Digital Gammasphere Master Trigger Firmware User Guide

Trigger Module (Router & Master Trigger)

-- *PRELIMINARY* --

August 31, 2015
Version 1.0

Originator: J. Anderson

1. GENERAL INFORMATION

This document describes the specific firmware functionality of the Master Trigger and Router Trigger firmware developed for the Digital Gammasphere (DGS), Digital Fragment Mass Analyzer (DFMA) and Helical Orbit Spectrometer (HELIOS) experiments. The same modules, with different firmware, are also used in the Gamma-Ray Tracking In-Beam Nuclear Array (GRETINA) and the Majorana experiments; the specific functionality of the firmware for those experiments is found elsewhere.

A general overview of the functions of the module is found in the *Trigger Module User's Guide*, and the first-time reader is directed to that document as a precursor to understanding this document. The *Trigger Module User's Guide* defines the hardware structure, general functionality and connections between the trigger boards and the rest of a digital acquisition system. This document presupposes familiarity with the system as a whole, the trigger modules and the communications links of the system, and focuses on the trigger algorithms and setups unique and specific to DGS and its sister experiments at Argonne National Laboratory.

The first section of this document will provide timing details generic to all trigger algorithms, focusing upon the Master State Machine logic. The second section will examine in detail how each of the trigger algorithms works in the local environment of a single detector such as Gammasphere. The third section will explore in detail the timing relationships and trigger algorithms associated with multiple-detector setups. Finally, the fourth section will document the implementation and details of the TDC function.

1.1 Overview of Control and Timing function

The Master Trigger module is the source of the master clock that synchronizes the activity of all other data acquisition electronics. The timing link operates using a two microsecond timing interval that is sub-divided into twenty command frames, each 100 nanoseconds in duration. In each command frame five 16-bit words are serially transmitted (plus embedded clock information) at a rate of 1Gbit/second. The word rate clock has a period of 20nsec (50 MHz). This 50MHz clock is extracted by a receiver of the serial command stream connected to the Timing and Control Link and is the master clock used by every front end data acquisition object. The DS92LV18 embedded-clock SERDES chips used for the SERDES link, unlike more common 8b/10b SERDES chips, transmits the data with fixed and repeatable latency. Thus, it is possible to reliably regenerate the 50MHz word clock of the transmitting board at the receiving board. The 50MHz clock may be multiplied by the acquisition module for use as a faster sampling clock through the use of FPGA resources or other PLLs. The command frames have a well defined pattern that allows a state machine in the front end readout controller to synchronize to the 2usec timing interval and to the pattern of command frames, insuring that all front ends are fully synchronized. Timing coherence is achieved by exploiting the consistent latency exhibited by the chosen SERDES parts and by trimming cables to matched lengths.

The control aspect of the control & timing function is embedded within the data sent within the command frames. The various frames have defined control features that allow the TTC to propagate slow control information throughout the system, alert readout controllers when a trigger has occurred and initiate debugging activities.

1.2 Overview of Triggering Function

Data necessary to support trigger decisions is sent at regular intervals from the front end data acquisition modules. The same bandwidth as the timing and control link, 1Gbit/sec, is available from each data acquisition module to the Routers. However, since the Routers then send all information to the Master over a single 1Gbit/sec link, either the Routers must either pre-process or compress the data, or only a fraction of the available bandwidth must be used by each front end module. Firmware within the Router has the responsibility to limit the data sent to fit within aggregate bandwidth. A variety of trigger algorithms may be developed within the Virtex-4 FPGA of the Master Trigger. Some examples include multiplicity, energy sum and pattern recognition. The data sent by the front ends may be sent as frames or each word may be treated individually. GRETINA and DGS have taken very different paths here, each optimized for the energy resolution and maximum hit rate characteristics of their detectors; the specifics of the data links for both systems are fully described in the individual documents for each system.

Once pre-processed by the Routers, the net trigger data is sent from each Router to the Master Trigger over that Router's individual 1Gbit/sec serial link. The Master's firmware design allows for up to eight distinct triggering algorithms to be simultaneously active. Each active trigger algorithm, when satisfied, latches the global system timestamp at the moment the algorithm is satisfied. Multiple trigger algorithms may fire simultaneously. A "trigger accept" message containing that timestamp is then transmitted over the timing & control link to all front end modules. The front ends are expected to implement sufficient buffering to allow for a few microseconds of trigger formation time, and are also required to implement an event queuing system such that multiple pending events may be present in the buffer at any moment. When the trigger acceptance message arrives, all events within the front end module are searched by timestamp and any whose internal timestamp falls within a given window of time relative to the timestamp contained in the messaged from the trigger should be marked for readout. The allowable limits of the window are determined by the design of the front end. There is no acknowledgement from the front end whether any trigger message has any effect; once the message is sent the trigger's task is complete.

The Master Trigger is capable of implementing a trigger Veto based either upon an external signal input or as part of the data stream coming from the front ends. A flow control method has been demonstrated whereby the half-full flag of each LBNL digitizer module's board-wide FIFO is continuously transmitted to each Router. The Routers collect these flags and pass them to the Master. If any digitizer's FIFO becomes half-full, the Master then enters a self-veto mode in which triggers are blocked until the readout has brought all FIFOs below half-full.

1.3 General Purpose I/O

The cable connection at the front of trigger modules provides for two extra twisted pairs besides those used for the Gbit serial links. These additional pairs are simple LVDS signals not intended to carry serialized data. These signals may be used for flow control and/or simple, fast trigger purposes. One of the LVDS lines is directly routed to the main FPGA of the trigger module, whereas the other is routed to a small, very fast programmable logic chip (CPLD). Five 8-bit LVTTL ribbon cable connections to/from the CPLD are found on the front panel, to support development of high speed "any channel hit" or multiplicity signals.

Two NIM inputs and two NIM outputs on the front panel of the Master Trigger module may be used to connect other detectors or signals. Firmware within the trigger module defines

the functions of the NIM connections. Generally the two NIM inputs are defined as “Auxiliary trigger in” and “Trigger Veto In”, although firmware may use them for any purpose desired. The two NIM outputs are typically set to be an “Any Trigger” output and a “Sync” output for connection to legacy devices requiring synchronization to the digital system. Finally, two differential ECL output signals are provided, along with some RS-485 connections, on an Auxiliary I/O header.

1.3.1 Special I/O considerations for the Revision D printed circuit board

The NIM receivers in the prototype, revision B and revision C printed circuit boards are slow devices intended only for inter-module connections over long lines. Starting with the revision D printed circuit boards the speed of the NIM receivers has been increased to allow for reception of fast timing signals. In the DGS version of the Master Trigger firmware, this has been exploited to implement a time-to-digital converter capable of measuring the time difference between the leading edge of the NIM input signal and the next edge of the 50MHz master clock to better than 300ps accuracy (single shot, no averaging).

A second important I/O consideration is that of the main front panel SERDES connector. Starting with printed circuit board revision D, three pin of this connector are optionally capable of directly supplying +3.3V power, intended to support powering of “paddle cards” that would plug in directly to the SERDES connector; earlier boards do not have this option.

All trigger boards manufactured after 2013 are revision D or later.

2. Module Photo, Connector Descriptions and Cabling

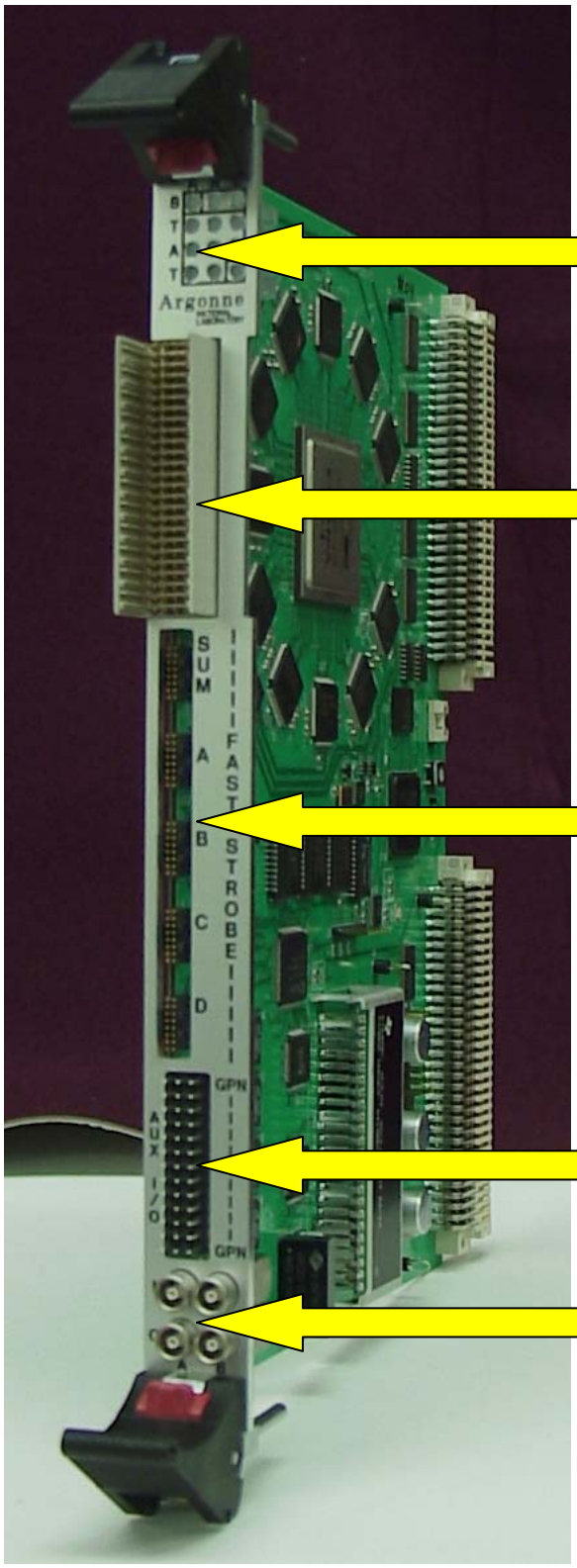


Figure 1 – Photograph of trigger module

Status LEDs. The top left LED blinks when the module is accessed over VME. The other two at the top indicate the mode of the middle eight LEDs. The lower right LED indicates the lock status of SERDES link 'L'.

SERDES link connector. This 125 pin, 2mm x 2mm pitch connector provides connection for all 11 SERDES links of the module (links A-H, L, R and U). The Trigger Paddle Board mates to this connector. Revision D and later Trigger Modules optionally provide +3.3V power via this connector for fiber-optic Trigger Paddle Cards.

Ribbon cable connectors for fast trigger cables between Trigger Modules programmed as Routers and a Trigger Module programmed as a Master Trigger. These low-voltage TTL connections are associated with the small, fast CPLD.

Auxiliary I/O connector, provided for diagnostic and general purpose connections. The top 11 rows are RS-485/TTL compatible signals, selectable as inputs or outputs in groups of four. The lower two rows are differential ECL outputs.

General purpose NIM I/O. Two inputs and two outputs are provided. The functions are dependent upon firmware load plus register settings. Generally one input is used for an auxiliary trigger input and the other for a trigger veto input. The outputs generally provide timing and/or trigger signals.

2.1 Details of SERDES Link Connector

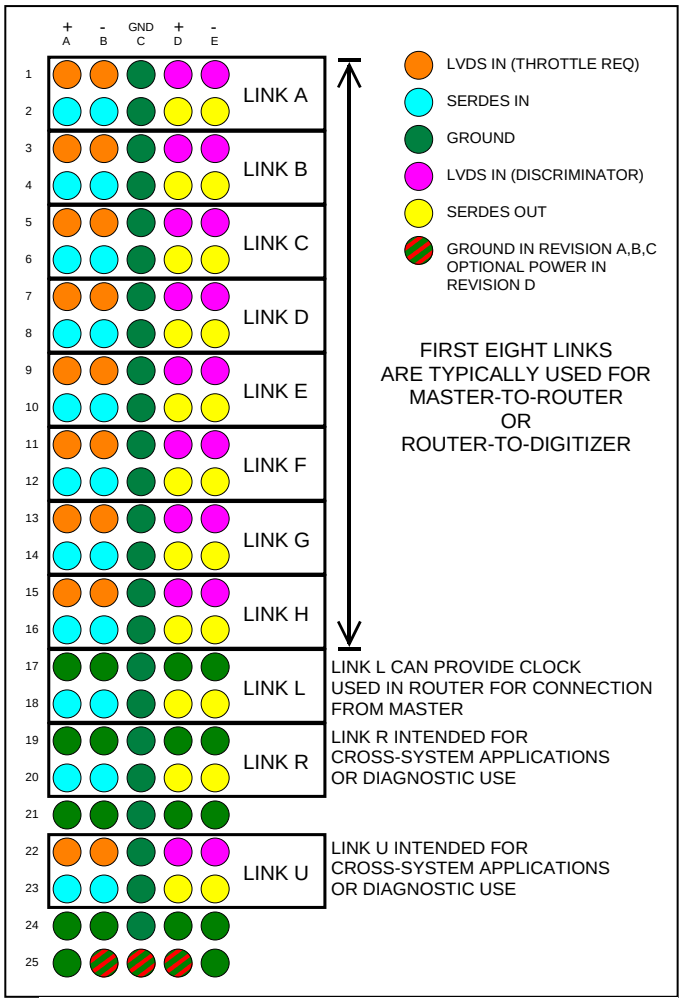


Figure 2 – Detail of trigger SERDES connector

The SERDES Link connector is a 125 pin, 2mm X 2mm "hard metric" connector. Each of the 11 links is implemented using two rows, or "wafers". Direct cable connections may be made using 2mm cable wafer assemblies such as those available from AMP/Tyco, ERNI and Meritec. All signal levels are LVDS with directionality relative to the trigger board as indicated in Figure 2 at left. The 1Gbit/sec SERDES differential pairs are shown with directionality with respect to the trigger module.

The other two LVDS lines are normally inputs to the trigger, with the most typical usage as indicated. The "discriminator" lines are internally connected to the fast CPLD for fast multiplicity triggers, specifically for the GRETINA architecture. The "throttle request" lines are connected to the main FPGA.

Links A-H, and link U, are all identically wired. Links L and R do not provide the extra LVDS lines and only provide the SERDES connection.

2.1.1 Special properties of link L

Link L has an additional special function in that the recovered RCLK from the SERDES chip is connected to a clock multiplexer chip. Under firmware control, the logic clock to the main FPGA can be selected to come from either the on-board oscillator or the RCLK recovered from link L. This allows a trigger module to operate from a clock signal sourced from another module. The typical application of this feature is to slave the clock of each Router in a hierarchical arrangement of trigger modules to the clock transmitted by the single Master, such that all boards operate synchronously. Once a set of Routers are synchronized to run off of the clock of a single Master, LBNL digitizer modules may then be synchronized to the Routers, resulting in a completely synchronous system with all ADCs running from the Master's clock.

It is also possible to connect multiple Master Triggers together in the same way to synchronize the data acquisition systems of multiple detectors to a single source clock. A MγRIAD module may be connected to a trigger module to extend the clock to an auxiliary VME crate.

2.2 Details of Master-to-Router cabling

Any of a Master's links may be connected to a Router's link L for clock distribution. The connection is detailed in Figure 3. A cable with two twisted pairs is used that plugs into any of links A-H of the Master. One pair carries the command link from Master to Router, whereas the other pair carries data from Router to Master. Up to eight Routers may be connected to a single Master using links A-H; links L, R and U of the Master are typically reserved for Master-to-Master, Master-to-MyRIAD or Master-to-Digitizer Tester links.

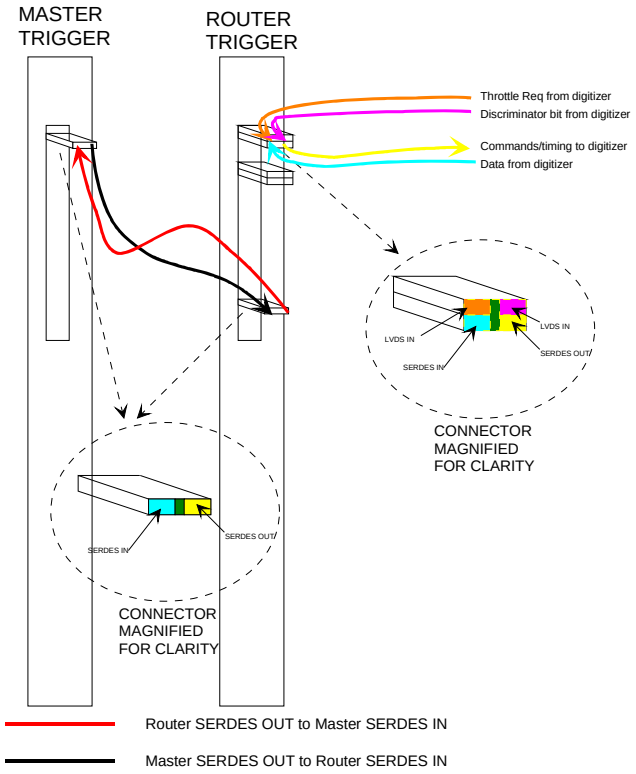


Figure 3 – details of master-to-router connection

2.2.1 Trigger Paddle Cards

Two forms of Trigger Paddle Card have been developed by Argonne National Lab personnel to ease the burden of connecting many cables to the trigger module, as shown in Figure 4. The small blue card is for use with copper cables (e.g. Cat 5e patch cords) only, and requires that the trigger module ***not*** be configured to provide paddle card power. An array of through-hole pads matching the layout of the front panel connector are provided for mounting of twisted pair cables. In the typical application Cat5 Ethernet patch cords are cut in half so that the connector end plugs into an LBNL digitizer, Digitizer Tester or MγRIAD module, and the individual wire ends are soldered into the holes of the paddle card. Larger holes are provided to strain relief the cables using small nylon wire ties.

The larger green paddle card provides identical solder points for connections to links A-H, but breaks out links L, R and U to *either* an SFP fiber optic connection *or* an RJ-45 jack. Due to impedance considerations, a green fiber-capable paddle card can only be stuffed, on a per-link basis, with one type of connection or the other; the example in the picture is for visual demonstration only. As the SFP modules require power the green paddle card is intended for use with Revision D or later trigger modules that can provide power via the SERDES connector. The card may be used with older trigger modules if an external +3.3V power source is provided.

The smaller green card in the lower left of Figure 4 is a single fiber-to-RJ45 adapter board that is physically compatible with either the Digitizer Tester, MγRIAD or LBNL digitizer modules. The small connector at the right mates with a front panel connector of the Digitizer Tester or MγRIAD for power, and a short Cat5 Ethernet jumper is routed from the RJ-45 jack of the adapter card to the RJ-45 jack of the MyRIAD. A second set of holes at the lower right of the card may be used to solder a connector compatible with a front panel connector of the LBNL digitizer, solely for mechanical stability. When used with the LBNL digitizer an external +3.3V power supply is required to power the SFP module.

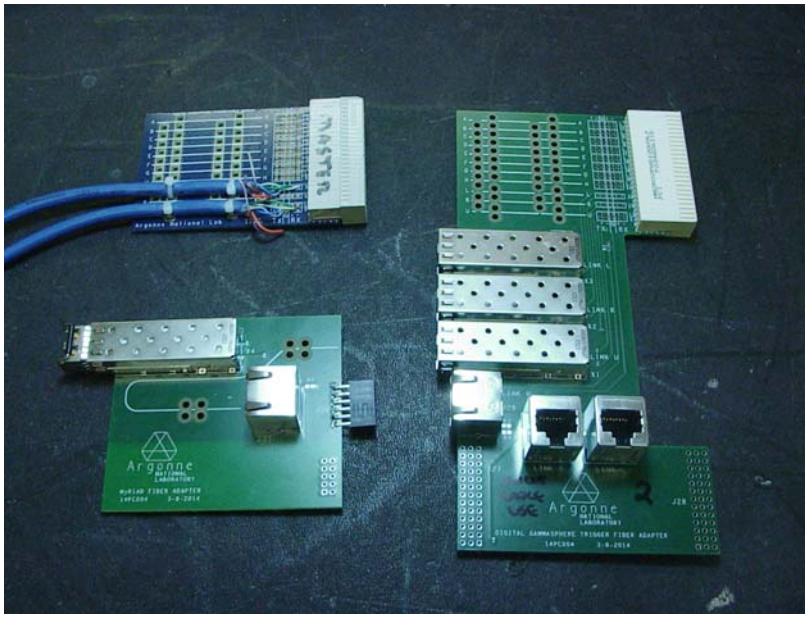


Figure 4 – paddle cards.

2.2.2 Examples of fully cabled setups

Figure 5 shows a set of modules plugged into a test stand crate to illustrate the use of the paddle cards. A Master Trigger, in the middle of the crate, has a green fiber-capable paddle card plugged into it. A copper cable connection runs from link A of the Master Trigger to a blue paddle card, plugged into a 2nd trigger module configured as a Router. A few of the links of the Router connect to a couple of LBNL digitizers and a Digitizer Tester at the left of the crate. A fiber connected to the Master's link L runs to a MγRIAD at the right side of the crate, using the single-link adapter module. The board at far right is the VME crate controller.

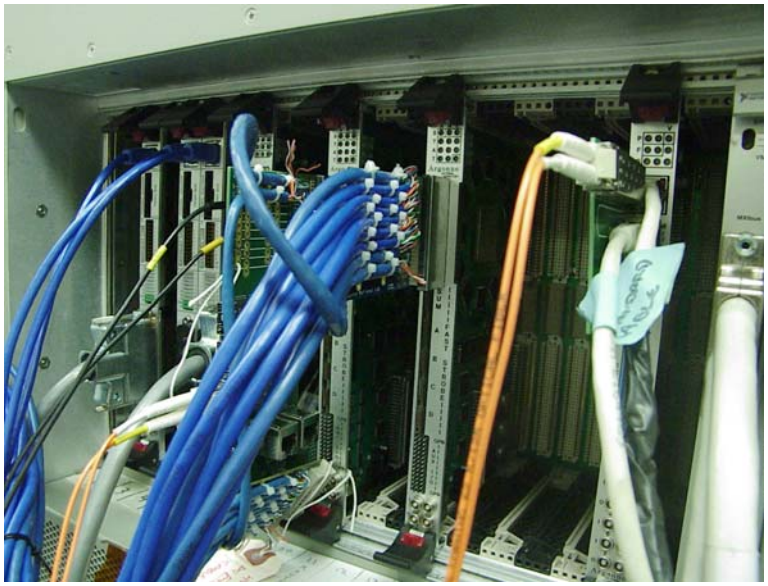


Figure 5 – Test stand demonstration of paddle card and fiber adapter use.

Figures 6 and 7 both illustrate the full DFMA system, where one Master Trigger drives 5 Router trigger boards; each Router drives 8 digitizer modules to form a 400 channel data acquisition system. The picture shows only one of the two relay racks so only 32 of the 40 digitizers are visible.

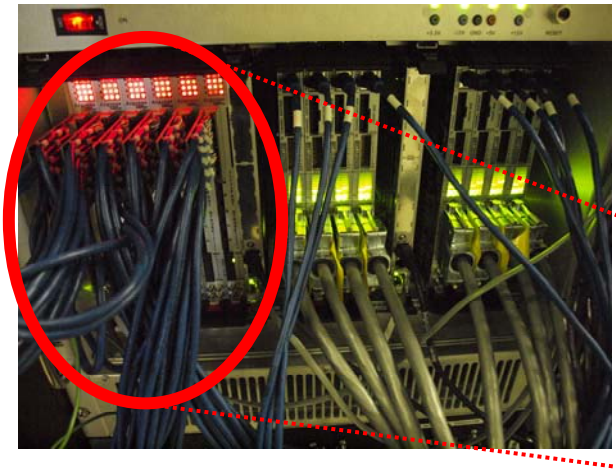


Figure 6 – Detail of paddle cards in DFMA

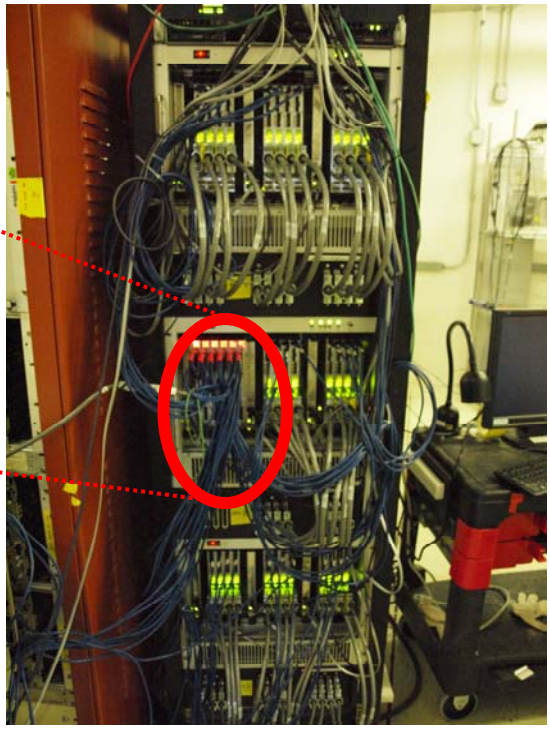


Figure 7 – A full rack of the DFMA system

Figure 8 shows, schematically, how a full system is connected. Up to 64 digitizer modules may be connected to a single Master using one rank of Routers. Links 'A' through 'H' of the Master and the Routers are used for the control and data links. Links 'L', 'R' and 'U' of the Master may be used for expansion to either auxiliary detector systems or to other Master Triggers. Links 'R' and 'U' of Routers may also be used for expansion, typically to Digitizer Testers or MγRIADs within the same system.

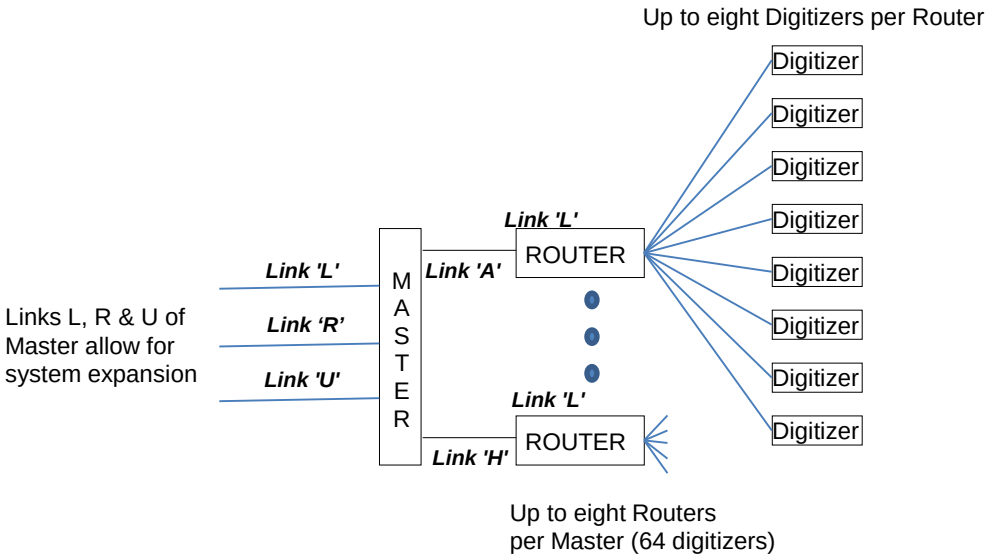


Figure 8 – Schematic of system interconnections

2.3 CPLD-to-CPLD cabling between trigger modules

Router triggers receive discriminator bits from the digitizer modules, and these are internally processed by the fast, small CPLD associated with the ribbon cable connectors. The CPLDs of the Routers, in concert with the CPLD of the Master, may be used to develop fast multiplicity sums and a signal that is asserted when such sums are above a programmable threshold. Within the Router, the sum of the number of bits set within the enabled channels is output on the front panel SUM connector. Each Router's SUM output is connected via a short ribbon cable to the A, B, C or D input of the Master Trigger, as shown in Figure 9 below.

The Master Trigger's CPLD implements a final adder to generate the aggregate sum that is compared to the threshold. If the sum is greater than the threshold, the Master Trigger's CPLD asserts the condition to the main FPGA's logic. This signal may be used by the main FPGA to generate a trigger decision.

The CPLDs use the same 50MHz clock as the rest of the trigger. Operation of a digital multiplicity summation algorithm is dependent upon the length of time any given discriminator bit is asserted. Detector signals may not be perfectly time-aligned and discriminators have time walk. The pulse width of a discriminator bit is a function of the front end digitizer. Excessively long assertion times may result in false triggers, so the Master trigger firmware typically is edge-triggered, requiring the signal from the CPLD to go back low before re-arming.

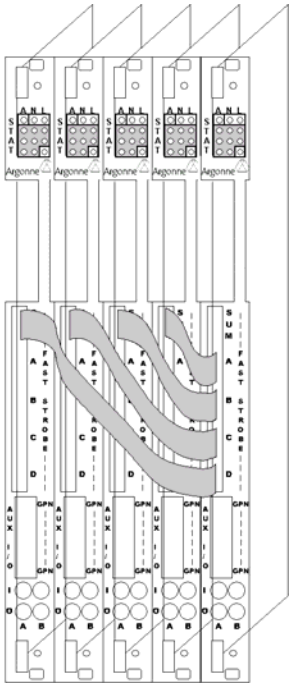


Figure 9 – Routers to Master Trigger front panel cabling

2.4 Auxiliary I/O Connector

The front panel auxiliary I/O connector is intended to provide generic input and output capability for the trigger module. This section of the hardware is implemented differently in the revision B and revision C trigger modules than it is in the revision D trigger modules. The revision D boards have a simplified structure with fewer resistor stuffing options. The AUX I/O connector is presented on the front panel as a 13 row, 3 pins per row connector, as shown in Figure x.

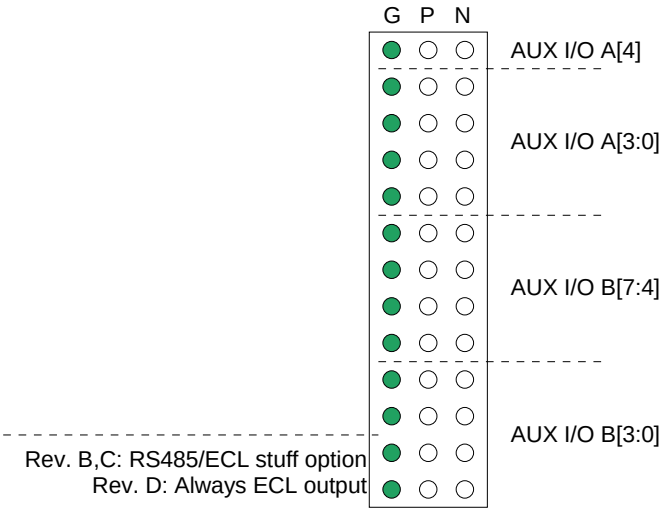


Figure 6 – front panel auxiliary I/O connector

Each row of the connector provides a differential signal connection on the P and N pins, plus a connection to the ground plane of the trigger module on the G pin. The P/N pairs are organized into four banks (A[7:4], A[3:0], B[7:4] and B[3:0]), with the *directionality* of each bank (input or

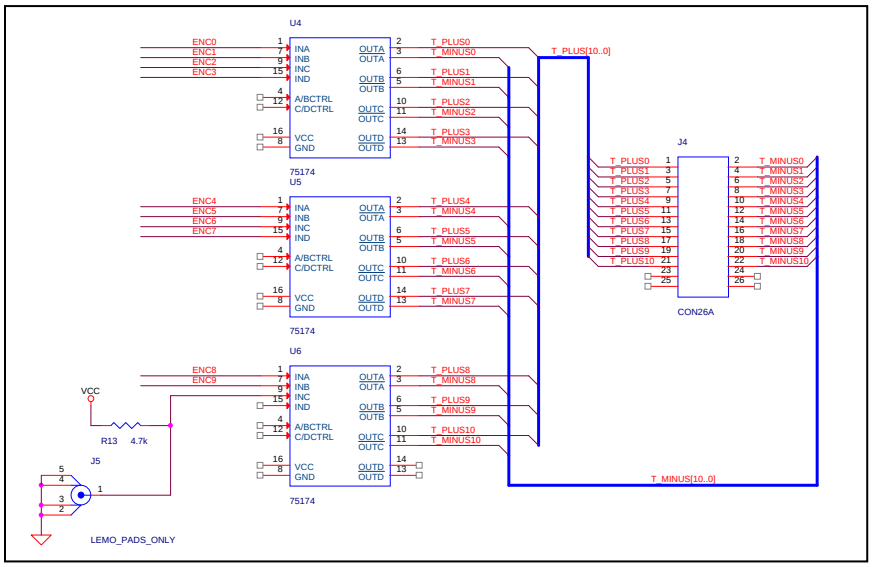
output) individually selectable through the AUX_IO_CTL register. For all pin pairs except the bottom two the input/output signal level is RS-485. TTL signal levels may be driven into the P pin, referenced to the G pin, if the bank is selected as an input.

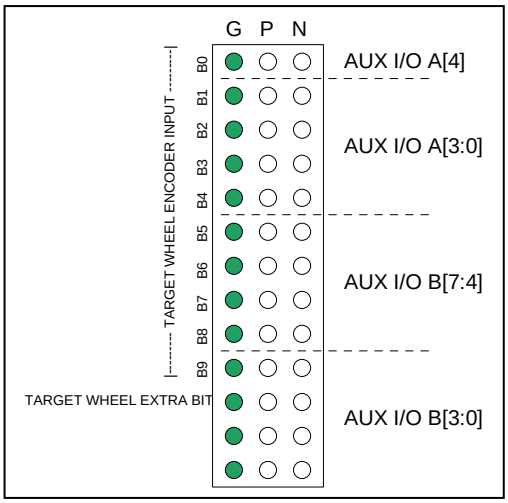
2.4.1.1 ECL option

The lower two rows of pins are typically configured as differential ECL outputs. This is a stuffing option in revision B and C boards, and the only mode of operation in revision D boards.

2.4.2 Target Wheel Interface

The Digital Gammasphere master trigger firmware implements two lookup RAMs whose function is intimately tied to the auxiliary I/O connector. In the DGS Master Trigger, the AUX I/O directionality for all banks defaults to input (except, of course, for the lower two pins that are ECL outputs), expecting that a standard 26-pin ribbon cable will be connected to the AUX I/O connector. In this configuration the top 11 pin pairs are driven by RS-485 drivers connected to a rotary encoder on the target wheel, plus a user-defined TTL input, as shown in Figure x. The least significant bit of the encoder is wired to the top-most pin.





To facilitate testing of this setup, the Router version of the firmware implements a diagnostic mode in which a programmable frequency counter is output on the Aux I/O pins.

Three RAM blocks are implemented in the master trigger, the *trigger* RAM, the *veto* RAM and the sweep RAM. Each RAM is organized as 1024 locations by 1 bit wide, and is addressed by the encoder position as read from the AUX I/O pins.

- The *trigger* RAM's output bit is connected to trigger algorithm 0 as an alternate source of auxiliary/manual trigger, enabled by a bit in the TRIG_MASK register. The trigger signal occurs once at the leading edge of the transition from a non-selected code to a selected code, so that only one trigger per encoder value can be generated irrespective of how long a given encoder position persists.

- The *veto* RAM similarly looks up one bit based upon the encoder position, and this is used as a veto signal to the trigger logic. Unlike the bit from the *trigger* RAM that only generates a trigger on a 0 => 1 transition, the output from the *veto* RAM is a level; the veto is asserted for as long as the address is present. The veto from the *veto* RAM is enabled or disabled by a bit in the TRIG_MASK register.
- The *sweep* RAM output may be used to decode which encoder positions generate the control signal for a beam sweep magnet. The sweep output is optionally routed to NIM output 2 by the AUX_IO_CTL register. Bits 7 & 6 of the AUX_TRIG_WIDTH register select one of four modes of beam sweep signal generation:
    - If the bits are 00, the output of the *sweep* RAM is directly driven as the sweep output.
    - If the bits are 01, the 'extra' input bit (driven by the LEMO input to the external module) directly drives the sweep output.
    - If the bits are 10, a leading edge in the output of the *sweep* RAM begins a timing interval, controlled by bits 15:8 of the AUX_TRIG_WIDTH register. The beam sweep output turns on with the leading edge of the output of the *sweep* RAM and stays on for the number of 655.36usec periods defined by the 8-bit counting interval (max about 167 milliseconds).
    - If the bits are 11, the same logic as case 10 applies, but with the start of the timing interval determined by the leading edge of the 'extra' input bit.
    - In cases 10 and 11, the timing interval is reset by a new leading edge and thus it is possible for the sweep control output to be on for very long periods, or even forever.

2.5 Triggers from the MyRIAD

The MyRIAD module sends a continuous stream of serial data to the master trigger module including a bit indicative of an "external detector local trigger". The master trigger implements a MyRIAD trigger as one of the choices for trigger algorithm #8.

3. General structure of trigger logic and command generation

The main FPGA of all varieties of Master Trigger firmware consists of a Master State Machine that generates the endlessly repeating 20-frame Timing and Control Link data. The Master State machine is connected to eight ***trigger algorithm*** blocks that generate ***trigger decision messages***, collected by a collection machine. Trigger messages plus other information is sent out all links A-H, plus also links L, R and U, as shown in Figure 10. A Timestamp Generator logic block in the master trigger's firmware provides a 48-bit timestamp (LSB = 10ns) that is used in all trigger algorithms and also broadcast to all front ends.

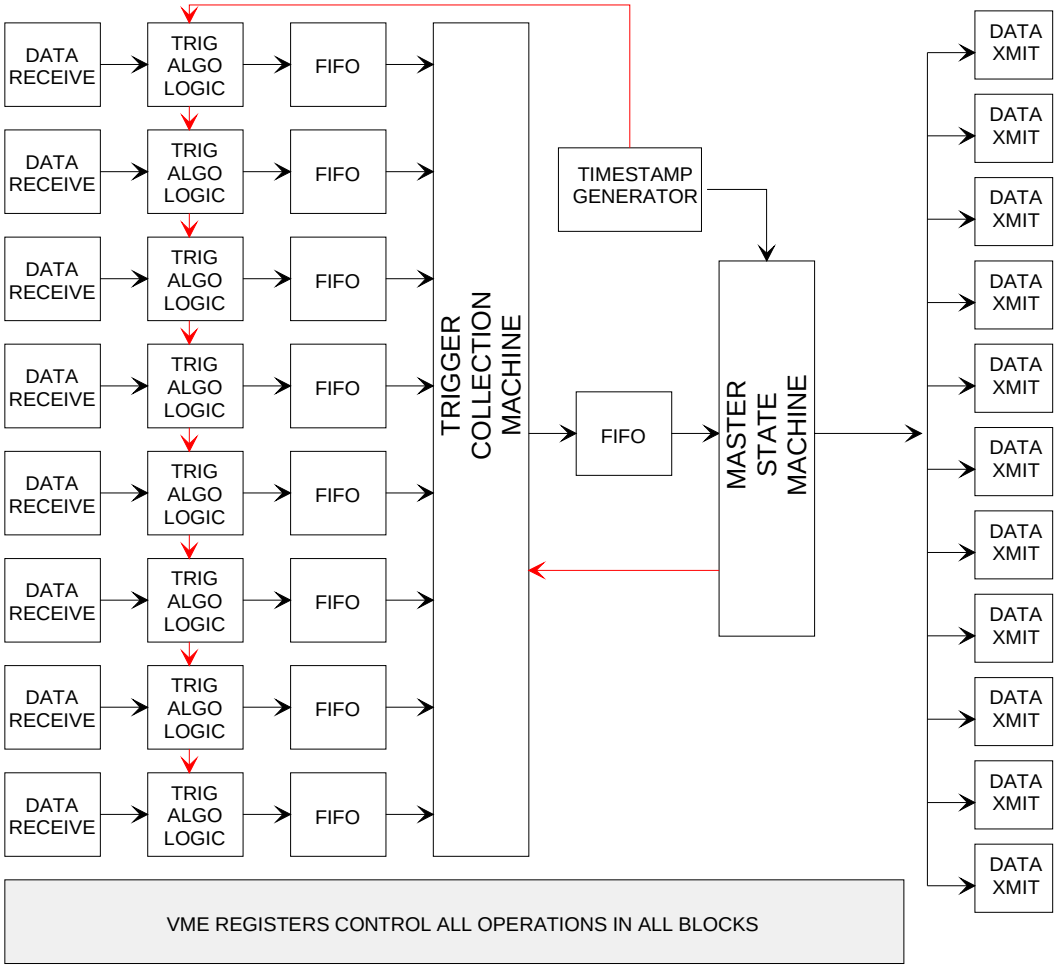


Figure 10 – block diagram of master state machine firmware structure

3.1 General methodology of triggers

The eight ***trigger algorithm*** blocks are specific to the needs of the experiment. Readers are directed to experiment-specific firmware documentation for further details. Each trigger algorithm block is enabled or disabled by a ***trigger mask*** register. A trigger algorithm may be based upon a simple signal (e.g. manual trigger, NIM input), use some combination of signals in a state machine (e.g. trigger if signal X occurs and then signal Y occurs within Z nsec of signal X), or use data collected by the Router modules from the digitizers (e.g. multiplicity or energy sum).

Each trigger algorithm latches the timestamp at the moment the algorithm is satisfied then writes the ***trigger decision message*** into a local FIFO immediately thereafter. The normal dead time after a trigger decision occurs to when the next decision may occur is typically 120ns, the time required to write the message into the FIFO. Under control of the Master State Machine, the Trigger Collection machine scans the eight trigger algorithm FIFOs every 2usec machine cycle. For each algorithm, only a single trigger decision may be transferred from the local algorithm FIFO to the collection FIFO each cycle, allowing each algorithm to momentarily accept a burst of triggers at a rate up to 8MHz, but limited to an average rate over time of no faster than 500kHz.

This triggering methodology is based upon the idea that the front end modules will internally mark *putative* events by timestamp when the analog signal crosses a discriminator threshold, and then save a list of *pending* events (that is, *putative* events that have survived internal cuts such as pileup). All *pending* events should be associated with discriminator or energy information transmitted by the front ends to the trigger system over the data link. Some subset of *pending* events will generate ***trigger accept messages***, that will be sent by the trigger to the front end after some delay, typically within a few microseconds. *Pending* events must therefore be held in the buffer of the front end for sufficient time to allow the trigger to make the decision and transmit the acceptance message back to the front end. A typical holding time is 10 to 20 microseconds. 10us should suffice for all stand-alone systems, but 20us is preferred if multiple DAQ systems are chained together to allow for cross-system transmission latency.

The front end, upon receipt of the ***trigger accept message,*** should then compare the timestamp contained within the message to the timestamps of all *pending* events. A window subtraction is assumed; for each *pending* event, the timestamp within the trigger accept message is subtracted from the timestamp of the *pending* event. If the signed result is greater than the *window max* value but also less than a *window min* value, the *pending* event becomes an *accepted* event and is read out. Events that are never selected are discarded when the holding time expires.

This technique does not require any specific correlation, other than the window of timestamp difference, between the ***trigger accept message*** and the *pending* events. Thus, external triggering from other detector systems or a sampling clock works identically to any internally generated trigger. Multiple detector systems may be chained together with one detector forming the trigger and the other simply responding to those triggers; the correlated events between detectors are automatically selected.

3.2 Timing and Control Link format summary

Table 1 summarizes the 20 frames of the trigger command format as generated in the various versions of the Master Trigger firmware. The ***Timestamp Generator*** block creates a 48-bit timestamp (LSB = 10ns), based upon the trigger board's 50MHz oscillator (or the 50MHz clock recovered from the link L SERDES). That timestamp is sent out in the ***Sync Command*** frame, once every 2 microseconds. Frames 3-10 are filled with ***Trigger Decisions***, that provide the timestamp at which a given trigger algorithm was satisfied; in DGS/DFMA, the last three of these eight frames are reserved for re-propagation of remote trigger messages.

Within the GRETINA system, the ***Demand Slow Data*** frame synchronizes transmission of energy information back to the trigger system. The ***End of Cycle Command*** frame contains fixed data values, so that the receiving state machines may synchronize to the data structure. Other frames are used for internal communication within the trigger system or for broadcasts of commands to the front ends. The specific details of the timing and command link for a given firmware type (DGS vs. GRETINA) are found within the Trigger Timing and Control Link specification document for that experiment.

Frame number	Start Time, ns	End Time, ns	Purpose of Command Frame in GRETINA and MAJORANA	Purpose of Command Frame in DGS/DFMA
1	0	100	***Sync command***	
2	100	200	Debugging Control Command ***(may affect other commands within frame)***	Unused in DGS/DFMA; null frame.
3	200	300	Trigger Decision command	
4	300	400	Trigger Decision command	
5	400	500	Trigger Decision command	
6	500	600	Trigger Decision command	
7	600	700	Trigger Decision command	
8	700	800	Trigger Decision command	Selectively propagated Trigger Decision
9	800	900	Trigger Decision command	Selectively propagated Trigger Decision
10	900	1000	Trigger Decision command	Selectively propagated Trigger Decision
11	1000	1100	Spare (Null until defined otherwise)	
12	1100	1200	Internal trigger reserved frame (Null to all front ends)	
13	1200	1300	***Demand front end slow data command***	***Unused in DGS, but checked by firmware***
14	1300	1400	Unused, reserved frame	Internal trigger commands to Digitizer Tester
15	1400	1500	GRETINA Front End Asynchronous command (Calibration Inject, Latch Status or Reset)	Unused (null)
16	1500	1600	Unused (null)	Synchronous System Capture command
17	1600	1700	Auxiliary Detector Reserved Command (Null if not required)	
18	1700	1800	Spare (Null until defined otherwise)	
19	1800	1900	Spare (Null until defined otherwise)	
20	1900	2000	***End-of-Cycle command***	

Table 1 – summary of the 20 frames of the command cycle

3.3 Internal communication between Master Triggers and Routers

Control frames sent from the Master Trigger to the Routers is buffered and re-transmitted to front end modules with minimal interference by the Routers. With the exception of frames #12 and #14, all frames issued by the Master Trigger are simply retransmitted. The Routers receive and may act upon any commands issued in frame #12 or frame #14, but shall always transmit a Null frame to the front ends in these frames. This allows the Master Trigger to have a private communication channel to the Routers, typically used for testing purposes.

The Routers act as simple fan-outs, so that the command data generated by the Master and sent over one SERDES link is replicated on all enabled SERDES links of the Router. The normal fan-out is 8:1, where a Router's links A-H are all re-broadcasting the data received by the Router on link L. Links R and U of a Router rebroadcast the unfiltered control sequence; this allows for the connection of other modules such as MγRIADs or Digitizer Testers.

The Master-to-Router connection must enter the Router on link L, as this allows the Router to use the clock embedded in the SERDES data from the Master as the Router board-wide clock by use of the clock multiplexer logic. Thus, all Routers can be locked to the clock of the Master so that all commands to all front end boards are sent simultaneously.

3.3.1 System Synchronization

Front end modules implement the same clock multiplexer logic as the trigger modules. Thus, a general system synchronization sequence may be defined:

- Set the Master to transmit the command stream with timestamps
- Set the Routers to receive the command stream, and once established, switch the clock mux to run all Routers from the Master's clock as received via SERDES link L.
- Set the Routers to retransmit the command stream to the front ends.
- Set the front end boards to receive the command stream from the Routers, and once established, switch the clock mux chips to run all front ends from the Master's clock, as retransmitted by the Routers, received in the front end via the SERDES link.
- Set the Master to issue an Imperative Sync command to reset all timestamps
- Set the front ends to send data via the SERDES to the Routers
- Configure the Routers to receive and process the front end data, and then to send aggregated front end data to the Master.
- Configure the Master to receive and process the aggregated front end data from the Routers.
- Set the parameters within the Master associated with the desired trigger algorithms.
- Enable the Master to process the desired trigger algorithms.

The details of this broad sequence will, of course, be extremely dependent upon the exact detector configuration, number and type of channels, methods of triggering desired, etc. Development of the appropriate software and scripts to initialize a particular data acquisition system using trigger modules and digitizers is a complex process that likely requires consultation from firmware and software engineers and should be approached on a case-by-case basis.

3.3.2 Clock jitter performance

Measurement of jitter performance within the trigger modules shows that the local oscillator of the trigger module has jitter of approximately 40ps. The combination of clock multiplexer and SERDES introduce another 40ps of jitter for each board-to-board jump; the vast majority of this comes from the clock multiplexer. The DS92LV18 SERDES adds very little jitter, a consequence of its phase-locked loop structure. The DS92LV18 is capable of withstanding significant jitter in the input clock, but as a consequence the DS92LV18 provides very little jitter reduction.

In a typical single detector DAQ system, the expected jitter of the data received by the Master trigger will be approximately 200ps (40ps base, plus 40ps each for the Master-to-Router, Router-to-digitizer, digitizer-to-Router, and Router-to-Master jumps). The 1Gbps SERDES has

a unit interval (UI) of 1ns, so 200ps is only 20% UI; this is within typical PLL tracking limits for the DS92LV18. Normally, data errors will not become evident until the total accumulated jitter is above 30% UI.

3.3.3 Notes on embedded SERDES elements in modern FPGAs

Many modern FPGA devices now provide multiple SERDES elements within the FPGA itself, and this often prompts queries as to whether the Trigger Timing and Control Link can be directly received by a modern FPGA without use of the DS92LV18 SERDES chip. In general, the answer is no. This is because the SERDES elements within FPGAs are specifically optimized for 8b/10b encoding of the data, not embedded-clock encoding. If a DS92LV18 is connected to the internal SERDES of a modern FPGA, even if the 8b/10b decoding blocks are bypassed, the internal SERDES will lose clock recovery when data words are sent that do not contain multiple 0-1 or 1-0 transitions within the 18-bit payload of the 20-bit serial data structure. This is a consequence of the design of the clock recovery logic of the internal SERDES.

It should also be noted that the latency of 8b/10b SERDES devices is not constant and can change each time the link is re-locked. SERDES units within FPGAs also have this feature, although it can be disabled with sufficient firmware effort. Unfortunately, even though constant-latency can be achieved, the clock recovery design still renders the internal SERDES units incompatible with the data format of the Trigger Timing and Control Link.

The Trigger Timing and Control Link was designed in 2007, before internal SERDES units in FPGAs were easily available. At that time, to meet the needs of constant latency, the DS92LV18 was the proper choice.

4. General structure of data reception

The format and structure of the Trigger Data Link is extremely experiment-specific. The formats that have been specified in the detailed documentation for Digital Gammasphere and for GRETINA explain the many differences and the reasons for them. Generally, however, digitizer modules will provide some form of discriminator information to the Router trigger modules, and in some cases may also provide some kind of data in addition to discriminator information. Irrespective of the actual format of the input data from the digitizer modules, the Router must in general always perform some kind of bandwidth reduction as a Router receives eight data streams at 1Gbit/sec, but has only a single 1Gbit/sec link up to the Master.

4.1 Methods of bandwidth reduction

Two obvious methods of bandwidth reduction have been employed by various experiments. In the GRETINA experiment, the bandwidth is reduced by having the digitizer module only report short records in response to a command (Demand Slow Data) sent by the Master trigger. During the 100 words of data sent by the Master, the digitizer will at most send back 14 words. The 14 word format from the digitizer to the Router is then compressed by removing synchronization patterns and performing a small amount of data compression so that in each 2 microsecond system cycle, there are only 12 words of data to send from each of the eight input channels of a Router. Thus, the bandwidth from Router to Master is 96 words per system cycle, just under the 100 word limit.

In Digital Gammasphere, the data from the digitizers is full bandwidth but consists of only discriminator-based information. The firmware in the Routers performs arithmetic and grouping on the discriminator bits so that the 8 words (1 per input link) received every 20ns only result in one word with data sums to send to the Master every 20ns.

4.2 Clock Domains

Data received at each input of the Routers is latched by the Router using the RCLK from the SERDES chip, as this clock is synchronized to the input data stream. However, to enable any form of data processing, all the data must cross from the local RCLK domain into the main clock domain that is synchronous to the clock as received from the Master trigger. This is normally accomplished by using FIFO buffers on each of the eight inputs, implemented within the firmware. Synchronization bits within the data stream from each digitizer are used to insure that words from multiple digitizers are simultaneously processed. At system startup, the reception FIFOs are allowed to buffer a few words in each channel before the master clock domain begins reading the FIFOs. Data is initially flushed in each channel until a word with the synchronization pattern is found; after all FIFOs are aligned at the outputs, processing begins.

4.3 Router Status Bits

As the Router processes data, it may encounter errors. The Router also is the direct receiver of any error or throttle request bits from the digitizers connected to the Router. Thus, when designing the firmware of a Router, some amount of the data bandwidth from Router to Master must be reserved for the Router to retransmit such status/error bits.

5. Circuit Implementation

A block diagram of the physical implementation is shown in Figure 12.

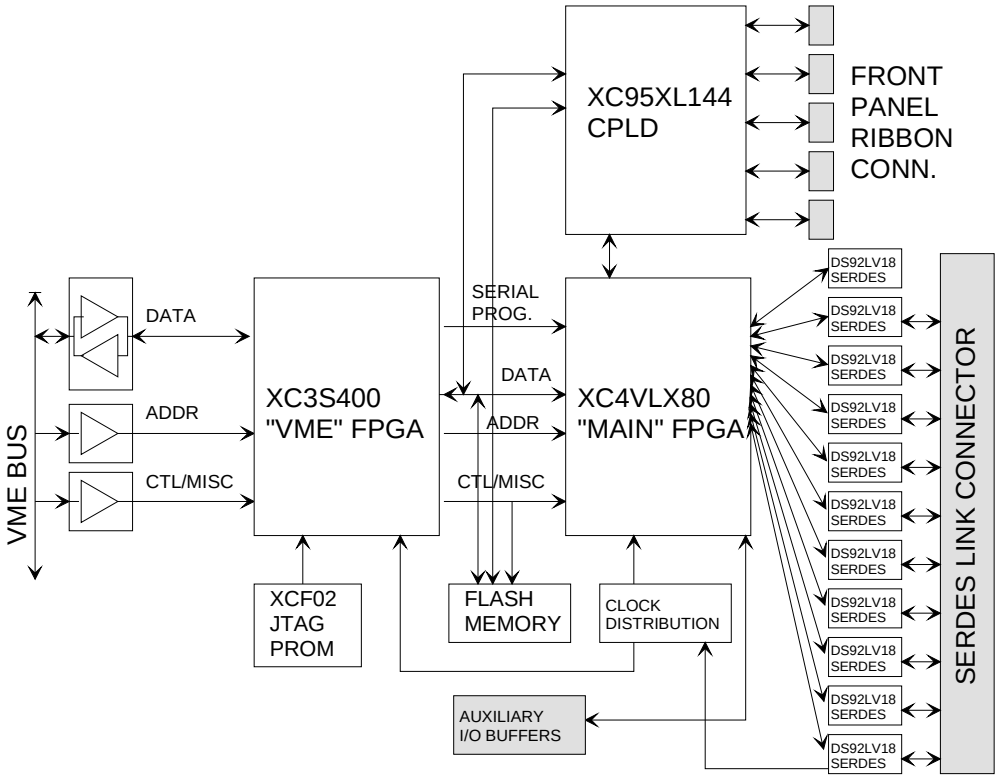


Figure 12 – Block diagram of trigger module

5.1 VME Interface

A set of 74ALVT16244 buffers connect the trigger module to a VME64X backplane. All buffered signals are connected to a small Xilinx FPGA (XC3S400). This FPGA receives its firmware load from a standard XCF02 serial non-volatile memory as its programming is expected to be stable and change rarely if at all. A subset of the VME address lines plus the VME data lines are routed to the main logic FPGA of the board. This "VME FPGA" generates internal strobe, address and data direction signals to the main logic FPGA (XC4VLX80) whenever a VME transaction matching the A32/D32 geographic address of the trigger module is decoded. The "MAIN FPGA" is connected to the "VME FPGA" alone and has no direct connection to the VME bus. Generation of VME bus errors, the VME DTACK and any timing considerations are totally under the control of the "VME" FPGA. Accesses to the "main" FPGA are mediated transparently by the "VME" FPGA.

The address space allocated to a given trigger module is typically 16K addresses starting at a base address defined by the Geographic Address lines of a VME64x backplane. While all transactions to the board are A32/D32, the internal bus between the two FPGAs is only 16 bits wide; thus all "main" FPGA registers only use bits 15:0 of the 32 bit word.

The bus connecting the "VME" FPGA to the "Main" FPGA provides 20 bits of address space and 16 bits of data. This same bus also connects to a non-volatile Flash memory chip that is used to hold the program for the "Main" FPGA. This "FPGA behind FPGA" architecture allows the board to be a fully functioning VME board with only the "VME FPGA" programmed, so that the firmware for the "Main" FPGA may be re-loaded into the flash memory and the "Main" FPGA reconfigured, at any time over the VME bus; the "VME" FPGA manages the reconfiguration automatically using state machines. The more traditional serial PROM architecture of the "VME" FPGA, however, does require the use of a JTAG dongle to change the program of the "VME" FPGA.

An alternate version of the "VME" FPGA firmware is available that supports A24/D16 VME operation if geographic address pins are not available within the VME crate. While it is possible to make the board work in a standard VME crate with only +5V power, this requires modifications to firmware in both the "VME" and "MAIN" FPGAs and thus a VME64x backplane is strongly recommended.

5.2 Main Logic FPGA

All the trigger logic is housed within a single Xilinx Virtex-4 FPGA. All modules produced to date use the XC4VLX80-FFG1148. The Virtex-4 LX series was chosen at the time of design (circa 2007) due to the relatively large number of clock domains that may be supported. Each of the 11 DS92LV18 SerDes chips presents parallel data synchronous to its own RCLK pin and the main FPGA must first reliably latch said data and then translate all received data into a common processing clock domain. The Virtex-4 series is, as of this writing, still available but is starting to become dated. At some point the board may need to be redesigned with another FPGA, but it is estimated that this will not occur for at least a few more years.

5.3 Power Distribution

Commercial DC-DC converters take bulk power from the +5V and +3.3V buses of the VME64x backplane and generate needed voltages on-board. A power monitor circuit indicates overall "power good" status and holds the FPGAs in reset until power is stable. The order of power application and rise times are controlled using high-side switches as necessary.

5.4 I/O connectivity

All data transmission between Trigger modules in the crate and between Trigger modules and front end digitizers utilizes high speed serial LVDS links. The DS92LV18 SERDES (Serializer-Deserializer) is chosen for all such connections. A single "hard metric" male connector on the front panel houses all high speed serial link connections. A mix of a few LEMO connectors and a small ribbon cable connector allow communication to and from auxiliary detector front ends.

LEMO I/O connections provide a small number of NIM-level I/Os. A 75112 open-collector driver biased to -5V (generated by a small, local, DC-DC converter) provides the required 16mA into a 50 ohm cable. Similarly, a differential line receiver converts the NIM inputs to LVTTL levels for use inside the board. Auxiliary I/O in RS-485 and ECL signal levels, plus the NIM I/O, is provided through level translators connected to the main FPGA.

5.5 DS92LV18 Embedded Clock Protocol and Latency

The following two pictures show how the serial link delay works. The delay T_{sd} defines the total amount of time from when the data is loaded into the transmitter until the *first* data bit of the serial stream associated with that data is present on the cable. Similarly, the delay T_{dd} defines the amount of time from when the *first* serial bit arrives at the receiver to the assertion of the *entire* packet on the parallel outputs. Thus, the total time from parallel data in to parallel data out is given as

$$T_d = T_{sd} + T_{dd} + T_{pd}$$

Where T_{pd} is the propagation delay of the cable interconnect plus any driver/receiver buffer chips that may be present.

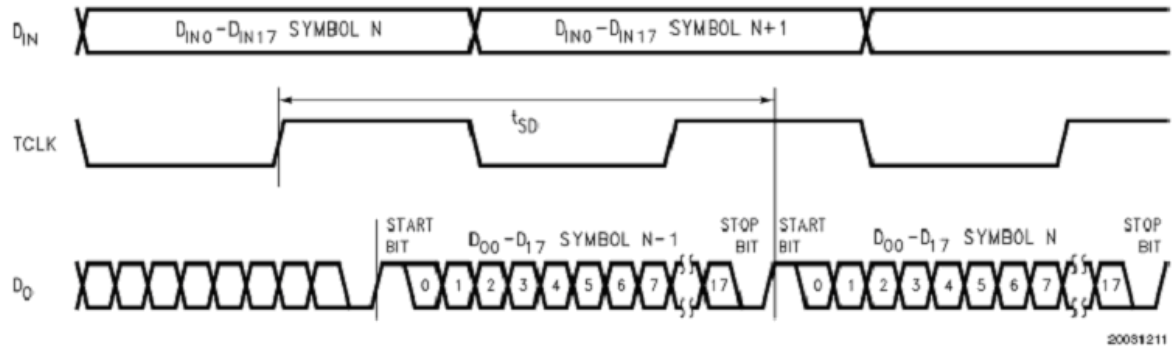


Figure 7 - Total serializer transmission delay (from DS92LV18 data sheet)

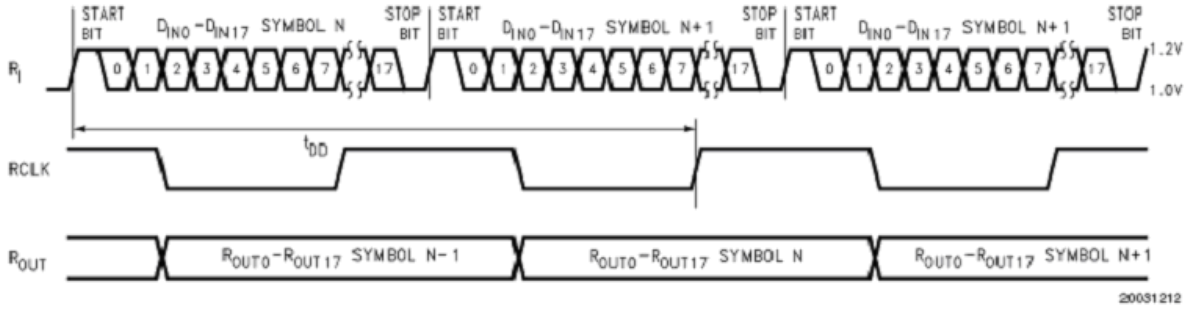


Figure 8 - total deserializer receiver delay (from DS92LV18 data sheet)

According to the data sheet, T_{sd} is defined as (T_{tcp}+1ns) ➔ (T_{tcp}+4ns) where T_{tcp} is defined as the period of the parallel input clock. In this system, a 50v.0MHz clock is used, thus T_{tcp} is 20nsec. Thus, T_{sd} ranges from 21ns to 24ns. On the receipt side, T_{dd} is defined as ranging from [(1.75*T_{rcp})+2.1ns] to [(1.75*T_{rcp})+6.1ns], where T_{rcp} is also the period of the system clock. Thus T_{dd} ranges from 37.1ns to 41.1ns.

To estimate T_{pd}, a fairly common delay value for 100 ohm twisted pair cable of 1.5ns/foot may be used. The cabling for GRETINA will likely be contained within a few relay racks; a length of

eight feet for the distance from a digitizer to a trigger module, and a distance of 0.5 foot for the interconnects between trigger boards will suffice. The total delay between modules is then estimated as

T_{d1} = (21 ➔ 24)ns + (37.1 ➔ 41.1)ns + 12ns , or 70.1 ➔ 77.1 ns.

6. CONTROL INTERFACE SPECIFICATIONS

The trigger module is implemented as a 6U Eurocard device consistent with IEEE 1101.1 mechanical specifications. For the control interface, a VME64x implementation is chosen.

6.1 VMEbus Interface

A straight-forward VME interface consisting of buffers plus an FPGA is used. This smaller FPGA performs necessary decoding functions and provides read/write strobes to the main trigger logic FPGA when a given trigger module is addressed. Each trigger module appears to VME as a block of contiguous addresses.

6.1.1 Addressing Modes

The trigger module responds to A32/D32 addressing cycles in user or supervisory mode. The BERR* (bus error) line is asserted for all unsupported AM codes.

6.1.2 Data Cycle Types

Either single or block transfers may be used to communicate with the trigger module. The trigger module does not utilize an internal address pointer so block transfers may be used either to read blocks of registers (address auto-incremented by controller) or FIFOs (address does not change).

6.2 Interrupts

The trigger module is physically capable of asserting the IRQ* lines of the VME backplane. The present firmware design has no condition under which an interrupt is asserted.

7. RESET, POWER UP RESET & INITIALIZATIONS

The DC-DC converter used for the power supply provides simple sequencing to insure the order of voltage turn-on is correct. Upon application of power the "VME" interface FPGA automatically loads its program from its separate non-volatile memory. The main logic FPGA does not implement an automatic download upon power-up. Instead, the main logic FPGA is configured in a slave mode under the control of the interface FPGA. After the interface FPGA has initialized itself, a second stage of bootstrap occurs wherein the interface FPGA programs the main logic FPGA from a small array of flash memory chips on board.

A state machine within the interface FPGA controls the programming of the main FPGA. Upon VME request, it is possible to initiate another programming cycle of the main FPGA using information from a different area of the flash memory. At any time desired, VME processors may download a new trigger logic program into the flash memory that takes effect the next time the main FPGA is reconfigured. This allows development of new algorithms and remote download without the need to remove the board. It is possible for software to manage two copies of firmware in the flash memory so that a "standard" and an "alternate" firmware load may be available.

The Trigger module executes a full reset and reload of both FPGAs in response to the SYSRESET* line being asserted. There is a manual pushbutton reset, but due to front panel space limitations this is located at the rear of the board and is generally inaccessible when the board is inserted into a VME crate.