 0
14 - 8	Neighbor Distance 1
22 - 16	Neighbor Distance 2
30 - 24	Neighbor Distance 3

Table 15: LC Cable Length Down

2.2.12 Calibration Source Control

The "Calibration Source Control" has the top level control of all the calibration sources in the DOM.

The DOM main Board has two calibration sources with arbitrary waveform generators. The arbitrary waveform generators use R2R ladders. The waveform generators are the Frontend R2R Ladder and the ATWD R2R Ladder. The Frontend R2R Ladder is connected the analog front-end after the delay line. The ATWD R2R Ladder is connected to the multiplexer at channel 3 on the ATWDs. The arbitrary waveform for the two R2R ladders is stored in the R2R Ladder Pattern in section 2.2.33. When the R2R ladders are used, all 256 entries get send with a rate of 40MSamples.

Bit	Function
0	"Dark"
1	Frontend PMT-like pulser
2	Onboard LED
3	Flasher Board
4	Frontend R2R Ladder
5	ATWD R2R Ladder
14 - 12	Calibration Mode
19 - 16	ATWD launch offset
28 - 24	Pulser rate

Table 16: Calibration Source Control

The "Dark" mode allows to generate a launch signal for the ATWDs without activating one of the pulser sources. This mode can get used to acquire pedestal waveforms.

More than one Calibration Source can get turned on at a given time. In this case, they "flash" at the same time.

The pulser rate is given in equation 1. One must not set the "pulser rate" higher than 17. Functionality is not guaranteed beyond this point.

$$\begin{aligned} Rate &= \frac{1}{25ns\ 2^{26}} 2^{PulserRate} \\ PulserRate &= 0 \cdots 17 \end{aligned} \tag{1}$$

$$OR$$

$$Rate = low - highedge\,(systimebit\,(26 - PulserRate))$$

The "Calibration Mode" (Table 17) selects when the calibration Sources should get used. The setting "OFF" doesn't use the calibration Sources at all. The setting "Repeating" flashes the calibration sources at the low-high edge of the to the rate corresponding bit in the systime. The setting "Time Match" flashes the calibration sources once the next time the time in the "Calibration Time" register matches the systime. The setting "CPU Forced" flashes the calibration sources when the CPU writes the correct value into the "Calibration CPU Launch" register.

Value	Function
0	OFF
1	Repeating
2	Time Match
3	CPU Forced

Table 17: Calibration Mode

The ATWD launch offset is the time between triggering the Calibration Source and forced triggering an ATWD. The offset is given as 4 bit twos complement from -8 to $+7$ $25ns$ clock cycles. See equation 2.

The "flash" of the calibration sources is fixed relative to the set "flash" time. The ATWD launch moves relative to the calibration source "flash" as set by the "ATWD Launch Offset".

$$ATWD\ Launch\ Offset = -(t_{trigger\ calibration\ source} - t_{trigger\ ATWD}) \qquad (2)$$

2.2.13 Calibration Time

The "Calibration Time" register holds the 32 bit time when the Calibration Sources should flash the next time. The "Calibration Time" is given as the lower 32 bit of the "Systime". The upper 16 bit are don't care.

In order to use the "Calibration Time" feature the calibration sources have to get "armed" after time is set. The "arming" done by setting the "Calibration Mode" to OFF and then to "Time Match".

2.2.14 Calibration CPU Launch

The "Calibration CPU Launch" register is used to initiate a CPU forced flash of the calibration sources. The CPU must write $A5_{hex}$ into the low 8 bit in order to flash the calibration sources.

2.2.15 Last Calibration Flash Time

The "Last Calibration Flash Time" register holds the system timestamp of the last flash. The LSB register contains the lower 32 bit of the timesatmp and the MSB register contains the upper 16 bit of the timestamp. *The actual "flash" happened 2 clock cycle later.*

2.2.16 Rate Monitor Control

The Rate Monitor Control register (see Table 18) has the setup information for the rate monitors. the rate monitors count discriminator hit with a gate time of one second. There are two rate monitors, one for the SPE and one for the MPE discriminator. The settings are the same for the SPE and the MPE rate meter. The Artificial Discriminator deadtime ranges from $100ns$ to $102.4\mu s$ is $100ns$ increments.

Bits	Function
0	Enable SPE
1	Enable MPE
25 - 16	Artificial Discriminator deadtime

Table 18: Rate Monitor Control

2.2.17 Rate Monitors

The Rate Monitors hold in the low 16 bits the discriminator hits counter in the last gate time window of 1s. The unused bits are 0. There are two registers, one for the SPE and one for the MPE discriminator.

2.2.18 Supernova Meter Control

The Supernova Meter counts continuously discriminator hits within the given "gate Time" and puts the results into the "Supernova Data" register (Sec-

tion 2.2.19. The counter employs a set able "Dead Time" after each counted discriminator hit. The Supernova Meter can count either the SPE or the MPE discriminator (see table 20).

Bits	Function
1- 0	Enable
22 - 16	Dead Time

Table 19: Supernova Meter Control register

The Dead Time ranges from $6.4\mu s$ to $819.2\mu s$ in $6.4\mu s$ increments. The Dead Time is calculated as follows.

Dead Time = ("Dead Time Register" + 1) $*6.4\mu s$

The Gate Time is fixed to $1.6384ms = \frac{1}{40MHz}2^{16}$.

Value	Function
0	OFF
1	SPE
2	MPE
3	TBD

Table 20: Supernova Meter Enable

2.2.19 Supernova Data

Bits	Function
3 - 0	Data time slot 0
7 - 4	Data time slot 1
11 - 8	Data time slot 2
15 - 12	Data time slot 3
31 - 16	Timestamp 31 - 16

Table 21: Supernova Data register

The 16 bit timestamp shows the systime bits 31 - 16 at the end of the gate time for "Data time slot 3". At then end of "Data time slot 0" the systime bits 15 - 0 are always 0001_{hex}.

2.2.20 Interrupt Enable

The "Interrupt Enable" register is used to enable the interrupt sources in the FPGA. For a list of the available interrupt sources see table 32. Disabling an active interrupt will also clear it.

2.2.21 Interrupt ACK

Reading the "Interrupt ACK" register will show the pending interrupts. Writing the "Interrupt ACK" register will clear the selected interrupts. Writes to the "Interrupt ACK" register are self clearing. For a list of the available interrupt sources see table 32.

2.2.22 Flasher Board Control

This signal is directly mapped to the corresponding signal of the flasher board interface.

The DOMAPP FPGA does not support the flasher board JTAG interface.

Bits	Function
0	AUX_RESET / PRE_TRIG

Table 22: Flasher Board Control Register

2.2.23 Flasher Board Status

This signal is directly mapped to the corresponding signal of the flasher board interface.

The DOMAPP FPGA does not support the flasher board JTAG interface.

Bits	Function
0	ATTN

Table 23: Flasher Board Status Register

2.2.24 Communication

This section needs input from Kalle. But at lest a Control and a Status register is required.

For the time being we need a TX and RX FIFO until Communications is mature enough for DPM.

Bits	Function
0	Reboot request

Table 24: Communication Control Register

Bits	Function
0	Reboot granted
1	Packet sent
2	Transmit buffer almost empty
3	Packet received
4	Communication Reset received
5	Receive buffer Almost full
6	Communication available (DOR-DOM connection established)

Table 25: Communication Status Register

TX and RX data is stored in a ringbuffer. Where ptr read, write ...
Transmit write address:
transmit read address:
receive read address:
receive address:

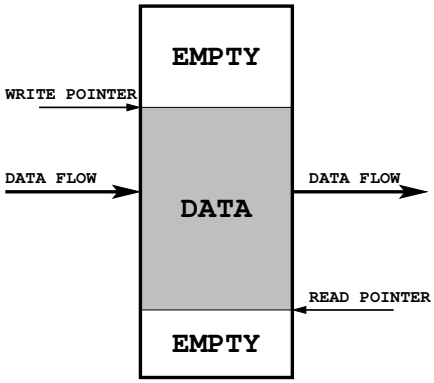


Figure 2: Ringbuffer

Writing TX_HEAD indicates ONE complete new message is in the TX buffer!! Used as indication, "message ready for transmit".

Writing RX_TAIL indicates ONE complete message got read from the RX buffer!! Used to calculate the "packets in RX buffer" counter.

2.2.25 Communication packets in RX buffer

2.2.26 Communication error counters

2.2.27 Communication Parameter Levels

Bits	Function
9 - 0	Minimal Good Receive Level
25 - 16	Maximal Good Receive Level

Table 26: Communication Parameter Levels register

2.2.28 Communication Parameter Thresholds and Delays

Bits	Function
7 - 0	Communication Threshold
9 - 8	DAC max
23 - 16	Receive Delay
31 - 24	Transmit Delay

Table 27: Communication Parameter Thresholds and Delays register

2.2.29 DOM ID

The DOM ID is a 48 bit number to identify the DOMMB by the communications firmware.

2.2.30 Compression Control

This register holds information necessary to control compression. The register is specific to the "1-2-3-6-11 Delta Compression" which is implemented in the FPGA

Bits	Function
0	Only compress last ATWD Channel
	...

Table 28: Compression Control register

2.2.31 IceTop Control

This Register controls IceTop specific functionality.

Bits	Function
1 - 0	IceTop ATWD Charge; Channel select

Table 29: IceTop Control Register

2.2.32 Firmware Debugging

The Firmware Debugging register is used for development only. Do not use.

2.2.33 R2R Ladder Pattern

The R2R Ladder pattern holds the waveform to be send to the frontend R2R and the ATWD R2R ladder. The waveform is used for both ladders. The memory is 8 bit wide and 256 entries deep. Only the low 8 bit of the 32 bit word are mapped to the R2R Ladder Pattern memory.

Pattern Bit	R2R Bus Bit (ATWD R2R)	FE Pulser (frontend R2R)
0	0	FE_pulser_N(0)
1	1	FE_pulser_N(1)
2	2	FE_pulser_N(2)
3	3	FE_pulser_N(3)
4	4	FE_pulser_P(0)
5	5	FE_pulser_P(1)
6	6	FE_pulser_P(2)
7	7	FE_pulser_P(3)

Table 30: R2R Ladder Pattern Mapping

2.2.34 ATWD Pedestal Pattern

The ATWD Pedestal Pattern location holds the pedestal information for the given ATWD for pedestal subtraction. The software has to write the pedestal pattern. In the case pedestal substitution is not desired the pedestal pattern must be filled with zeros. The pedestal pattern uses 10 bit signed numbers. The 10 bits of the pattern are the low 10 bits of the 32 bit word. There is one pedestal pattern for each channel of the ATWD. The pattern are in the order from channel 0 to channel 3. The lowest address holds the last sample. The pattern is in the order the data comes out of the ATWD. See Table 31. For ATWD B add 800_{hex} to the address.

The following pseudo code illustrates the pedestal subtraction.

```
/* for each sample */
```

```
/* raw and subtracted are unsigned */
/* pedestal and temp are signed */

temp = raw[index] - pedestal[index];
if (temp <= 0) then
{
    subtracted[sample] = 0;
}
else
{
    if (temp >= 1023) then
    {
        subtracted[sample] = 1023;
    }
    else
    {
        subtracted[sample] = temp;
    }
}
```

Addr	Entry		
90001000	ATWD A	Channel 0	Pedestal Sample 127
90001004	ATWD A	Channel 0	Pedestal sample 126
⋮		⋮	
900011FC	ATWD A	Channel 0	Pedestal sample 0
90001200	ATWD A	Channel 1	Pedestal Sample 127
⋮		⋮	
900017FC	ATWD A	Channel 3	Pedestal sample 0

Table 31: ATWD Pedestal Pattern

3 Dual Ported Memory

CHANGED! We use 32 bit now for TX and RX. Only the low 8 are used. On the CPU side the address increases by 4. For more information about how

to access DPM from the CPU see the Altera Excalibur Hardware Reference Manual and the Altera Application Note 173 DPRAM Reference Design. The base addresses for the DPM are 80000000_{hex} and 80010000_{hex}.

3.1 Communication Transmit Buffer

32kB buffer 32 bit wide at 80000000_{hex}. Interface and use TBD.

3.2 Communication Receive Buffer

32kB buffer 32 bit wide at 80000000_{hex} + 32K (80010000_{hex}). Interface and use TBD.

4 Interrupts

The Excalibur chip offers 6 interrupts to the CPU. The defined interrupts are listed in table 32.

Interrupt	Function
0	Calibration Source Flashed
1	Rate Meter Update
2	Supernova Data Updated
3	TBD
4	TBD
5	TBD

Table 32: Interrupt Sources

5 Look Back Memory

The FPGA writes the event information into the Look Back Memory section of the SDRAM through the PLD to STRIPE bridge. The 8MB SDRAM space reserved as Lock Back Memory is from 01000000_{hex} to $01FFFFFF_{hex}$. The SDRAM access from the FPGA is 32 bit only.

If it is selected to save the compressed and the full data there will be two events in the LBM. One for the raw event an done for the compressed event.

Events are aligned to the to 2 kbyte boundaries in the Look Back Memory.

5.1 Raw Data Format

The Raw Data is divided into three sections. The three sections are Header data, FADC data and ATWD data.

5.1.1 Raw Data Format Header

The Header starts at offset address $+0_{hex}$.

LWORD	Bits	Function
0	15 - 0	Timestamp Bits 15 - 0
0	31 - 16	Constant 0001_{hex}
1	31 - 0	Timestamp Bits 47 - 16
2	15 - 0	Trigger Source
2	16	ATWD A/B (A=0; B=1)
2	17	FADC available
2	18	ATWD available
2	20 - 19	ATWD size
2	24	Got LC from down
2	25	Got LC from up
3	15 - 0	dead time

Table 33: Raw Data Format Header

Value	Function
00	ATWD Channel 0
01	ATWD Channel 0 & 1
10	ATWD Channel 0 & 1 & 2
11	ATWD Channel 0 & 1 & 2 & 3

Table 34: ATWD Size

Dead time is given in 25ns clock cycles and measures the time between an “ATWD” launch and writing data into the internal buffer memory.

5.1.2 Raw Data Format FADC Data

If available, FADC data starts at offset address $+10_{hex}$.

Address (hex)	Bits	Function
+10	9 - 0	FADC sample 0
+10	25 - 16	FADC sample 1
+14	9 - 0	FADC sample 2
+14	25 - 16	FADC sample 3
⋮	⋮	⋮
+20C	9 - 0	FADC sample 254
+204	25 - 16	FADC sample 255

Table 35: Raw Data Format FADC Data

5.1.3 Raw Data Format ATWD Data

If available, ATWD data starts at offset address $+210_{hex}$.

Table 36 shows the location of the ATWD data in the LBM if available.

Address (hex)	Bits	Function
+210	9 - 0	ATWD channel 0 sample 127
+210	25 - 16	ATWD channel 0 sample 126
+214	9 - 0	ATWD channel 0 sample 125
+214	25 - 16	ATWD channel 0 sample 124
⋮	⋮	⋮
+30C	9 - 0	ATWD channel 0 sample 1
+30C	25 - 16	ATWD channel 0 sample 0
+310	9 - 0	ATWD channel 1 sample 127
+310	25 - 16	ATWD channel 1 sample 126
⋮	⋮	⋮
+40C	9 - 0	ATWD channel 1 sample 1
+40C	25 - 16	ATWD channel 1 sample 0
+410	9 - 0	ATWD channel 2 sample 127
+410	25 - 16	ATWD channel 2 sample 126
⋮	⋮	⋮
+50C	9 - 0	ATWD channel 2 sample 1
+50C	25 - 16	ATWD channel 2 sample 0
+510	9 - 0	ATWD channel 3 sample 127
+510	25 - 16	ATWD channel 3 sample 126
⋮	⋮	⋮
+60C	9 - 0	ATWD channel 3 sample 1
+60C	25 - 16	ATWD channel 3 sample 0

Table 36: Raw Data Format ATWD Data

6 Miscellaneous

This document uses absolute addresses but this depends on the memory map used by the CPU. The addresses in this document are based on the agreed memory map.

All undefined bit are reserved for future use and if written should be 0.

This document is compiled based on the information provided by[1] Azriel

[1]The name appear in no special Order

Goldschmidt, David Nygren, Arthur Jones, Kael Hanson.

7 Corresponding FPGA Versions

Name	Version
Communication FIFO	0
Communication DPM	
DAQ	
Calibration Sources	
Discriminator Rate Meter	
Local Coincidence	
Flasher Board Interface	
Trigger	
System Clock	
Supernova	
Compression	

Table 37: Current Versions

8 Acronyms

ADC	Analog to Digital Converter
ATWD	Analog Transient Waveform Digitizer
CPU	Central Processing Unit (here the ARM in the EPXA4)
DAQ	Data AcQuisition
DMA	Direct Memory Access
DOM	Digital Optical Module
DOMAPP	DOM APPlication
FADC	Fast Analog to Digital Converter
FIFO	First In First Out
LBM	Look Back Memory
LED	Light Emitting Diode
LSB	Least Significant Bit/Byte
MPE	Multi Photo Electron
MSB	Most Significant Bit/Byte
PLD	Programmable Logic Device (here the FPGA in the EPXA4)
PMT	Photo Multiplier Tube
R2R	A R2R Ladder is a simple Digital to Analog Converter
RX	Receive
SPE	Single Photo Electron
SDRAM	Synchronous Dynamic Random Access Memory
STRIPE	The CPU section in the EPXA4
TBD	To Be Defined
TX	Transmit

9 Change Log

Version 0.x	Initial Drafts
Version 0.7	Draft Changed Register Adresses
Version 0.8	Added Flabby Local Coincidence
Version 0.9	Updated Local Coincidence
Version 1.0	Local Coincidence Cable Length
Version 1.1	Added description of Raw Data Fomat
	Added Flasher Board registers
	Updated Local Coincidence window
	Fixed Interrupt Error
Version 1.2	Added Self Local Coincidence
Version 1.3	Added DOM Status Register
Version 1.4	Added Heart Beat Mode in DAQ Register
	Added Chargestamp note to “Timestamp only” Mode
Version 1.5	Added Communications Parameters
Version 1.6	Removed Roadgrader registers
Version 1.7	Changed pedestal pattern to signed numbers
Version 1.8	Added Delta Compression Register
Version 1.9	Changed Supernova Data Register after talking to Arthur
Version 1.10	Fixed Supernova dead time range and description
Version 1.11	Added IceTop Specific Register

Todo / Comments

- Dave: no calibration pulse if ATWD if busy already
- Note to myself: AHB2 is half the CPU speed
- LaTeX

List of Tables