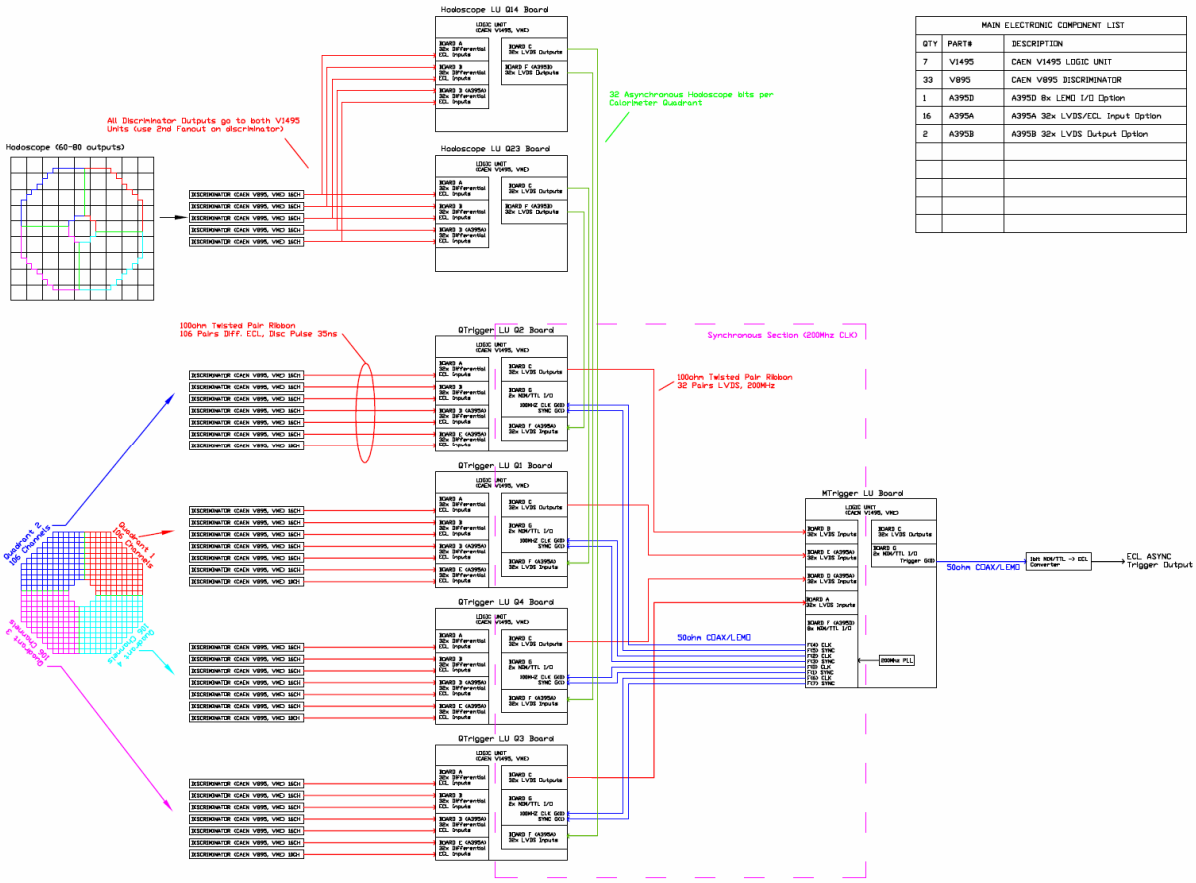
JLab Hall B 6 GeV DVCS Trigger Design

Benjamin Raydo, braydo@jlab.org, Nov. 18th, 2007

Trigger Design Description

The DVCS triggering logic uses a simple algorithm to detect clusters from the 424 channel calorimeter detector. Custom FPGA software was written for CAEN V1495 logic units. A diagram of the system is shown in Figure 1 showing the various boards, cables, and ports used for the trigger. The 424 channel calorimeter is split into 4 identical pieces (referred to by quadrants 1, 2, 3, and 4 as shown in Figure 1) where each piece is rotated 90° from the adjacent pieces. Each quadrant contains 106 channels which are discriminated and finally fed into CAEN V1495 logic units which are called QTrigger boards because each QTrigger board processes information for its corresponding quadrant of the detector. The 4 QTrigger boards send detected clusters to a central board called the MTrigger board. The MTrigger processes the detected clusters from each quadrant and detects any clusters that may be formed along the edges of adjacent quadrants. Finally, the MTrigger will generate a trigger when the current number of detected clusters exceeds a programmable value. A hodoscope is shown with corresponding hardware that will later be used for more efficient triggering. The hodoscope part of the trigger will be described when information is available on the detector (currently the rest of the DVCS trigger ignores all inputs from the hodoscope detector inputs).

Figure 1: DVCS Trigger System Setup



Summary of Specifications

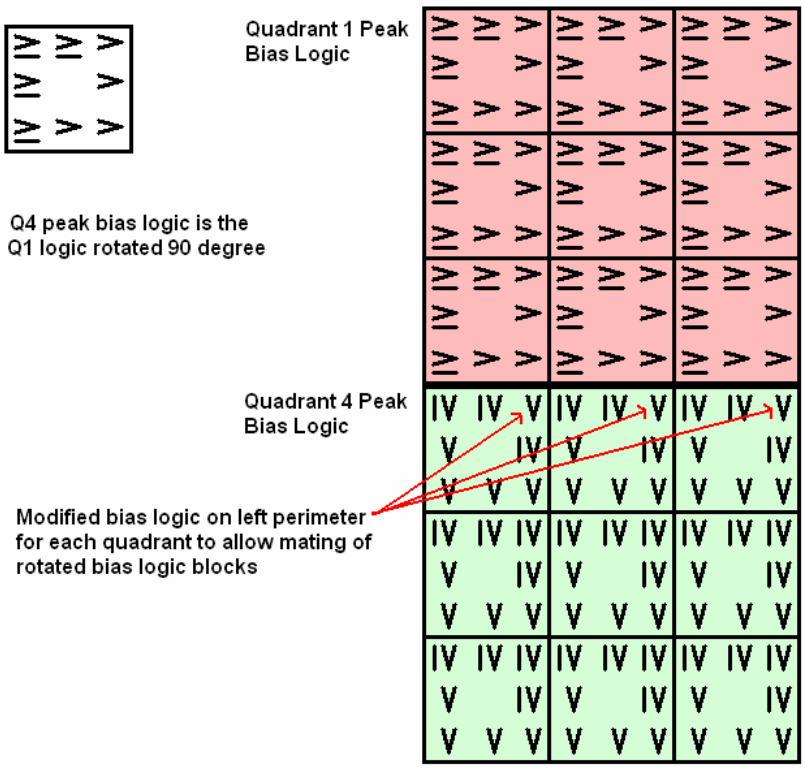
Trigger output delay: 55-70ns
Minimum trigger output width: 15ns
Tower discriminator minimum output width: 32ns
Programmable trigger on 1-31 detected clusters
Programmable cluster definition (in a 3x3 window): 1-9 tower hits
Maskable tower inputs: force 1, force 0, normal input
Cluster peak detection

Cluster Detection Algorithm

A cluster is defined as any point on the calorimeter that has more than **A_THRESHOLD** (a programmable value) number of towers hit on a 3x3 window formed by all adjacent towers (up, down, left, right, and diagonals). The trigger system computes the sum of hit towers inside each 3x3 window of the 424 points in the detector. This summing translates a map of 424 hit towers to a map of 424 window sums. The A_THRESHOLD is then applied to all 424 sums to detect the clusters.

A peak detector is applied to the resulting 424 window sums to filter out adjacent reported clusters that were formed from the same hit towers. This is important when the trigger is requiring 2 or more cluster hits, because there typically exist more than 1 identified cluster for the same hit towers because you can move the 3x3 window over by 1 and still be above the threshold. The peak detector will allow a cluster to be reported if the cluster has more towers in its 3x3 window than any of its neighbors. A bias is put in place to deal with adjacent clusters having an equal number of towers in their 3x3 windows. The following figure illustrates how the peak detection and bias is implemented:

Figure 2: Cluster Peak Detection and Bias Logic



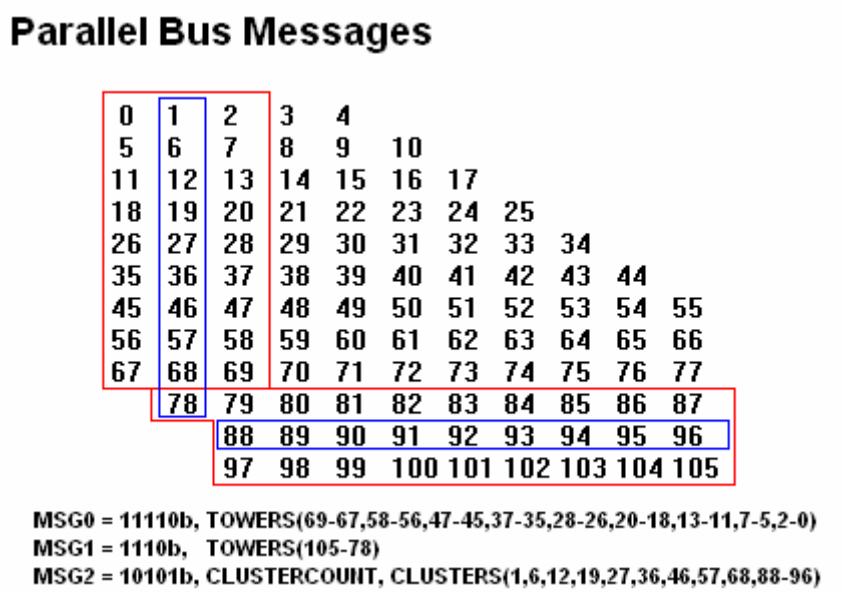
The above figure shows that at each cluster hit, it must be greater than its neighbor, or it must be greater-than-or-equal its neighbor. The “>” and “>=” were placed and chosen to ensure no loss of detection of clusters and no duplicate reporting of clusters.

A modification of the bias shown in the above figure is done because the basic bias logic is not compatible with adjacent version of itself with a 90° rotation. This modification makes the bias logic compatible at the intersections of adjacent quadrant (without this, clusters could be counted twice at these locations).

QTrigger Design

Four QTrigger boards (V1495 Logic Units) perform cluster detection on each quadrant of the detector. The symmetry of the system allows each QTrigger board to run with identical code and register settings. The QTrigger boards runs synchronously at 200MHz, which is derived from a 100MHz reference oscillator distributed by the MTrigger board. The MTrigger also distributes a SYNC signal that synchronizes the state machines and sampling times of each QTrigger. At 200MHz, the QTrigger synchronously samples the 424 discriminators every 3 clock cycles (every 15ns) to determine the current hit towers. Next, the QTrigger board looks at the 3x3 windows in its quadrant and computes a map for the quadrant containing the window sums. The window sums are then checked for valid clusters and go through the peak detection logic. Finally the found clusters are counted. During these processing stages, messages about the hit bits at the perimeter of the quadrant are sent to the MTrigger for cluster finding processing on 3x3 windows that overlap two adjacent quadrants. A 32bit parallel bus transmits information from the QTrigger to the MTrigger boards at the synchronous lock rate of 200MHz. Figure 3 shows the message format and contents sent from the QTrigger to the MTrigger.

Figure 3: QTrigger->MTrigger Message Format



The first and second message sends the MTrigger the information needed to detect clusters across quadrant as well as perform the peak finding logic. The third message sends the count of clusters found in its quadrant and the cluster bits for the bits in the quadrant that can be suppressed in the MTrigger (the QTrigger doesn’t know if these cluster need to be suppressed because it does not know about hit bits on the adjacent

quadrant that could suppress this clusters). Extra bits in each message above are fixed at a value that is currently used as keys for simple error detection.

Note: The QTrigger require that the following registers are written to:

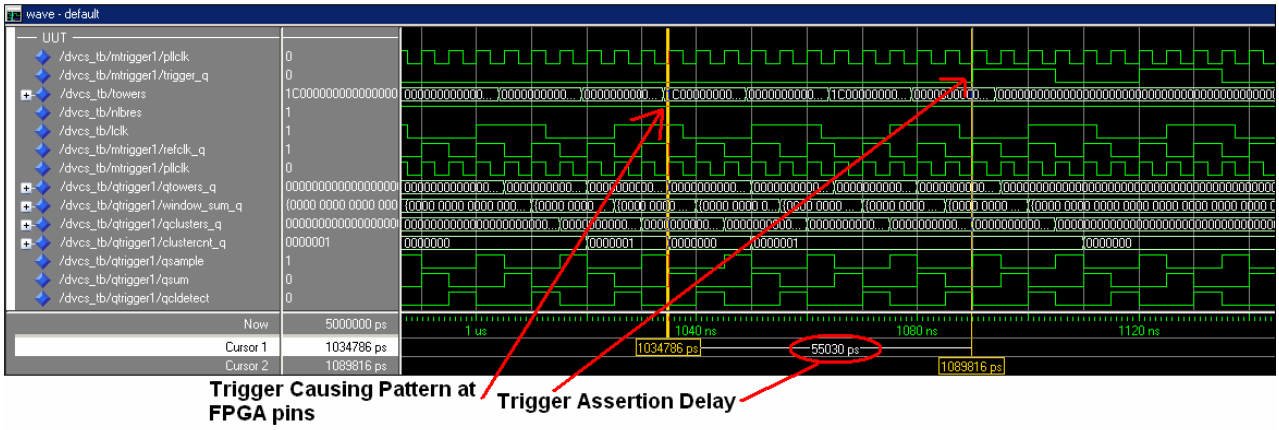
Register	Description
A_THRESHOLD	Threshold Register

See the VME Register description for further details on registers.

MTrigger Design

The MTrigger board (V1495 Logic Unit) distributes the synchronous clock and SYNC to each of the QTrigger boards and processes the output from each QTrigger board. The MTrigger board receives messages (see figure 3) from each QTrigger board over four 32bit busses running at 200MHz. The messages contain enough information for the MTrigger to detect clusters at the intersections of all QTrigger quadrants and complete the cluster peak detection processing. Any cluster found along the intersections are added to the cluster counts from each QTrigger board and compared against the programmable cluster count threshold (see below) to determine if there is a trigger. The MTrigger makes a trigger decision every 15ns (at the same rate as the QTrigger tower samples). The propagation delay for the trigger (from sampling the discriminator outputs to asserting the trigger) is ~55ns (see Figure 4), but has a jitter of 15ns with respect to the actual event (defined at the outputs of the tower discriminators) so the trigger could be asserted as much as 70ns after the tower pattern has hit the pins of the QTrigger. These delays are subject to change (by +/-10ns) until the timing has been finalized (to be completed shortly).

Figure 4: Trigger Assertion Delay



Register	Description
A_THRESHOLD	Threshold Register
A_CLUSTCNT	Cluster Count Register

See the VME Register description for further details on registers.

Figure 5: VME Module/Crate Layout

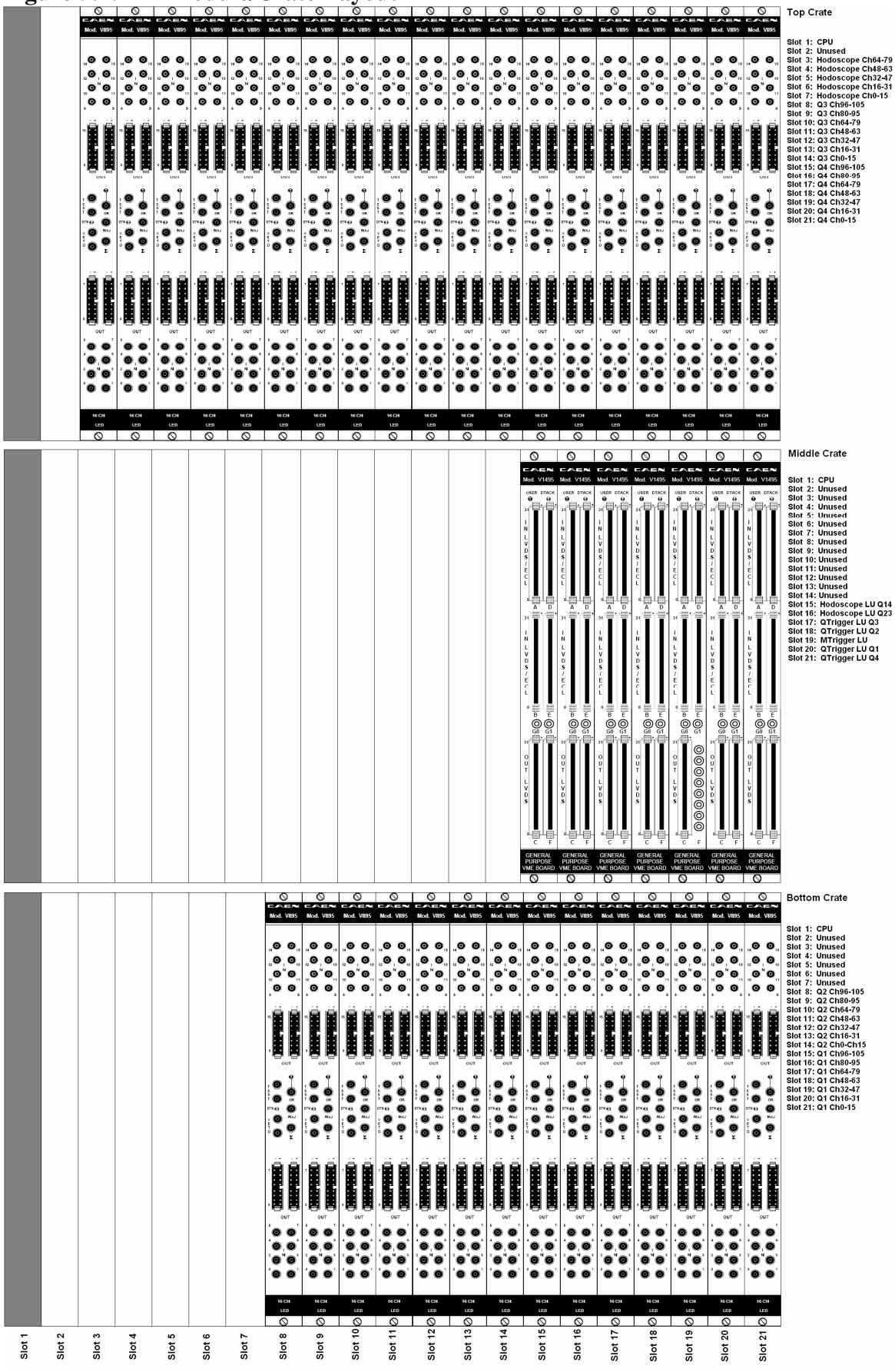
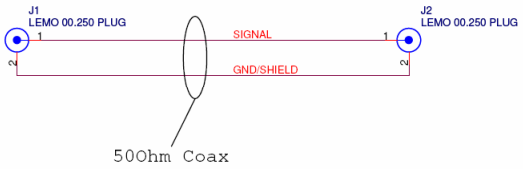


Table 1: Wiring Map

Frame Crate/Slot/Port	To Crate/Slot/Port	Cable	Description
M/19/E	M/20/C	DVCS_BUSCOM-20	Q1 Trig COM
M/19/B	M/18/C	DVCS_BUSCOM-20	Q2 Trig COM
M/19/A	M/17/C	DVCS_BUSCOM-20	Q3 Trig COM
M/19/D	M/21/C	DVCS_BUSCOM-20	Q4 Trig COM
M/19/F2,3	M/20/G0,1	DVCS_LEMO-TBD	Q1 CLK,SYNC
M/19/F4,5	M/18/G0,1	DVCS_LEMO-TBD	Q2 CLK,SYNC
M/19/F6,7	M/17/G0,1	DVCS_LEMO-TBD	Q3 CLK,SYNC
M/19/F0,1	M/21/G0,1	DVCS_LEMO-TBD	Q4 CLK,SYNC
M/15/F	M/20/F	DVCS_BUSCOM-20	Q1 Hodoscope Hits
M/16/C	M/18/F	DVCS_BUSCOM-20	Q2 Hodoscope Hits
M/16/F	M/17/F	DVCS_BUSCOM-20	Q3 Hodoscope Hits
M/15/C	M/21/F	DVCS_BUSCOM-20	Q4 Hodoscope Hits
M/15/D_ch0-15	T/7/A	DVCS_DISC-TBD	Hodoscope Hits 0-15
M/15/D_ch16-31	T/6/A	(shared)	Hodoscope Hits 16-31
M/15/B_ch0-15	T/5/A	DVCS_DISC-TBD	Hodoscope Hits 32-47
M/15/B_ch16-31	T/4/A	(shared)	Hodoscope Hits 48-63
M/15/A_ch0-15	T/3/A	DVCS_DISC-TBD	Hodoscope Hits 64-79
M/16/D_ch0-15	T/7/B	DVCS_DISC-TBD	Hodoscope Hits 0-15
M/16/D_ch16-31	T/6/B	(shared)	Hodoscope Hits 16-31
M/16/B_ch0-15	T/5/B	DVCS_DISC-TBD	Hodoscope Hits 32-47
M/16/B_ch16-31	T/4/B	(shared)	Hodoscope Hits 48-63
M/16/A_ch0-15	T/3/B	DVCS_DISC-TBD	Hodoscope Hits 64-79
M/20/E_ch0-15	B/21/A	DVCS_DISC-TBD	Q1 Tower Hits 0-15
M/20/E_ch16-31	B/20/A	(shared)	Q1 Tower Hits 16-31
M/20/B_ch0-15	B/19/A	DVCS_DISC-TBD	Q1 Tower Hits 32-47
M/20/B_ch16-31	B/18/A	(shared)	Q1 Tower Hits 48-63
M/20/D_ch0-15	B/17/A	DVCS_DISC-TBD	Q1 Tower Hits 64-79
M/20/D_ch16-31	B/16/A	(shared)	Q1 Tower Hits 80-95
M/20/A_ch0-15	B/15/A	DVCS_DISC-TBD	Q1 Tower Hits 96-105
M/18/E_ch0-15	B/14/A	DVCS_DISC-TBD	Q2 Tower Hits 0-15
M/18/E_ch16-31	B/13/A	(shared)	Q2 Tower Hits 16-31
M/18/B_ch0-15	B/12/A	DVCS_DISC-TBD	Q2 Tower Hits 32-47
M/18/B_ch16-31	B/11/A	(shared)	Q2 Tower Hits 48-63
M/18/D_ch0-15	B/10/A	DVCS_DISC-TBD	Q2 Tower Hits 64-79
M/18/D_ch16-31	B/9/A	(shared)	Q2 Tower Hits 80-95
M/18/A_ch0-15	B/8/A	DVCS_DISC-TBD	Q2 Tower Hits 96-105
M/17/E_ch0-15	T/14/A	DVCS_DISC-TBD	Q3 Tower Hits 0-15
M/17/E_ch16-31	T/13/A	(shared)	Q3 Tower Hits 16-31
M/17/B_ch0-15	T/12/A	DVCS_DISC-TBD	Q3 Tower Hits 32-47
M/17/B_ch16-31	T/11/A	(shared)	Q3 Tower Hits 48-63
M/17/D_ch0-15	T/10/A	DVCS_DISC-TBD	Q3 Tower Hits 64-79
M/17/D_ch16-31	T/9/A	(shared)	Q3 Tower Hits 80-95
M/17/A_ch0-15	T/8/A	DVCS_DISC-TBD	Q3 Tower Hits 96-105
M/21/E_ch0-15	T/21/A	DVCS_DISC-TBD	Q4 Tower Hits 0-15
M/21/E_ch16-31	T/20/A	(shared)	Q4 Tower Hits 16-31
M/21/B_ch0-15	T19/A	DVCS_DISC-TBD	Q4 Tower Hits 32-47
M/21/B_ch16-31	T/18/A	(shared)	Q4 Tower Hits 48-63
M/21/D_ch0-15	T/17/A	DVCS_DISC-TBD	Q4 Tower Hits 64-79
M/21/D_ch16-31	T/16/A	(shared)	Q4 Tower Hits 80-95
M/21/A_ch0-15	T/15/A	DVCS_DISC-TBD	Q4 Tower Hits 96-105
M/19/C	-	-	Not Connected
M/15/E	-	-	Not Connected
M/16/E	-	-	Not Connected
M/15/A_ch16-31	-	-	Not Connected
M/16/A_ch16-31	-	-	Not Connected
M/20/A_ch16-31	-	-	Not Connected
M/18/A_ch16-31	-	-	Not Connected
M/17/A_ch16-31	-	-	Not Connected
M/21/A_ch16-31	-	-	Not Connected
M/18/G1	-	-	Not Connected

Cable 1: DVCS_LEMO-XX

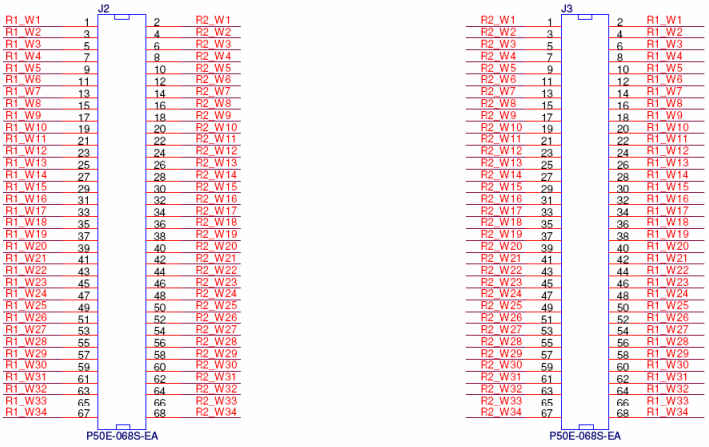


Notes:

1) Cable Length = XX" (where XX is defined in part#)
2) Cable Length is measured from Plug End to Plug End

Title: DVCS LEMO XX		
Size: A	Document Number: <Doc>	Rev: R1
Date: Friday, November 09, 2007	Sheet 1 of 1	

Cable 2: DVCS_BUSCOM-XX



Parts:

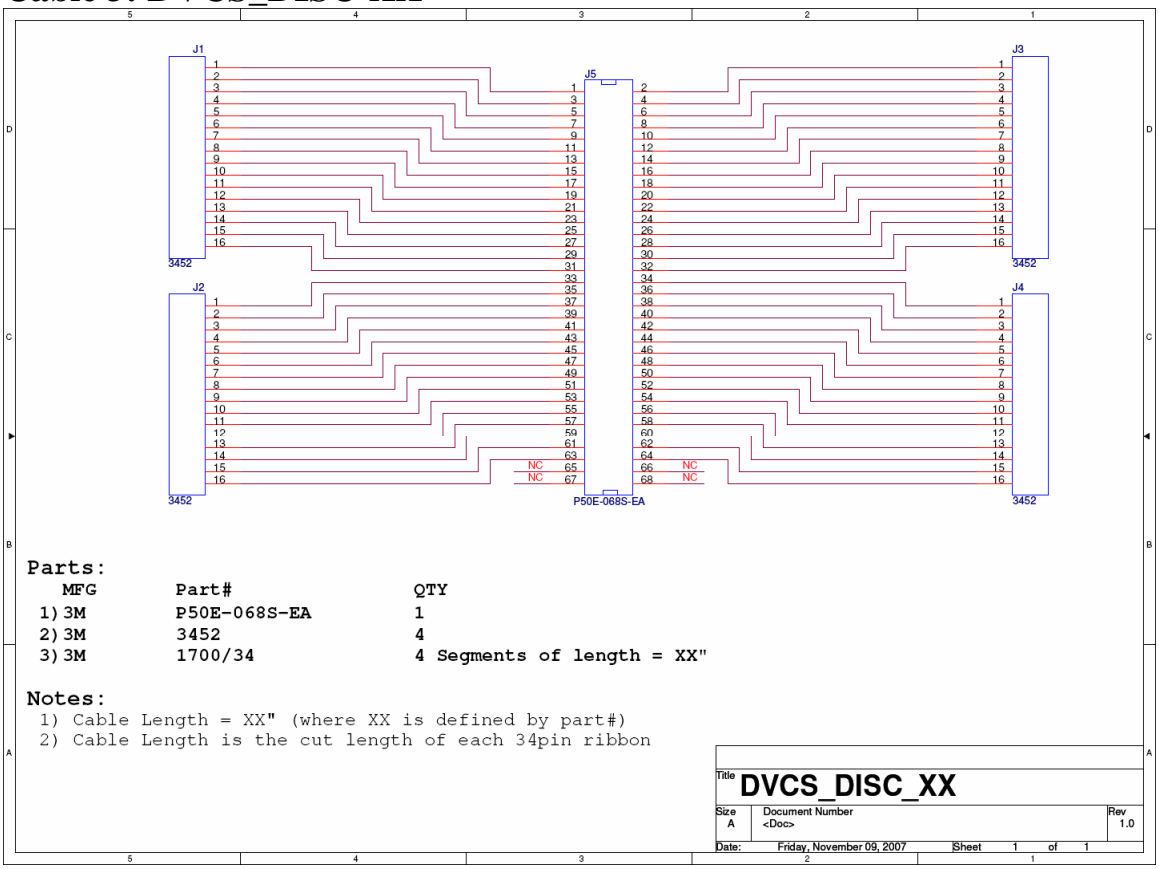
	MFG	MFGPart#	QTY
1)	3M	P50E-068S-EA	2
2)	3M	1700/34	Length = XX"

Notes:

1) Pinout is setup so that 2 ribbons lay on top of each other (no swap is need because this cable is not pinned out 1-1, 2-2, etc...)
2) Cut Ribbon Length = XX" (where XX is defined by part#)

Title: DVCS BUSCOM XX		
Size: A	Document Number: <Doc>	Rev: R1
Date: Friday, November 09, 2007	Sheet 1 of 1	

Cable 3: DVCS_DISC-XX



Parts:

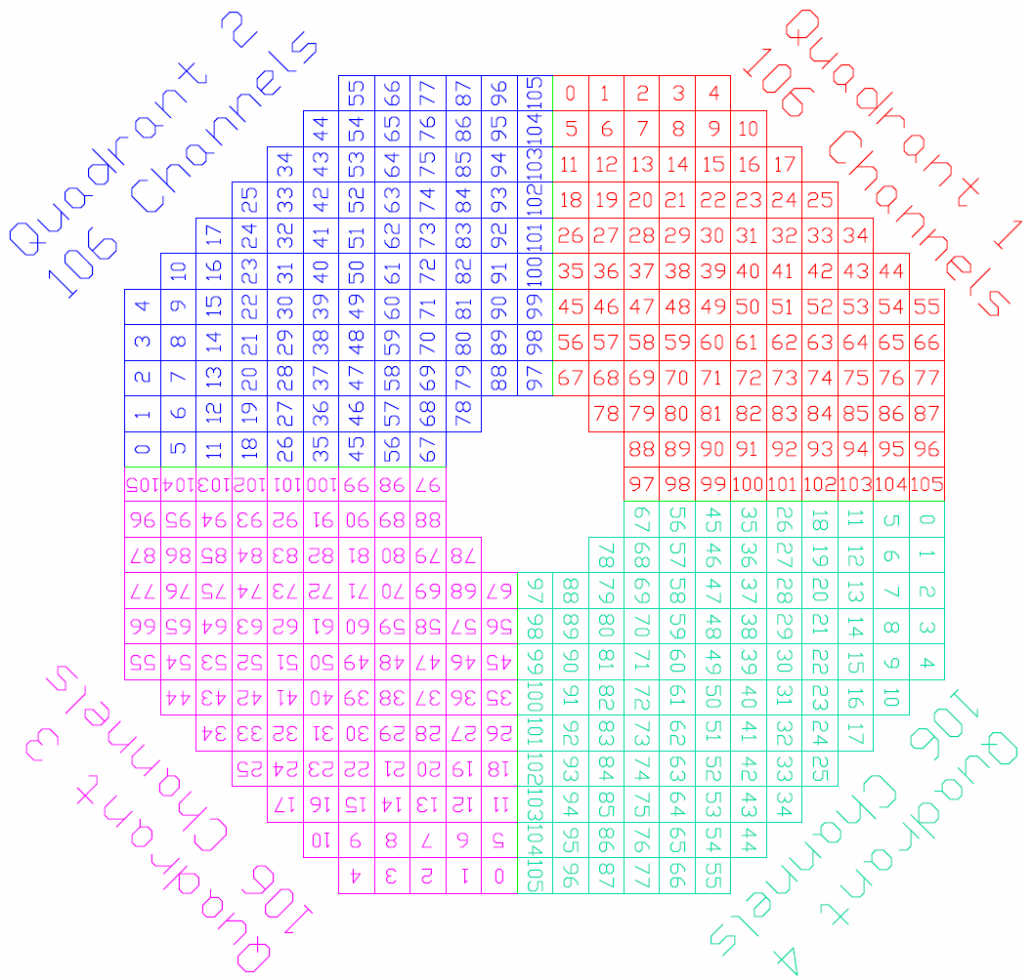
	MFG	Part#	QTY
1)	3M	P50E-068S-EA	1
2)	3M	3452	4
3)	3M	1700/34	4 Segments of length = XX"

Notes:

1) Cable Length = XX" (where XX is defined by part#)
2) Cable Length is the cut length of each 34pin ribbon

Title: DVCS_DISC_XX		
Size: A	Document Number: <Doc>	Rev: 1.0
Date: Friday, November 09, 2007	Sheet 1 of 1	

Figure 6: Detector Tower Map



Note: This image is a view of the towers looking in the same direction as the beam flow.

Follow the Table 1 for wiring the tower outputs into the discriminators. Quadrant channels should be placed into the discriminators in ascending order (i.e. Quadrant Channel 0 to Bottom Crate, Slot 21, Input 0, Channel 1 to Input 1, … Channel 16 to Bottom Crate Slot 20, Input 0, etc…).

QTrigger/MTrigger VME Register Map

Board ID Register – A_BOARDIDS (offset = 0x0000)

16bit Read-Only. Supported on QTrigger, MTrigger

Bits(15:9) Board Type ID

1001101b	MTrigger Board ID
1010001b	QTrigger Board ID

Bits(8:6)Option Board D ID
Bits(5:3)Option Board E ID
Bits(2:0)Option Board F ID

000b	CAEN A395A Board
001b	CAEN A395B Board
010b	CAEN A395C Board
011b	CAEN A395D Board

Board Revision Register - A_REVISION (offset = 0x0002)

16bit Read-Only. Supported on QTrigger, MTrigger

Bits(15:0) Board Software Revision

Ex: 0x0100 should be read as “V1.0”

Error Register – A_ERRORS (offset = 0x0004)

16bit Read-Only. Supported on QTrigger, MTrigger.
This register is automatically cleared each time it is read.

Bit(0) PLL Lock Indicator

0b	PLL has not lost lock since last read
1b	PLL lost lock since last register read

Note: MTrigger uses onboard 40MHz CLK for PLL reference oscillator. QTrigger uses 100MHz CLK on G0 for PLL reference.

Bit(1) Option Board F Error

0b	Correct Option F Board Installed
1b	Incorrect Option F Board Installed

Bit(2) Option Board E Error

0b	Correct Option E Board Installed
1b	Incorrect Option E Board Installed

Bit(3) Option Board D Error

0b	Correct Option D Board Installed
1b	Incorrect Option D Board Installed

Bit(4) Q1 Com Status (MTrigger board only, read as 0 on QTrigger)
Bit(5) Q2 Com Status (MTrigger board only, read as 0 on QTrigger)
Bit(6) Q3 Com Status (MTrigger board only, read as 0 on QTrigger)
Bit(7) Q4 Com Status (MTrigger board only, read as 0 on QTrigger)

0b	No comm error found with QTrigger comm
1b	Error occurred on QTrigger comm

Config0 Register – A_CONFIG0 (offset = 0x0010 – 0x001C)
Config1 Register – A_CONFIG1 (offset = 0x0020 – 0x002C)

16bit Write-Only. Supported on QTrigger.

Bit(x) Quadrant Tower Bit(x) configuration
Config(1:0)

00b	Tower Input uneffected
01b	RFU (to be used with Hodoscope)
10b	Force Tower Input to 0
11b	Force Tower Input to 1

Ex: Writing to the Quadrant 2 QTrigger board
Write 0x0004 @ QTrigger2_Base + 0x0010
Write 0x8004 @ QTrigger2_Base + 0x0020

This forces tower 2 input of quadrant 2 to be read as 1b regardless of actual pin value.
This forces tower 15 input of quadrant 2 to be read as 0b regardless of actual pin value.
Tower 0, 3-14 inputs of quadrant 2 are read as the real input value.

These registers represent the 106 tower inputs of the quadrant the QTrigger board represents. Refer to the Detector Tower Map. Quadrant Tower 0 begins with bit 0 at offset 0x0010 and 0x0020, and ends with Quadrant Tower 105 at bit 9 at offset 0x001C and 0x002C.

After reset, each tower input is configured are CONFIG0 = 0b, CONFIG1 = 0b

Threshold Register – A_THRESHOLD (offset = 0x0030)

16bit Write-Only. Supported on QTrigger, MTrigger.

Bit(3:0)	Defines the cluster for 3x3 windows. The number of towers hit in any 3x3 window must be greater than this value to be considered a cluster.
Bit(15:4)	RFU (Write values are ignored)

Cluster Count Register – A_CLUSTCNT (offset = 0x0032)

16bit Write-Only. Supported on MTrigger.

Bit(6:0)	When the number of clusters found are greater than or equal to this value a trigger is generated.
Bit(15:7)	RFU (Write values are ignored.

Hit Tower Register – A_TOWERS (offset = 0x0040 – 0x004C)

16bit Read-Only. Supported on QTrigger.
This register is automatically cleared each time it is read.

Bit(x)	Tower Hit Indicator	
	0b	Tower x was not hit since last read of register
	1b	Tower x was hit since last read of register

These registers represent the 106 tower inputs of the quadrant the QTrigger board represents. Refer to the Detector Tower Map. Quadrant Tower 0 begins with bit 0 at offset 0x0040, and ends with Quadrant Tower 105 at bit 9 at offset 0x004C.

HDL Source Code

MTrigger & QTrigger Files

File	Description
V1495usr/spare_if_rtl.vhd	CAEN V1495 Supplied Source Code
V1495usr/tristate_if_rtl.vhd	CAEN V1495 Supplied Source Code
V1495hal/v1495usr_hal.vqm	CAEN V1495 Supplied Source Code
V1495usr/v1495usr_pkg.vhd	Register Address Names

QTrigger Files

File	Description
V1495hal/v1495usr_q.vhd	CAEN V1495 Supplied Source Code
QTrigger/QPLLBlock.vhd	Altera Quartus Megablock PLL Generator
QTrigger/ParallelCom.vhd	Parallel Bus Transmitter
QTrigger/QTrigger.vhd	Main QTrigger Body
QTrigger/QTrigger_pkg.vhd	Cluster Functions
QTrigger/QInputMask.vhd	Tower Input Masking
QTrigger/FF.vhd	Altera Quartus Megablock DFF Generator

MTrigger Files

File	Description
V1495hal/v1495usr_m.vhd	CAEN V1495 Supplied Source Code
MTrigger/MPLLBlock.vhd	Altera Quartus Megablock PLL Generator
MTrigger/ParallelComRx.vhd	Parallel Bus Receiver
MTrigger/MTrigger.vhd	Main MTrigger Body
MTrigger/QISectClusters.vhd	Intersecting Quadrant Cluster Detection
MTrigger/QISectSum.vhd	Intersecting Quadrant Window Summing
MTrigger/TriggerDetect.vhd	Cluster Counting, Trigger Generation