

Project Specification & Design

Project Name: ATLAS Second Level Trigger Prototype ROIB

Version 3.01
October 25, 2003

1. Scope

This project encompasses production and testing of a prototype system for interfacing the ATLAS first level trigger to the second level trigger.

Related projects and documents

- 1) ATLAS High Level Triggers, DAQ and DCS Technical Proposal at http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/SG/TP/tp_doc.html
- 2) S-LINK documentation at
- 3) Specification of the LVL1 / LVL2 trigger interface at <https://edms.cern.ch/document/107485/1>
- 4) A Prototype ROIB for the Second Level Trigger of ATLAS Implemented in FPGA's, R.Blair et al., LEB'99, Snowmass, September 20-24 1999.
- 5) ROIB Requirements at <http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/DataFlow/DataCollection/docs/DC-014.pdf>
- 6) The level-1/level-2 interface: ROI Unit, Y.Ermoline, ATLAS DAQ note 94-34, 8 December 1994.
- 7) The Level 2 Supervisor Requirements, at <http://press.web.cern.ch/Atlas/GROUPS/DAQTRIG/DataFlow/DataCollection/docs/DC-009.pdf>
- 8) Technical Manual, 8101/8104 Gigabit Ethernet Controller, LSI Logic, November, 2001
- 9) Altera Data Book, Altera Corporation, San Jose, California

2. Technical Aspects

The Region of Interest Builder (ROIB) is intended to build ROI records from data received from the Level 1 trigger elements, select a target Supervisor processor, and distribute the records at high input rate to a number of commodity Supervisor PC's. Figure 1 shows a use-case indicating how the ROIB interacts with the first level and LVL2. The ROIB takes raw event fragments from various level one sources, assembles all the fragments of a given event into an ROI record, and then sends the ROI record to one of a number of Supervisor PC's. From there the ROI record will be distributed to

L2PU's that require it for further event selection and disposition. Using this *divide and conquer* approach a single Supervisor Processor never sees the full Level 1 rate and it can manage the required IO.

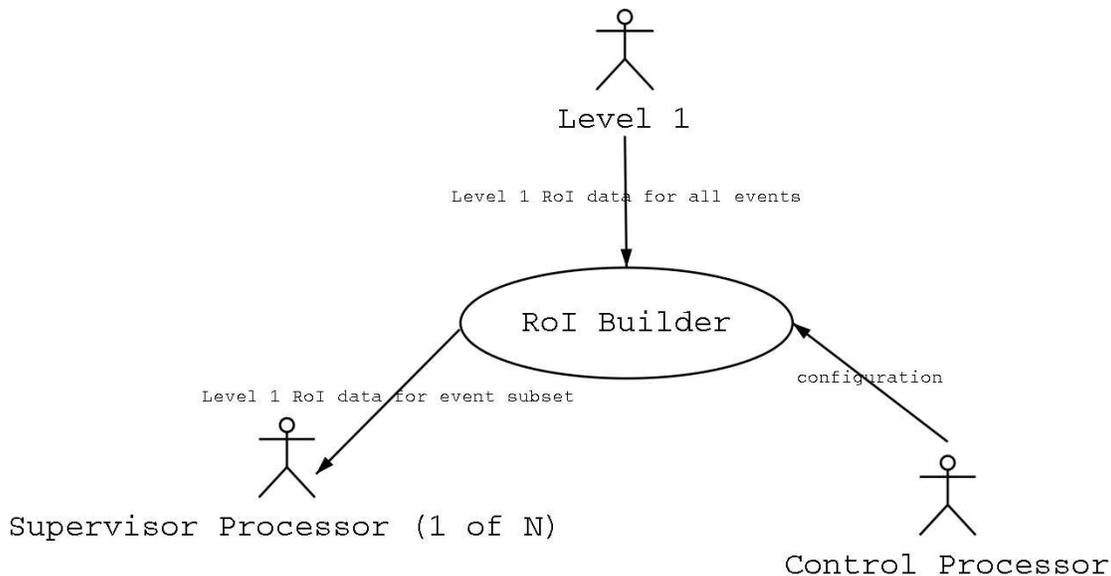


Figure 1: Use-case showing the ROIB context.

3.1 Requirements and Specifications

The ROIB must satisfy the following major requirements

- 1) Collect ROI Fragments from the active Level 1 sources
- 2) Assemble an ROI record for each Level 1 trigger
- 3) Allocate a Supervisor processor to each event, and send the ROI record for each event to the selected Supervisor processor
- 4) Interface with a control system to allow for run coordination and reset functions
- 5) Operate at Level 1 event rate
- 6) Handle corrupted or missing input data in a well defined manner that allows for proper error recovery. The ROIB must continue to run in the event of corrupted or missing data.
- 7) Have on-board diagnostics that allow verification of the operation without the need of other hardware
- 8) Provide histogramming of various quantities of interest

The prototype ROIB shall satisfy all of the requirements of the final system.

3.2 Technical Description

The architecture is conceptually similar to the prototype ROIB that was developed for the hardware integrations of trigger elements that were accomplished over the last few years,

and described in our paper for the Fifth Workshop on LHC Electronics at Snowmass in 1999. Basically, the system implemented a highly parallel architecture realized in FPGA's. Incoming fragments were distributed to several record building channels. Using the embedded EL1ID's the logic was able to allocate ROI fragments from particular events to the correct channels. (see, Proceedings of the Fifth Workshop on LHC Electronics, 1999, page 323)

In this new prototype ROIB the ROI fragments will be brought to Input cards of the ROIB via S-LINK. Each Input card accommodates 3 input S-LINK LDC's, and can service up to 4 ROIB cards. All transfer of information from the Input Cards to the ROIB cards is via J3 and AUX Cards mounted in the rear of the crates. Each Input card also has a diagnostic RAM, 32K initialized from VMEBUS which allows an on-board diagnostic system to emulate Level 1 fragments, and enables the operator to verify the ROIB system in a stand alone mode. In our previous prototype the diagnostic RAM was 32K S-LINK words deep. We will make the diagnostic RAM's on these Input Cards 32K words deep. In *Stand Alone Mode* the ROIB System performs its functions without any input from Level 1 Trigger Elements or from external devices emulating Level 1. Instead input data streams are provided by the diagnostic RAM's resident on the Input Cards. These RAM's are loaded from VMEBUS in block transfers, and the contents may be data for diagnostic purposes (such as 5's and A's, shifting 1's or 0's, etc.), or may be test vectors from Monte Carlo or simulation results. For these purposes, the S-LINK Supervisor output from the ROIB card could be routed via S-LINK to a processor resident in the crate which would execute the diagnostic codes.

We will provide the capability to use this diagnostic RAM in an alternate mode, where instead of being written from the VMEBUS, the RAM is written with Fragments from the incoming data stream from Level 1. The contents of the diagnostic RAM can then be accessed in block transfers from VMEBUS, and the received Fragments can be examined for diagnostic or system monitoring purposes.

We will produce an Expander card which will accept fragments from an Input card and expand the number of outputs so that three additional ROIB cards can be serviced. This will allow the ROIB system to be expanded to service additional Supervisor processors. We also expect to produce a mezzanine card for bringing TTC into the ROIB system. This card will appear to an Input card to be an S-LINK LDC, but will instead receive input from the TTC fiber into a TTCRX. The TTC information will be formatted on the mezzanine card to resemble an ROI fragment, and will be accepted and processed by the ROIB as if it were an ROI fragment.

Each fragment contains ROI data collected from a portion of the Level 1 trigger system. The Level 1 information required for the Level 2 system is the collection of all such fragments for an event. This includes both the information about the trigger decision as well as eta and phi data for the subsystems that cause an event trigger. We will refer to the collected ROI fragments for a given event as an ROI record, and to the subsystem on the ROIB card that builds the record, as the Assembly Unit (AU). The Input cards will pass ROI fragments to a set of ROIB cards. Each ROIB card communicates ROI records to four Supervisor processors. The compiled ROI records are transferred to the target

Supervisor processors using S-LINK (see figure 2). Each of the ROIB cards is responsible for a subset of the events that trigger Level 1.

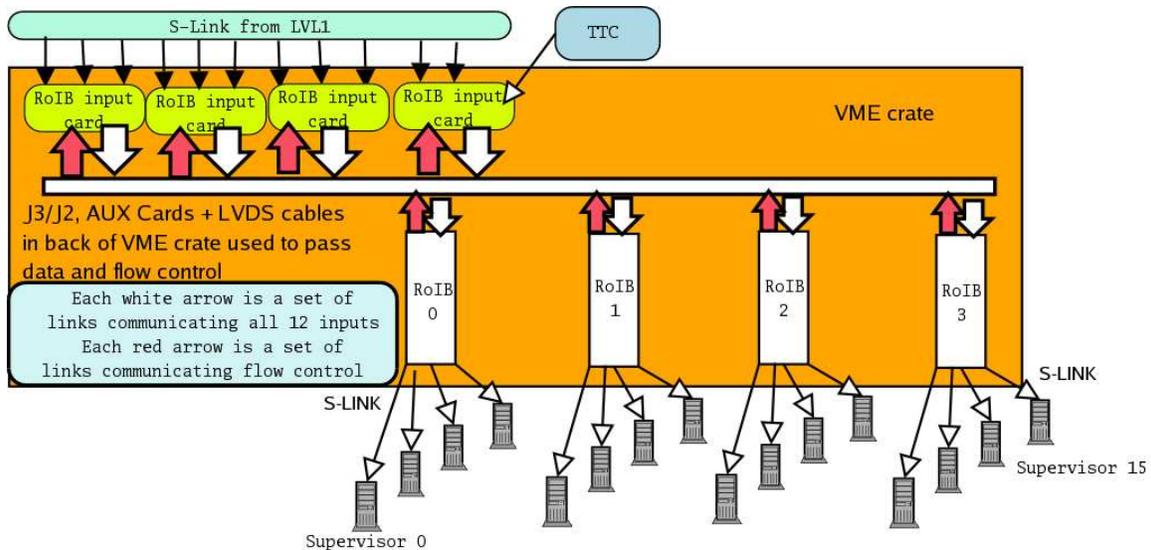


Figure 2: Diagram showing the ROIB and how the Level 1 data is expected to be distributed.

In the original prototype the system selected the target Supervisor processor strictly by a round robin algorithm. In this new prototype ROIB there will be a basic round robin algorithm, but we intend to make it considerably more sophisticated. In all cases, the system can skip target processors where flow control is active from the AU associated with the target processor. The system is expandable in units of four Supervisor processors by adding another ROIB card. The system will be able to accommodate an unlimited number of ROIB cards. Each ROIB card has registers which tell it which of the Level 1 channels are active, how many ROIB cards there are, which card it is, what variables to histogram, etc.

The event allocation algorithm must treat flow control properly and must deal with timeouts. A timeout may occur as the result of a tardy fragment or a missing fragment. The logic must distinguish the cases, and deal with either case. The events are allocated to Supervisor processors on a modified round robin basis, with the hardware dealing automatically with the number of cards, etc. If the S-LINK channel to a Supervisor processor is asserting flow control and flow control has backed up through the AU, the event may be allocated to another Supervisor processor served by the same card. The firmware also allows the allocation of events to specific AU's based on criteria other than the EL1ID (Extended Level 1 ID), for example the Event Type. Each ROIB card is more or less autonomous. Events are allocated to ROIB cards on the basis of $\text{Mod}(\text{EL1ID}, \# \text{ of cards})$. In every case the timeout system must interact with the event selection algorithm so that if a fragment is missing the problem is handled properly.

It is essential that the system be able to function in the presence of flow control. It is not easy to build records arriving from a multiplicity of sources when flow control is going

on and off from various elements, but it is important that the data integrity not be affected. There will be deep FIFO's on every input so that the peaks of data activity will be averaged, but will have flow control going back via the individual S-LINK channels to the Level 1 sources.

The individual ROI fragments can be as long as 128 S-LINK words including headers and trailers, and are in the S-LINK format (see the L1/L2Document). This length constraint is imposed by the available ESB resources in the 20K200E FPGA's which we are using. It is necessary to accommodate the time skew of arriving fragments, and accordingly a timer is started at the arrival of the first fragment of each event. If all the fragments have been received before the timeout the compiled record will be transferred immediately to the target Supervisor processor. If the timeout occurs first the system transfers an incomplete record to the target Supervisor processor. The timeout and other parameters are selectable via VMEBUS. The maximum value of timeout that the system can implement is a critical parameter. To the extent that a partially built ROI record has to wait for fragments the ROIB card must provide buffering so that other records can be built concurrently. The ROIB will accommodate a timeout as long as 1ms, but it should be understood that this places a severe strain on the hardware. Dealing with tardy fragments is a particularly vexing problem, and requires significant logic. If a fragment is lost, and an incomplete record is built, there will not be any later problems. However, if a fragment is tardy, an incomplete record is built and the tardy fragment subsequently arrives, it is necessary that the logic recognize this fragment as having belonged to a previous event. As far as the implementation in hardware is concerned, it is much easier to use local information to reject these tardy fragments. We handle this problem by having the local allocation algorithm logic retain the ELIID of the last complete record. The local logic knows if it has built an incomplete record, and how many incomplete records there have been since the last complete record, and so knows what ELIID subsequent fragments must have to be valid. If a tardy fragment from a previous incompletely built record is received it is discarded.

If a fragment is received without a header, that fragment is ignored and treated as missing. Since the allocation algorithm uses the ELIID as the basic data input, and the position in the frame of the fragment of the ELIID is determined in relation to the header, the allocation algorithm cannot reliably function, so the fragment is ignored. If the trailer is missing, the partial fragment will be built into the record, but it will be evident to the supervisor processor that the fragment had an error. The input buffer on the ROIB card can contain 128 S-LINK words. In the event that a Fragment is longer than 128 words the buffer will go full and the fragment will be truncated at the 128th word. This truncation will be evident to the Supervisor Processor, but will not affect the building of the relevant record.

3.3 Additional architectural details

This system consists of Input cards and ROIB cards as well as auxiliary cards (AUX cards) to simplify interconnection. Input and RoIB cards are VMEBUS 9u 400 mm, and compliant to the VMEBUS64X specifications. The large format card is necessary because the connections bringing data into and out of the cards require space, and because the logic itself implemented in 20K200E's must have adequate space.

VMEBUS64X is useful because it allows easy initialization of registers and memories, execution of diagnostic routines, block transfers of accumulated histogram data, monitoring of operation, etc.

All connections between the Input cards and ROIB are made via J3 and AUX Cards in the rear, so that there are no cables or fibers running between the front of the Input cards and the front of the ROIBs. We expect that an Input card can accommodate S-LINK inputs from 3 Level 1 Trigger Elements, and provide ROI Fragments to as many as 4 ROIB cards. Since one ROIB card services four Supervisor Processors, the basic system is limited to only 16 Supervisor Processors. If more than 16 Supervisor processors are desired the number of ROIBs served may be expanded in multiples of 4. This is accomplished by an expander card which we will develop as part of the system. This expander card will receive one of the outputs from an Input Card, and fan it out to four outputs, simply repeating on the four outputs the data stream received at the input.

The S-LINK LDC's are mounted on the front of the Input cards, and an Input card communicates with ROIB cards via J3 and an AUX card. An ROIB card can accommodate as many as 12 inputs one of which can be TTC information, and build as many as 4 records simultaneously to service as many as 4 Supervisor Processes. Information is carried from the Input cards to the ROIB cards via differential LVDS, using commercial molded cables and Mini D Ribbon Connectors (for example, 3M Corp.). It is necessary to parse the S-LINK words of the ROI fragment, and transfer the ROI fragment from the Input card AUX cards to the ROIB card AUX cards as 20 bit fragments on a 40 MHz clock. This allows the ROIB cards to build records from 12 sources with a 250 pin J3, although it is still necessary to bring flow control back via unallocated pins on J2.

Figures 3-8 indicate the general logic included in each card. The logic is implemented for the most part in Altera APEX 20K200E FPGA's.

Figure 3: Block diagram of the input card.

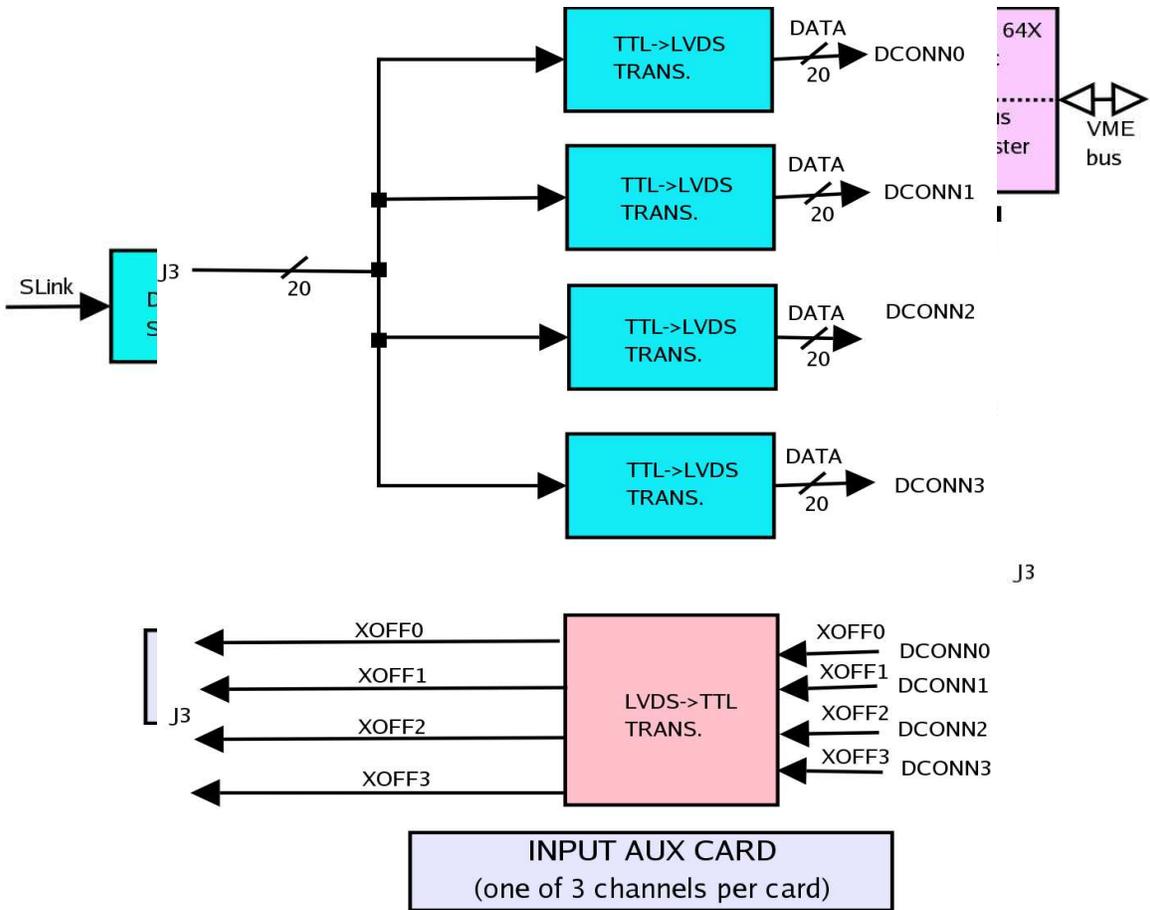


Figure 4: Block diagram of the input AUX card.

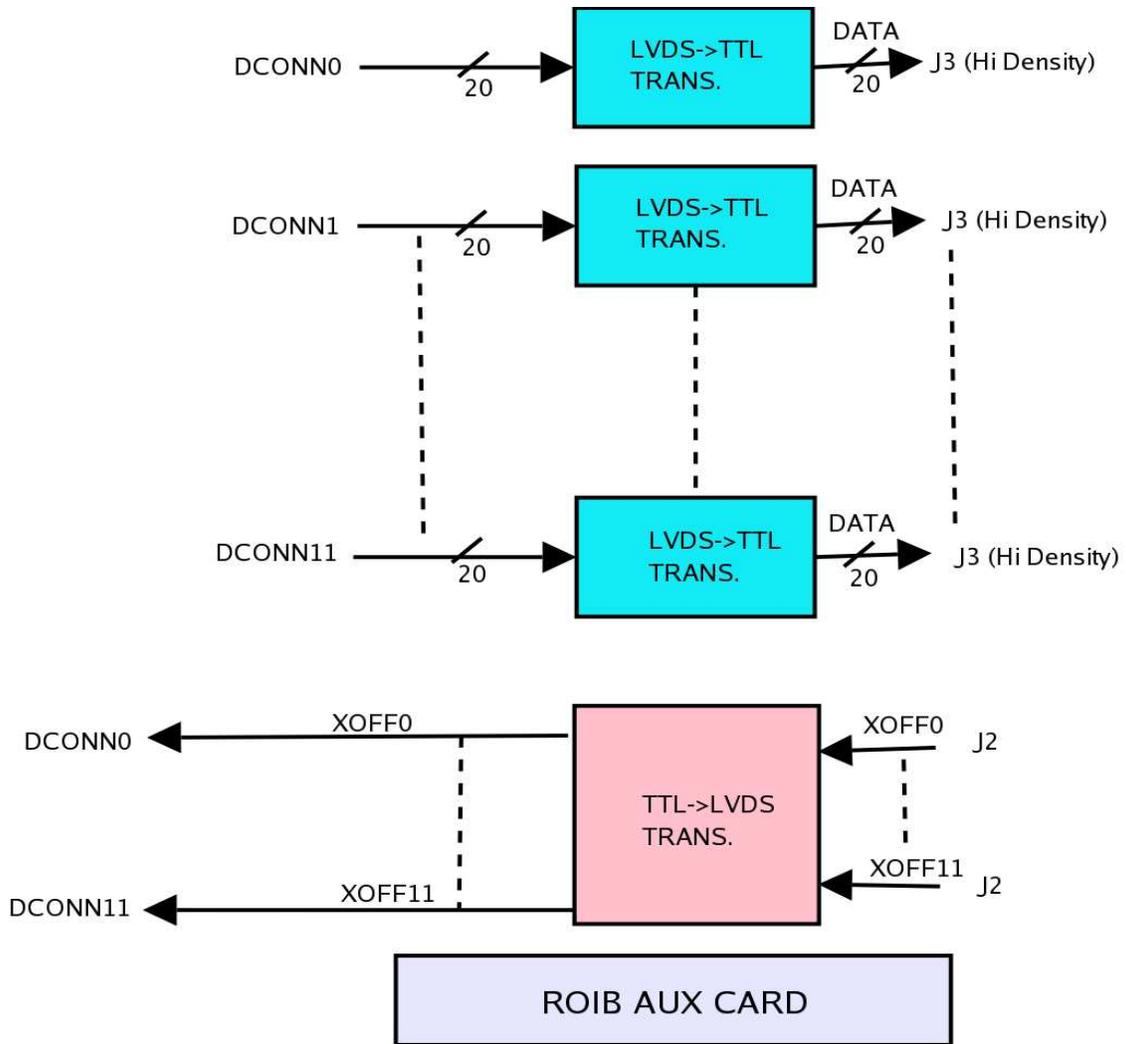


Figure 5: Block diagram of the RoIB AUX card.

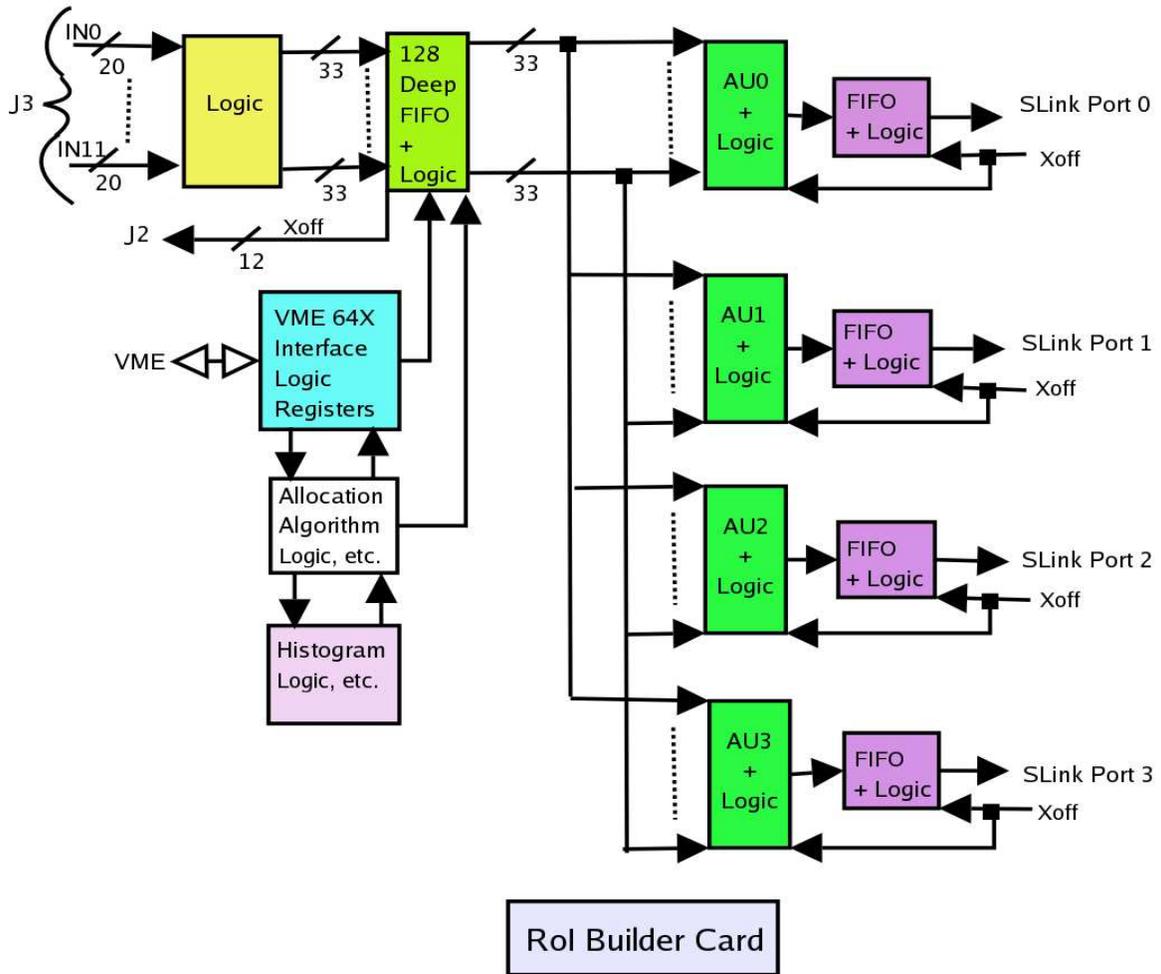
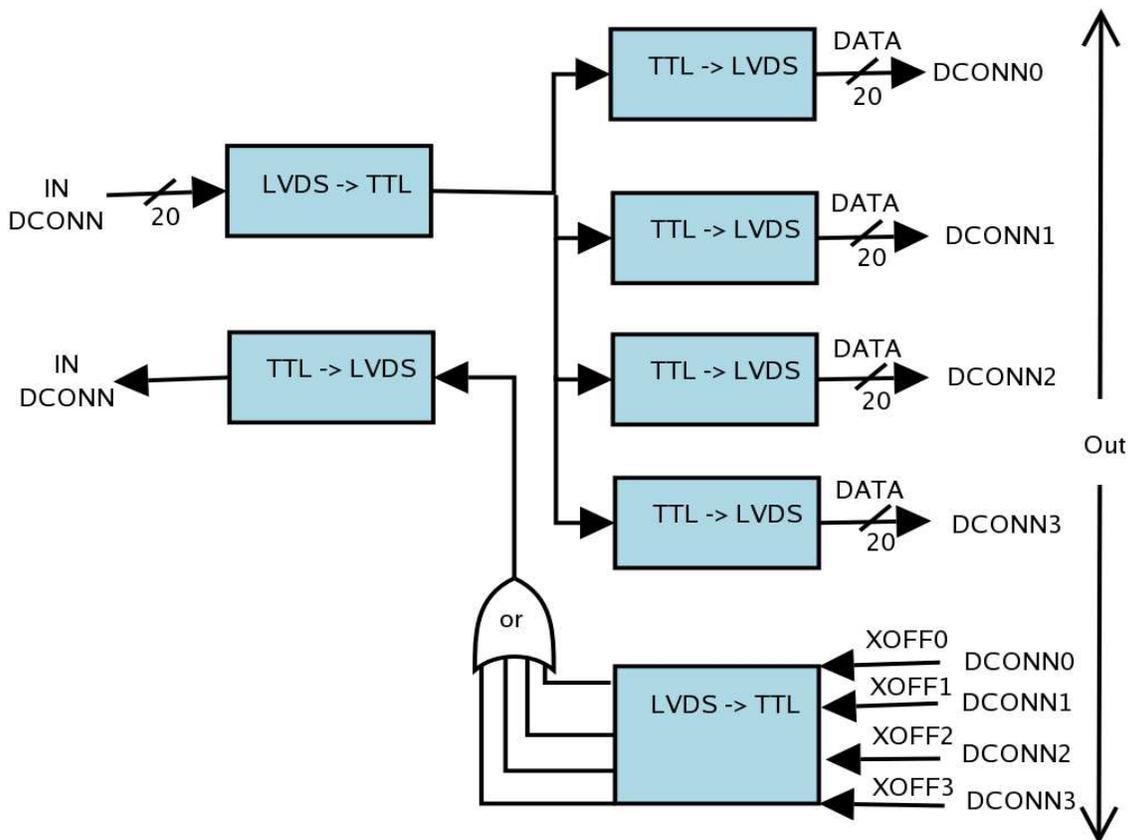


Figure 6: Block diagram of the RoIB card.



EXPANDER CARD
(one of 2 channels per card)

Figure 7: Block diagram of an expander card.



TTC MEZZANINE CARD
(receives TTC fiber and emulates S-LINK fragment source)

Figure 8: Block diagram of the TTC Mezzanine card.

The input of the Input cards is fully compliant with standard S-LINK. Each ROI Fragment must have a control word header with “b” in the top nibble, and a control word

trailer with “e” in the top nibble. This is the fragment format that has been agreed upon by Level 1 and Level 2. In the event that it is desired to have one or more data words precede the header and/or one or more data words follow the trailer this modification can be accommodated, and is relatively straightforward (the limit of 128 words for the fragment size still applies). When an ROI Fragment arrives via the input port it is written to a synchronous FIFO 8K words deep by the S-LINK input clock which we assume is 40 MHz although it could be something else. Flow Control which is transferred back to the Level 1 Trigger Element is derived from the Almost Full output on the input FIFO, and does not use Flow Control originating upstream.

Logic on the Input card keeps track of whether there is at least one complete fragment in the input FIFO and polls the FIFO Empty on the Input FIFO. When the Input FIFO becomes non-empty, if there is at least one complete fragment in the FIFO and Flow Control from the ROIB cards is not active, the logic reads one fragment from the FIFO. The ROI Fragment is transferred through J3 and the fanout on the AUX Card to as many as 4 ROIB cards (expandable), as 20 bit words on a 40 MHz clock. It is necessary to parse the S-LINK words in order to be able to service 12 inputs via J3, and even so it requires a high density J3.

The 20 bit halves of the S-LINK words are received as differential LVDS at the ROIB card AUX Cards, translated to single ended 3.3 volts logic levels, and transferred via J3 to the ROIB Module. The words are received by a 20K200E, and reassembled in a FIFO configured in the ESB, and buffered by this FIFO. The FIFO is 128 words deep, which says that an ROI Fragment can be 128 words long. When the trailer has been seen or the buffer FIFO is full the ROIB card sets Flow Control back to the Input card. Each ROIB card sees the ROI Fragment simultaneously and sets its Flow Control back to the Input card where Flow Control from all the active ROIB Modules are OR'ed.

Each ROIB card, when it sees the trailer of a fragment, starts a state machine which first determines from the EL1ID if this ROI Fragment is due to be built into a record on this card. The allocation algorithm expects to see a 32 bit EL1ID, however if only the lower 24 bits of the EL1ID are supplied the algorithm will not have a problem. If the answer is no, the state machine immediately clears the FIFO and releases Flow Control to the Input card. If the record is to be built on the ROIB card, the state machine on that card determines from the EL1ID which (of the four) to allocate the event, and the ROI Fragment is immediately shifted on a 40 MHz clock into the AU FIFO. When the AU FIFO sees the trailer, Flow Control to the Input card is released. Because Flow Control from all the active ROIB Modules is OR'ed, a new Fragment can now be transferred. The time involved for a Fragment of 60 S-LINK words, for example, would be 3 μ s to shift in, a few clocks for logic, and 1.5 μ s to shift into the AU, for a total of about 5 μ s.

When the first Fragment of a new record is transferred into the appropriate AU, a clock is started in the central logic. This clock continues incrementing, and the accumulated time is compared against a register. At the same time, when each AU FIFO receives its fragment it notifies the central logic, and the logic continuously compares the numbers of the AU's with fragments, against the Active Input Register. The amount of time required to time out is defined via VMEBUS. If the clock times out before the record is

completely built, the incomplete record is transferred to the target Supervisor Processor. If the tardy ROI Fragment is subsequently received, it is ignored. This actually involves a significant amount of logic, because a missing fragment may be tardy or lost. If it is lost there is no possible confusion with subsequent events, but a tardy fragment must be positively identified and discarded.

When all ROI Fragments for an event have been received and a complete ROI Record has been built, or the event has timed out and the Record has been built incompletely, the Record is transferred via a buffer FIFO an S-LINK LSC in the front panel of the ROIB card. The clock for the S-LINK transfer to the Supervisor processor may be either 40 MHz or 32 MHz, selectable via VMEBUS. Again, the output ports to the target Supervisor Processors are fully compliant with the S-LINK specifications.

3.3.1 Flow Control

Flow Control can become active at a number of points in the ROIB/Supervisor system, and it is critical that it be taken into account properly. Flow Control should be effective in dealing with situations where the event rate temporarily exceeds the maximum average rate that the system can accommodate. There are other times when Flow Control will become active, for example if a Supervisor Processor crashes. Where the event rate exceeds the maximum average rate, the ROIB can process for a period of time. Ultimately, the Flow Control system and corresponding back pressure will force trigger deadline thus losing events.

In the implementation of the ROIB-Supervisor link there are four FIFO's, probably 4K words deep, at the output of the ROIB card just before the transfer to the S-LINK LSC's, one for each S-LINK. This FIFO allows buffering of the records to the Supervisor processor, and avoids having to terminate data transmission of an ROI record if the Supervisor processor fails to service the port promptly. If flow control becomes active on the S-LINK to a Supervisor processor the ROIB card stops data transfer, but the record from the AU can continue writing into the FIFO to the end of the record. If at least one complete(or incomplete) record has been written to this output FIFO and flow control back from the Supervisor Processor continues to be active, the allocation algorithm will skip the AU corresponding to this Supervisor, and will instead allocate further Fragments to another AU on the card. This condition, for example, could exist if a Supervisor Processor crashed.

Within each ROIB card there are input buffer FIFO's 128 words deep on each of the 12 input data streams. These FIFO's receive the incoming ROI Fragments from the Input cards, and provide buffering. At the time the buffer FIFO sees the fragment trailer or goes full, the allocation algorithm determines if the fragment is an element of a record to be built on this card. If so, the algorithm then determines which of the four assembly units on the card should receive the fragment, and it is shifted to the appropriate assembly unit. In the event that the allocation algorithm determines that the fragment is not an element of a record to be built on this card, the buffer FIFO is cleared and the fragment is discarded. Logic associated with the buffer FIFO's sets flow control back to the Input card at the time the fragment trailer is seen in the buffer FIFO. If the algorithm determines that this is not the right card, flow control back to the Input card is

immediately cleared. If the algorithm determines this is the right card, flow control is cleared as soon as the fragment is shifted to the AU and the trailer has been seen in the AU. Flow control from all ROIB cards is OR'ed at the Input card.

If an AU has been selected by the allocation algorithm, it collects fragments with that ELIID until it has received a fragment from each active input. If a subset of these fragments is delayed the AU must wait. After a programmable period, if a subset of fragments is still missing the AU times out and the incomplete record must be transmitted to the Supervisor processor. If the tardy fragment or fragments are subsequently received they are discarded and not passed to the AU, so the logic must be aware that this record has already been built. If a fragment has been transferred from an input buffer FIFO to an AU, and the AU is waiting for fragments from other Level 1 trigger elements, this buffer can receive another fragment, but that fragment cannot be transferred into the AU until the record being built is transferred to the output FIFO.

If the Level 1 Accepts are happening at a rate of 100 KHz then the Input Card FIFO's which are 8K words deep can accommodate something like 1ms. of the data stream before becoming full. This implies that the system can wait approximately 1 millisecond for the tardy subset of fragments. If 1 millisecond is not enough time for slow fragments to arrive additional FIFO storage must be provided on the Input Cards. This also says something about the allowable variation in the latencies of the various Level 1 trigger elements. Of course the input FIFO's could be twice as deep and double the allowable amount of time. In any event, when an input FIFO is almost full it must raise Flow Control, and send Flow Control back through S-LINK to the Level 1 element. It is important that the ROIB logic requires records to be built such that all fragments have identical Level 1 ELIID's. For example, if the ROIB waits for a tardy fragment, times out, and subsequently sends the incomplete record to the Supervisor Processor, and then the tardy fragment arrives, the allocation algorithm must recognize that this fragment is not an element of a record to be built, but must be discarded.

Flow Control can be raised by an AU and this will be sensed by the allocation algorithm and that AU may be passed over for subsequent events. This might happen, for example, if the Supervisor Processor serviced by this AU were to crash. If the allocation algorithm cannot allocate an event to a AU (and hence a Supervisor Processor), it must suspend operation, i.e. Exert Flow Control, until an output is available. If Supervisor Processors are available, but complete records cannot be built, the AUs must wait for the tardy fragments. Timers can be started when the first fragment of an event is allocated to an AU, and a time can be set for the AU to time out and build an incomplete record, which can be tagged and sent on to the appropriate Supervisor Processor. If some subset of fragments is tardy and all AUs are waiting to build records, the input FIFO's will begin to fill. When they are almost full they will raise flow control on the links to the Level 1 Trigger elements.

It is important that if an output fails, for example if a Supervisor processor crashes, the system handles the problem with minimal loss. If an output S-LINK has Flow Control set, and an AU has built a record allocated to that output link, but has not been able to transfer that record because flow control is set, then fragments designated for subsequent

records which would otherwise be allocated to the jammed channel will be allocated to an alternate AU on the same card. This alternate AU could be chosen on a round-robin or other basis. In the event of a Supervisor crash, at least one record will be lost, but not more than two.

To summarize Flow Control, the logic always requires that when Flow Control is raised during the transmission of a fragment or record the whole fragment or record be transmitted, and then nothing further be transmitted until Flow Control is removed. Accordingly, if an ROI Record is allocated to a Supervisor no other ROI Record will be allocated to that supervisor until the first is transmitted, and if that Supervisor has crashed or that link is otherwise incapable of transmitting the record, that link will remain inactive. Flow Control can occur in several places, but generally speaking Flow Control in one place is independent of that in another.

Flow Control from the input card back to the Level 1 trigger element is raised by almost full in the input FIFO of the Input card, and is cleared by Almost Full becoming inactive. Flow Control from the ROIB cards back to the input cards is raised by the fragment trailer being seen in the ROIB card buffer FIFO, and is OR'ed for all the ROIB cards at the input card. It is cleared by the "wrong" ROIB cards immediately after the fragment trailer is seen in the buffer FIFO or the buffer FIFO goes full, and by the "right" ROIB card when the fragment is shifted to the relevant AU and the trailer is seen or the AU goes full (i.e. 128 words are transferred). Flow Control from an AU back to a buffer FIFO is raised by a fragment from that particular trigger element in the AU waiting for a record to be built, and is cleared immediately when the record is shifted to the output link. Flow control from an output FIFO to an AU is raised by the FIFO having a record which it cannot transfer to the S-LINK LSC because Flow Control is active on the S_LINK from the Supervisor. Flow Control from the output S-LINK to the output FIFO is raised by the LDC, and is cleared by the LDC.

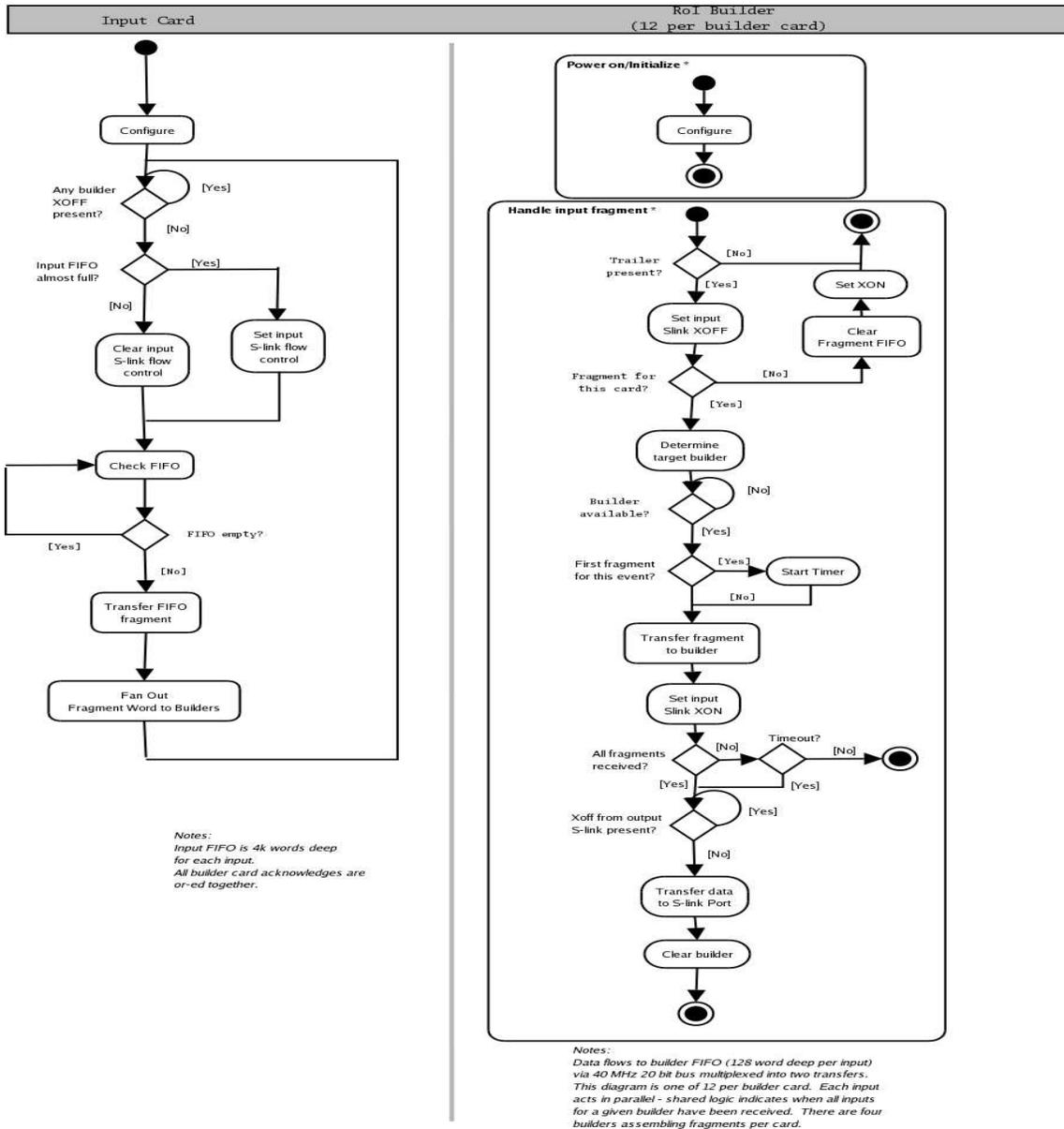


Figure 9: Diagram indicating the logic and flow control aspects of the ROIB system given a simple round robin style selection.

3.4 Simulation

A simulation based on Ptolemy II has been developed to reproduce the main aspects of the fragment handling. Ptomemy II and the Vergil GUI were used plus a few custom actors (one to simulate a resettable counter and one to simulate a finite depth FIFO). The simulation is not meant to represent the exact timing of the system, but to test the logical approach. Work is still being done to better represent the precise system details. A preliminary version shows that under normal input flows conditions the system works correctly.

The system developed using the standard actors available to the Vergil GUI is unduly complicated (figure 10 shows a test setup with two ROIB cards and an Input card, figure 11 shows the diagrams for the Input card and the ROIB card – note that there are several additional layers of composite modules which are not detailed in these diagrams). It is our intention to code the details into higher level custom actors (i.e., an Input Card, ROIB card and fragment generator) in java and simplify the model considerably. This will also facilitate review of the simulation since the Ptolemy II discrete event domain actors used are likely less transparent than simple java code. This will also facilitate generating error conditions to test the ability of the system to deal with common failure modes. An advantage of using the Ptolemy II framework will be the possibility of setting up an applet based web page that would allow interested parties to run and modify the simulation for themselves (see the various examples under the DE domain of Ptolemy II on <http://ptolemy.berkeley.edu>).

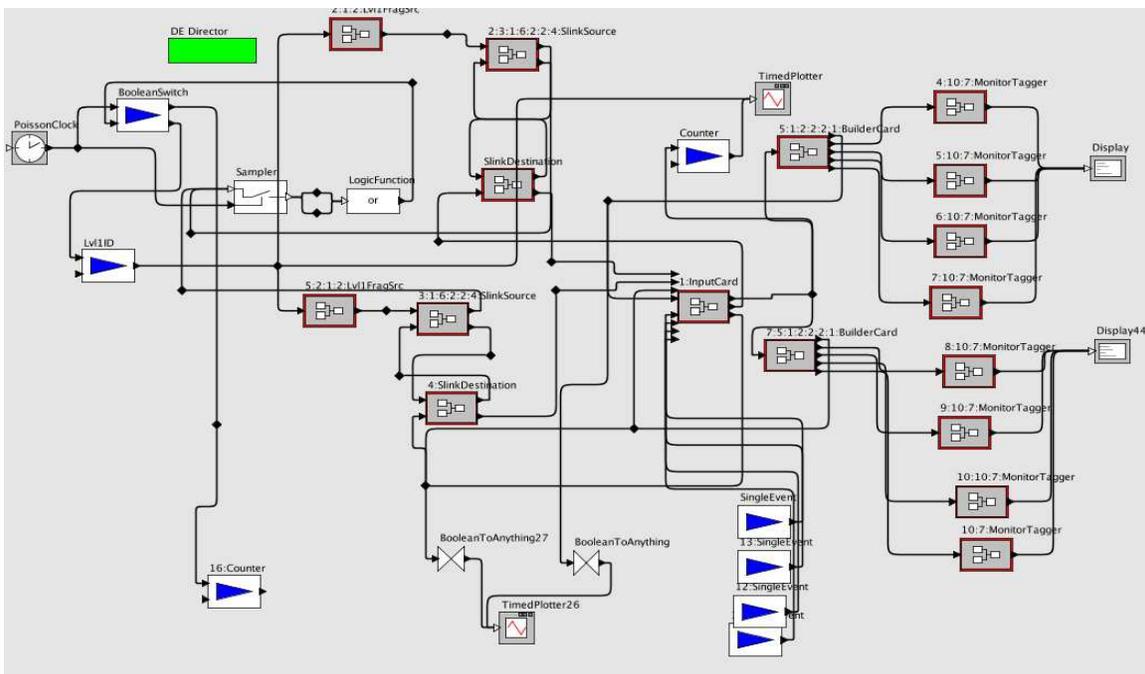


Figure 10: Test setup in Ptolemy. This test has eight outputs (to monitors/supervisors) and four level one input sources.

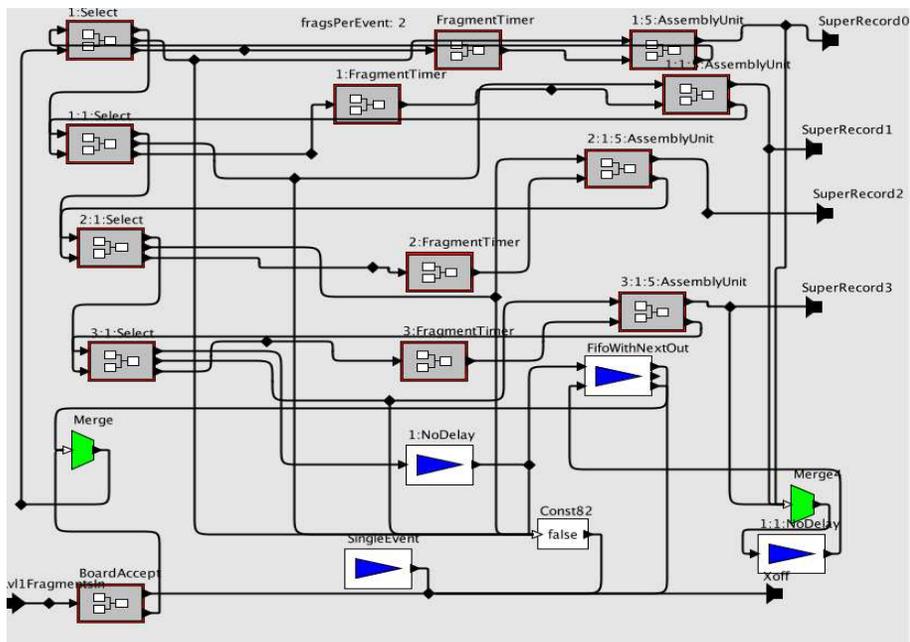
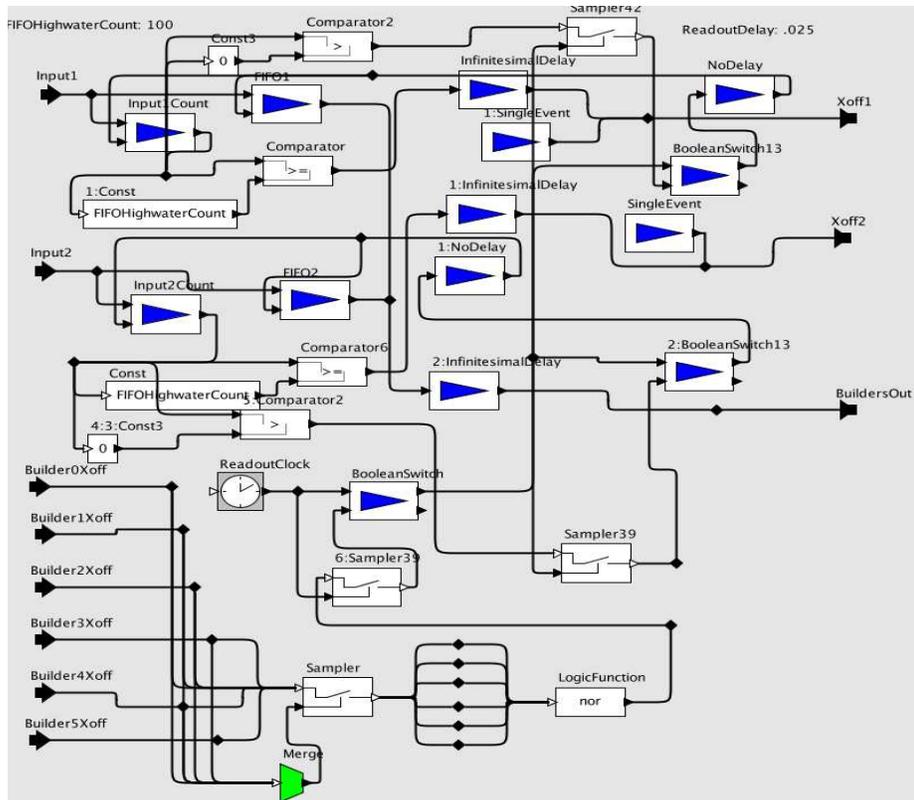


Figure 11: Input card (top) and RoIB card (bottom) as modeled in Ptolemy. This model has only two inputs per input card (increasing this to three as is expected in the final design will be done along with various other improvements to better represent the final design).

3.5 Error Handling

A number of exceptional conditions may arise while running and the system needs to be able to handle them gracefully and also provide useful feedback on what hardware if any is failing. Normal flow control, described above, will smooth out traffic bursts, but if the average Level 1 rate exceeds the system capacity flow control will exert back pressure on the Level 1 trigger elements and this in turn should throttle Level 1.

The link to the Level 1 subsystems is standard S-LINK and error handling on this link should be the same as that used throughout the rest of the experiment in the ROD to ROB interface. The link from the Input cards to the ROIB cards is differential LVDS and can be regarded as redundant. The normal redundancy provided by a link per fragment makes recovery possible since the ELIID of the event is carried by the fragments that are sent normally. For full event recovery the DAQ can use this to retrieve the missing fragment since it is stored in a ROB on an independent data path.

If data is corrupted on the S-LINK to the Supervisor. The Supervisor should initiate a recovery procedure. Here the ELIID is not carried on redundant links so the ROIB will have to provide the pending output ELIID via VMEBUS to the crate controller when the DAQ recovers the event. This can be achieved by having a register that can be interrogated for each output link. When an error is detected the Supervisor will have to assert XOFF until the ELIID is recovered via this VMEBUS recovery procedure.

The logic will be implemented in Altera 20K 200E FPGA's. The logic will be very dense and will be designed such that there is a large amount of resources unused in the FPGA's which allows for changes, modifications, and additional features to be implemented easily.

3.5.1 Monitoring

There are a number of quantities that should be monitored in order to anticipate errors and to evaluate the causes of malfunctions as well as to act as a cross check on other system components. An exhaustive list is not yet available but some items that should be available as histograms and as values that can be retrieved for each AU in the system shall include:

- fragment arrival times (. the initial fragment) for each input
- input corresponding to the first fragment for an event
- input corresponding to an out of order fragment (i.e. a fragment that is not the same ID as all other inputs for the Nth event)
- input corresponding to any fragments with BCIDs different from the other fragments of an event
- AU event counts
- AU current event and preceding 15 events
- flow control state/FIFO size for all queues in the system

Another cross check and a feature that would allow for more faithful simulation of

collider running is the addition of a TTC input as a 12th Level 1 component. We intend to bring the TTC information into the ROIB by providing a mezzanine format card which appears to one of the Input Cards as one more S-LINK LDC. In reality, this card is similar to a TTCPR module, in that it receives the TTC fiber, and has a TTCRX. However, the received TTC information is reformatted to resemble a ROI fragment, and transferred directly through the S-LINK port of the Input Card.

The ROIB will include an additional TTC input that will act as an additional Level 1 component. The TTC input can be used to emulate Level 1 by providing an input to the ROIB when Level 1 is not available. The TTC input can be used to verify that the EL1ID/ trigger was properly sent to readout components via TTC before the Supervisor sends the event to Level 2. This allows yet another cross check on the BCID corresponding to the trigger.

3.5 Manufacturing

Argonne plus outside suppliers, as needed, will provide the PCB's and component assembly.

3.6 Testing

The testing will be done in several stages. Communications tests will be performed at Argonne using existing PCs and adapters with software modified from that used to test the Gigabit Link Source Card. After rudimentary checks on the functionality binary tests will be scheduled with several Level 1 systems followed by tests with at least two such systems simultaneously. These tests will be similar to those performed with the previous pre-prototype ROIB. Unlike the previous tests the control functions for the ROIB will be integrated into the software framework and will conform to the current online framework for Level 2. Final tests should be performed with a vertical slice of the trigger including representative pieces of both the Level 2 and Level 1 trigger.

3.7 Installation

The system will consist of one or more 9U VMEBUS cards. Fiber connectors will be on the front panel. LEDs on the front panel will indicate media connection and activity.

3.8 Maintenance and Further Orders

This system is intended as a prototype and will not be in service for more than four years. Sufficient cards will be produced initially to accommodate any need for maintenance. Any future production may involve significant modification based on evaluation of the prototype performance and other desirable improvements resulting from advances in related technology. The US will maintain spares and provide support for the system as installed in the final ATLAS system.

4. Project Management

Currently the funding and project management are coordinated by the US ATLAS Project office at Brookhaven National Laboratory. Periodic reviews and monthly reports are coordinated by the project management team. The current US ATLAS TDAQ Level 2 manager is R. Blair.

4.1 Personnel

Customer

R. Blair
ANL
X7545

Project Engineer

J. Dawson
ANL
X7525

Software Professional

J. Schlereth
ANL
X6281

4.2 Milestones and Schedule

Test milestones are yet to be determined. This needs to be done in collaboration with the Level 1 groups. Initial card design and fabrication should be complete before Dec. 2003. A PDR occurred in Feb. 2002. Schematic level review will be done by Nov. 2003.

4.3 Costs and Reviews

All manufacturing, assembly and component costs will come under WBS 1.6 of the US ATLAS project management plan. Monthly progress will be reported via the US ATLAS reporting system. There will be a PDR prior to final design and a FDR prior to production.

4.4 Safety

General laboratory safety codes apply.

4.5 Environmental Impact

4.5.1 Disposal

ANL will dispose of cards at the end of their life.

4.5.2 EMC

Since the modules are prototypes they will be outside the scope of the EMC regulations. However, since the electronics must function as designed, without malfunction or unacceptable degradation of performance due to electromagnetic interference (EMI) within their intended operational environment, the electronics shall comply with specifications intended to ensure electromagnetic compatibility.

4.6 Handling Precautions

Anti-static precautions must be taken when handling the card to prevent damage to expensive components.