16-Channel Discriminator/Scalar VME Module
Upgrade Revision

(DRAFT Specifications)

Jefferson Lab
Fast Electronics Group
6Mar2009

Overview

The 16-Channel Discriminator/Scalar Board contains 16 non-updating dual-threshold discriminators, programmable digital delays, and two 32-bit scalars per discriminator. The discriminator pulses are output as differential ECL logic levels through two front-panel headers. One group of outputs will connect to a TDC and the other group can be used as input to trigger logic. Each discriminator channel can be programmed to provide no pulses (masked off), pulse to TDC only, or pulse to TDC and trigger logic. The trigger output channel widths and delays are programmable. All programming is done through VME registers.

All discriminators and logic reside on 6U VME64x mainboard. Each channel contains two analog receiver fast comparator (discriminator), and pulser. Each discriminator channel has 2 programmable thresholds which can be programmed from VME. The output pulse width is also programmable from VME, but is common for all discriminator channels. The digital delay circuit delays each discriminator pulse up to 512ns in 4nS steps. It is implemented with a high-speed (250MHz) shift register for each discriminator channel. The delay is software selectable per channel.

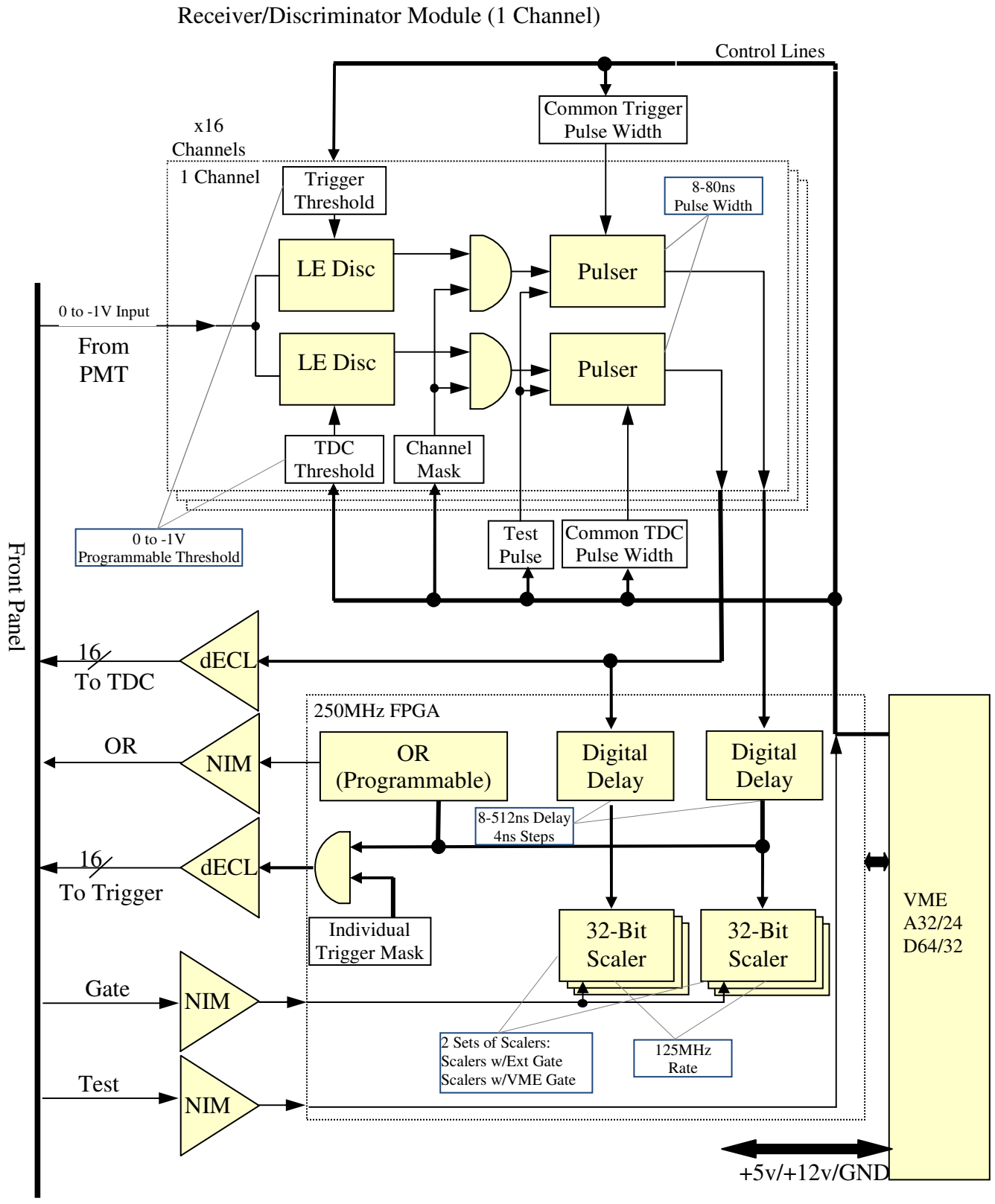
Each discriminator output pulse is recorded by a 32 bit counter (scalar) which can be gated with the external Gate (NIM) input. Scalars can be latched, read, and cleared through VME. There is a “OR” (NIM level) ouput that is the logical OR of all the unmasked discriminator outputs.

Discriminator outputs are provided as dECL levels on the front panel for interfacing with TDCs and trigger logic.

The VME interface is A32/A24/D32/D64 with support for interrupts.

Discriminator/Scaler

Functional Block Diagram



Specifications

General	Spec	Prototype
Power consumption	+12v, 500mA; +5v, 5.0A	Y: +12v,460mA; +5v, 5.89A
Fuses	+12v, 2.0A; +5v, 10.0A	Y
Dimensions	6U VME, Single-wide; 160mm card depth	Y
Front Panel I/O	Input Signals: 16 LEMO Gate Input: 1 LEMO Test Input: 1 LEMO Dual dECL Output: 2x34 Header OR Output: 1 LEMO	Y
Onboard connectors	JTAG: 2x7pin 2MM Xilinx	Y
LEDs	Green: VME Activity/Power Yellow: TDC Discriminator OR Red: TRG Discriminator OR	Y
Analog Inputs	From PMT or coaxial detector signals	Y
Channels	16	
Signal Level	+/-1.5v, DC-coupled, clamped	Y
Termination	50ohm	Y
Gate,Test Inputs	Gates scalars, Pulses Discriminator Outputs (NIM)	Y: test input to be added on next rev
OR Output	1 (NIM)	Y
Discriminator Channels		
Dual threshold control	0 to -1023mV Threshold (for each TDC and TRG output)	Y
Pulser	Non-updating	Y
Pulser Width control	8ns to 80ns width	N: 8 to 40ns width
Pulser dead-time	<2ns	N: <4.5ns
Maximum rate	100MHz w/8ns pulse setting	N: 80MHz w/8ns pulse setting
Channel-Channel Crosstalk	46dB	Y: >69dB
Input Hysteresis	5mV	Y
Noise band	<+/-2mV (no adjacent modules)	
Offset Error	<+/-3mV max, <+/-1mV typ.	Y
dECL Outputs		
Channels	Dual 16 channel output	Y
Connector	34pin header in LeCroy ECL format	Y
1st group of 16 (TDC output)	Fast discriminator output	Y
	Common width: 8 to 80ns	N: 8 to 40ns
	Programmable mask register	Y
2nd group of 16 (TRG output)	Common width 4 to 500ns	N: 8 to 120ns
	Programmable mask register	Y
Channel Threshold Control	12bit 0.25mV step (0 to -1023.75mV)	Y
Digital Delays		
Delay step size	4ns VME controlled	N: 8ns
Delay range	0 to 500ns	Y: 0 to 504ns
Uncertainty	4ns	Y
Input/Gate timing alignment	Matched	Y

Scalars		
Quantity	1 per threshold gated/free running (64)	Y
Width	32bit	Y
Input source	Digital delay	Y
Gating	External & free run scalars	Y
Count rate	125MHz	Y
Readout dead-time	none	Y
Control	VME latch, read, clear, overflow	Y
VME Interface		
Protocols	A32/A24, D64/D32	N: A32/A24, D32 only
Address space	~2kbyte	Y
Misc		
EEPROM	>1kbyte	Y – not used
Firmware Upgradable	Using VME	Y
Delays		
Input -> TDC Output	<6ns	Y: <4.5ns
Input-> TRG Output (min delay)	-	80ns min. can be improved if needed

dECL Output Connector J1

(To TDC)

Pin	Function	Pin	Function
1	Ch 1 +	2	Ch 1 -
3	Ch 2 +	4	Ch 2 -
5	Ch 3 +	6	Ch 3 -
7	Ch 4 +	80	Ch 4 -
9	Ch 5 +	10	Ch 5 -
11	Ch 6 +	12	Ch 6 -
13	Ch 7 +	14	Ch 7 -
15	Ch 8 +	16	Ch 8 -
17	Ch 9 +	18	Ch 9 -
19	Ch 10 +	20	Ch 10 -
21	Ch 11 +	22	Ch 11 -
23	Ch 12 +	24	Ch 12 -
25	Ch 13 +	26	Ch 13 -
27	Ch 14 +	28	Ch 14 -
29	Ch 15 +	30	Ch 15 -
31	Ch 16 +	32	Ch 16 -
33	No Connect	34	No Connect

dECL Output Connector J2

(To Trigger Input)

Pin	Function	Pin	Function
1	Ch 1 +	2	Ch 1 -
3	Ch 2 +	4	Ch 2 -
5	Ch 3 +	6	Ch 3 -
7	Ch 4 +	80	Ch 4 -
9	Ch 5 +	10	Ch 5 -
11	Ch 6 +	12	Ch 6 -
13	Ch 7 +	14	Ch 7 -
15	Ch 8 +	16	Ch 8 -
17	Ch 9 +	18	Ch 9 -
19	Ch 10 +	20	Ch 10 -
21	Ch 11 +	22	Ch 11 -
23	Ch 12 +	24	Ch 12 -
25	Ch 13 +	26	Ch 13 -
27	Ch 14 +	28	Ch 14 -
29	Ch 15 +	30	Ch 15 -
31	Ch 16 +	32	Ch 16 -
33	No Connect	34	No Connect

VME Accessible Registers

All discriminator board registers can be accessed through the VME bus in the following modes:

- A24/A32 non-privileged/supervisory data access (AM Codes: 0x39, 0x3D, 0x09, 0x0D)
- 32bit aligned read or write access (register specific)

Future firmware release will enable support for 32/64bit block-level-transfers if needed.

Register Summary:

Register	Description:	Address Offset:
A_THRESHOLD_CH0	Threshold Control Ch0	0x0000
…	Threshold Control ChX	…
A_THRESHOLD_CH15	Threshold Control Ch15	0x003C
A_PULSEWIDTH	Pulse Width Control	0x0080
A_CH_ENABLE	Channel Control	0x0088
A_OR_MASK	OR Output Control	0x008C
A_DELAY	Input/Output Delays	0x0090
A_VME_LATCH	VME Scalar Latch	0x0098
A_LATCH	Gated Scalar Latch	0x009C
A_TRG_SCALAR_CH0	Ext. Gate Trigger Scalar Ch0	0x0100
…	Ext. Gate Trigger Scalar ChX	…
A_TRG_SCALAR_CH15	Ext. Gate Trigger Scalar Ch15	0x013C
A_TDC_SCALAR_CH0	Ext. Gate TDC Scalar Ch0	0x0140
…	Ext. Gate TDC Scalar ChX	…
A_TDC_SCALAR_CH15	Ext. Gate TDC Scalar Ch15	0x017C
A_TRG_VME_SCALAR_CH0	VME Gate Trigger Scalar Ch0	0x0180
…	VME Gate Trigger Scalar ChX	…
A_TRG_VME_SCALAR_CH15	VME Gate Trigger Scalar Ch15	0x01BC
A_TDC_VME_SCALAR_CH0	VME Gate TDC Scalar Ch0	0x01C0
…	VME Gate TDC Scalar ChX	…
A_TDC_VME_SCALAR_CH15	VME Gate TDC Scalar Ch15	0x01FC
A_REF_SCALAR	VME Gate Ref Scalar	0x0200
A_REF_SCALAR_GATE	Ext. Gate Ref Scalar	0x0204
A_FIRMWARE_REV	Firmware Revision	0x0400
A_BOARDID	Board Identifier	0x0404

Register: A_THREHOLD_CH0 -> A_THRESHOLD_CH15

Address Offset: 0x0000, 0x0004, …0x003C
Size: 32bits
Reset State: 0x0FFF0000

31	30	29	28	27	26	25	24
-	-	-	-	TRG Threshold			
23	22	21	20	19	18	17	16
TRG Threshold							
15	14	13	12	11	10	9	8
-	-	-	-	TDC Threshold			
7	6	5	4	3	2	1	0
TDC Threshold							

TDC Threshold (R/W):
TDC CHx Threshold (in -0.25mV units)

TRG Threshold (R/W):
TRG CHx Threshold (in -0.25mV units)

Notes:
1) TRG threshold should be >25mV above TDC threshold (for same channel) to avoid introducing jitter onto timing sensitive TDC comparator

Register: A_PULSEWIDTH

Address Offset: 0x0080
Size: 32bits
Reset State: 0x01A501A5

31	30	29	28	27	26	25	24
TRG Output Pulse Width				TRG Pulser Width			
23	22	21	20	19	18	17	16
TRG Pulser Width							
15	14	13	12	11	10	9	8
-	-	-	-	TDC Pulser Width			
7	6	5	4	3	2	1	0
TDC Pulser Width							

TDC Pulser Width (R/W):
Controls pulser width for all TDC channels.
Count of 400 correspond to 10ns pulse width with slope of -180ps/count.

TRG Pulser Width (R/W):
Controls pulser width for all TRG channels.
Count of 400 correspond to 10ns pulse width with slope of -180ps/count.

TRG Output Pulse Width (R/W):
Digitally delayed/pulse stretched trigger output pulse width in 8ns counts

Notes:
1) Channel-channel pulse width variations +/-1ns. Next board revision will tighten this variation.
2) Maximum count of 420 => ~6ns pulse. Minimum count of 320 => ~30ns pulse.

3) TDC & TRG Pulser widths must be >4ns to guarantee detection of edge in FPGA (for scalars & trigger output).

Register: A_CH_ENABLE

Address Offset: 0x0088
Size: 32bits
Reset State: 0xFFFFFFFF

31	30	29	28	27	26	25	24
TRG15EN	TRG14EN	TRG13EN	TRG12EN	TRG11EN	TRG10EN	TRG9EN	TRG8EN
23	22	21	20	19	18	17	16
TRG7EN	TRG6EN	TRG5EN	TRG4EN	TRG3EN	TRG2EN	TRG1EN	TRG0EN
15	14	13	12	11	10	9	8
CH15EN	CH14EN	CH13EN	CH12EN	CH11EN	CH10EN	CH9EN	CH8EN
7	6	5	4	3	2	1	0
CH7EN	CH6EN	CH5EN	CH4EN	CH3EN	CH2EN	CH1EN	CH0EN

CHENx (R/W):
Channel X: 1 = Enable, 0 = Disable

TRGENx (R/W):
Trigger Channel X: 1 = Enable, 0 = Disable

Notes:
1) CHENx must be enabled to allow TRGENx to be enabled

Register: A_DELAY

Address Offset: 0x0090
Size: 32bits
Reset State: 0x00080008

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	TRGOutputDelay						
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	ScalarDelay						

ScalarDelay (R/W):
Scalar Input Delay 0-127 count (in 8ns ticks)

TRGOutputDelay (R/W):
Trigger Output Delay 0-127 count (in 8ns ticks)

Notes:

Register: A_VME_LATCH

Address Offset: 0x0098
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
VME_SCALAR_LATCH							
23	22	21	20	19	18	17	16
VME_SCALAR_LATCH							
15	14	13	12	11	10	9	8
VME_SCALAR_LATCH							
7	6	5	4	3	2	1	0
VME_SCALAR_LATCH							

VME_SCALAR_LATCH(WO):

Write any value to latch VME scalars.

Notes:

1) After latching scalars for readout, hardware scalars will be reset.

Register: A_LATCH

Address Offset: 0x009C
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
GATED_SCALAR_LATCH							
23	22	21	20	19	18	17	16
GATED_SCALAR_LATCH							
15	14	13	12	11	10	9	8
GATED_SCALAR_LATCH							
7	6	5	4	3	2	1	0
GATED_SCALAR_LATCH							

GATED_SCALAR_LATCH(WO):

Write any value to latch gated scalars.

Notes:

1) After latching scalars for readout, hardware scalars will be reset.

Register: A_TRG_SCALAR_CH0 -> A_ TRG_SCALAR _CH15

Address Offset: 0x0100, 0x0104, …0x013C
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
TRG SCALAR							
23	22	21	20	19	18	17	16
TRG SCALAR							
15	14	13	12	11	10	9	8
TRG SCALAR							
7	6	5	4	3	2	1	0
TRG SCALAR							

TRG SCALAR(RO):

Latched trigger threshold scalar for CHx. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF). Scalar is incremented on discriminator channel event and when external input gate = NIM logic 1.

Notes:

1) A scalar latch must be performed (by writing to register A_LATCH) to update these registers with current scalar counts

Register: A_TDC_SCALAR_CH0 -> A_ TDC_SCALAR _CH15

Address Offset: 0x0140, 0x0144, …0x017C
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
TDC SCALAR							
23	22	21	20	19	18	17	16
TDC SCALAR							
15	14	13	12	11	10	9	8
TDC SCALAR							
7	6	5	4	3	2	1	0
TDC SCALAR							

TDC SCALAR(RO):

Latched TDC threshold scalar for CHx. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF) . Scalar is incremented on discriminator channel event and when external input gate = NIM logic 1.

Notes:

1) A scalar latch must be performed (by writing to register A_LATCH) to update these registers with current scalar counts

Register: A_TRG_VME_SCALAR_CH0 -> A_ TRG_ VME_SCALAR _CH15

Address Offset: 0x0180, 0x0184, …0x01BC
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
TRG VME SCALAR							
23	22	21	20	19	18	17	16
TRG VME SCALAR							
15	14	13	12	11	10	9	8
TRG VME SCALAR							
7	6	5	4	3	2	1	0
TRG VME SCALAR							

TRG SCALAR(RO):

Latched trigger threshold scalar for CHx. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF). Scalar is incremented on discriminator channel event.

Notes:

1) A scalar latch must be performed (by writing to register A_VME_LATCH) to update these registers with current scalar counts

Register: A_TDC_VME_SCALAR_CH0 -> A_ TDC_VME_SCALAR _CH15

Address Offset: 0x01C0, 0x01C4, …0x01FC
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
TDC VME SCALAR							
23	22	21	20	19	18	17	16
TDC VME SCALAR							
15	14	13	12	11	10	9	8
TDC VME SCALAR							
7	6	5	4	3	2	1	0
TDC VME SCALAR							

TDC SCALAR(RO):

Latched TDC threshold scalar for CHx. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF) . Scalar is incremented on discriminator channel event.

Notes:

1) A scalar latch must be performed (by writing to register A_VME_LATCH) to update these registers with current scalar counts

Register: A_REF_SCALAR

Address Offset: 0x0200
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
REF VME SCALAR							
23	22	21	20	19	18	17	16
REF VME SCALAR							
15	14	13	12	11	10	9	8
REF VME SCALAR							
7	6	5	4	3	2	1	0
REF VME SCALAR							

REF VME SCALAR(RO):

Latched reference scalar for VME gated scalars. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF). Scalar is incremented at board clock rate (125MHz) and provides an accurate measurement of elapsed time since last latch of VME gated scalars.

Notes:

1) A scalar latch must be performed (by writing to register A_VME_LATCH) to update these registers with current scalar counts

Register: A_REF_SCALAR_GATE

Address Offset: 0x0204
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
REF GATE SCALAR							
23	22	21	20	19	18	17	16
REF GATE SCALAR							
15	14	13	12	11	10	9	8
REF GATE SCALAR							
7	6	5	4	3	2	1	0
REF GATE SCALAR							

REF GATE SCALAR(RO):

Latched reference scalar for external gated scalars. 32bit value indicates an overflow when count is saturated (at 0xFFFFFFFF). Scalar is incremented at board clock rate (125MHz) and provides an accurate measurement of elapsed time since last latch of externally gated scalars.

Notes:

1) A scalar latch must be performed (by writing to register A_LATCH) to update these registers with current scalar counts

Register: A_FIRMWARE_REV

Address Offset: 0x0400
Size: 32bits
Reset State: 0xXXXXXXXX

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
FIRMWARE_REV_MAJOR							
7	6	5	4	3	2	1	0
FIRMWARE_REV_MINOR							

FIRMWARE_REV_MAJOR(RO):
Major firmware revision
FIRMWARE_REV_MINOR(RO):
Minor firmware revision

Register: A_BOARDID

Address Offset: 0x0404
Size: 32bits
Reset State: 0x44534332

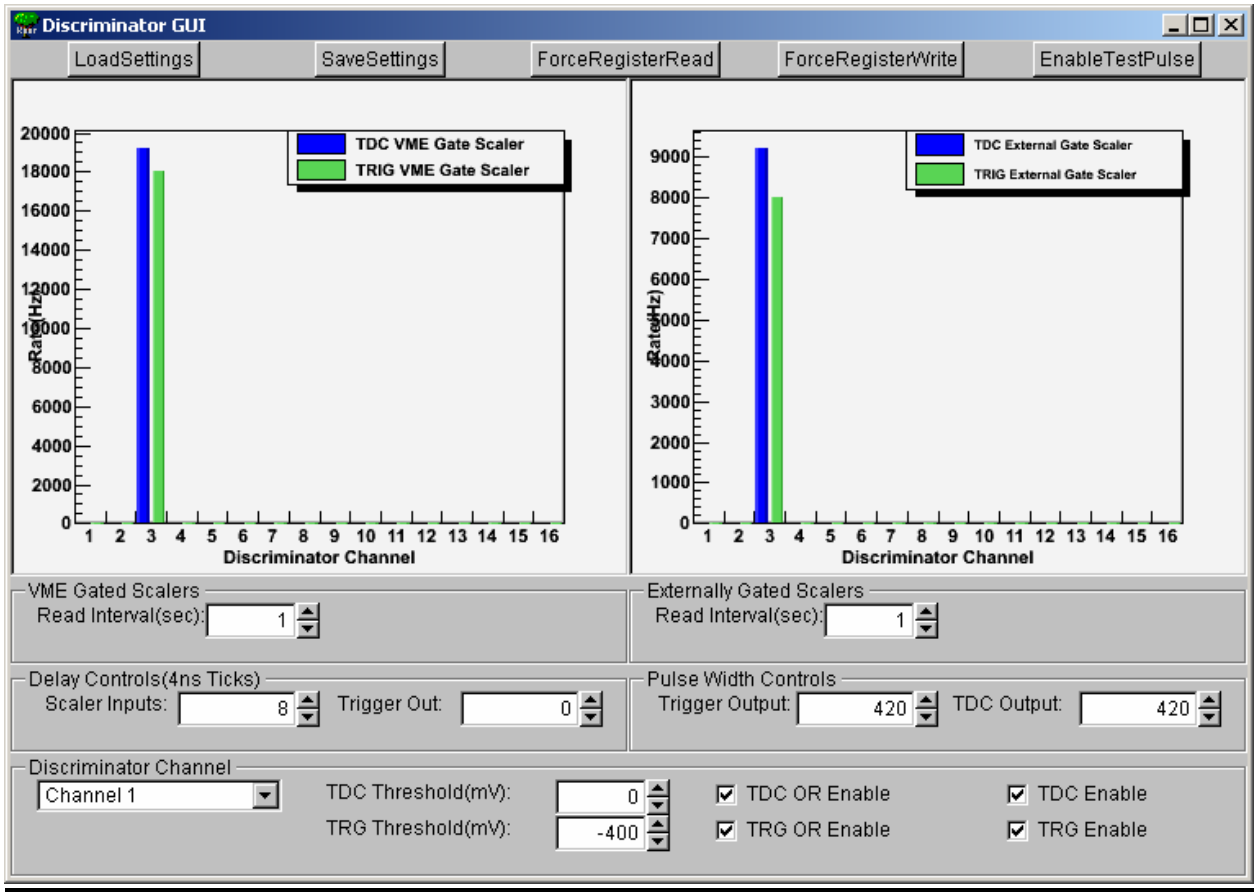
31	30	29	28	27	26	25	24
BOARD_ID							
23	22	21	20	19	18	17	16
BOARD_ID							
15	14	13	12	11	10	9	8
BOARD_ID							
7	6	5	4	3	2	1	0
BOARD_ID							

BOARD_ID(RO):
0x44544332 = “DSC2” in ASCII

Discriminator Test GUI

A ROOT based GUI has been implemented to allow for quick evaluation and testing of the dual threshold discriminator. The following features are supported:

- Implemented in ROOT for cross-platform support (Window, Linux, Solaris)
- Histogram of hit channels for both TDC & trigger thresholds (both external & VME gated scalars)
- Allows reading/writing discriminator configuration (delays, thresholds, masks)
- Configuration can be read/written to file which can easily be parsed by a simple script for configuration by separate program
- TCP/IP socket based (requires update to VxWorks server code for 32bit read/write)



Message Format:

The GUI interfaces to a TCP/IP socket based server which resides on a VME bus. The purpose of the server is to bridge a TCP/IP connection to the VME bus. The following 'C' message structures are used to send and receive information between the GUI and VME server after establishing a TCP based connection using socket **6002**.

Generic message wrapper (assumes integers are of size 32bit, Big-Endian format):

```
typedef struct
{
        unsigned int Length;
        unsigned int BoardType;
        unsigned int MsgType;
        unsigned char Msg[MAX_MSG_SIZE];
} RemoteMsgStruct;
```

Length size in bytes of Msg element for transmission
BoardType user defined to identify specific boards in case special access rules are required
MsgType defined message structure type to overlay Msg element the following. The following types are used in the discriminator GUI:

REMOTE_CMD_READ32 0x07

```
typedef struct
{
        unsigned int Address;
} Cmd_Read32;

typedef struct
{
        unsigned int Value;
} Cmd_Read32_Rsp;
```

REMOTE_CMD_WRITE32 0x08

```
typedef struct
{
        unsigned int Address;
        unsigned int Value;
} Cmd_Write32;
```

REMOTE_CMD_BLKREAD32 0x09

```
typedef struct
{
        unsigned int Address;
        unsigned short Count;
} Cmd_BlkRead32;

typedef struct
{
        unsigned int Values[BLK_MAX_LEN];
} Cmd_BlkRead32_Rsp;
```

Bit 6 of MsgType indicates a response to a request: 1 = response, 0 = request
Bit 7 of MsgType indicates an error to a request: 1 = error, 0 = no error