Feasibility studies of a Level-1 Tracking Trigger for ATLAS

I. Bizjak[a], R. Brenner[b], N. Konstantinidis[a], M. Sutton[c], M. Warren[a]

[a] Department of Physics and Astronomy, University College London, UK
[b] Department of Nuclear and Particle Physics, Uppsala University, Sweden
[c] Department of Physics and Astronomy, University of Sheffield, UK

warren@hep.ucl.ac.uk

Abstract

The existing ATLAS Level-1 trigger system is seriously challenged at the SLHC's higher luminosity. A hardware tracking trigger might be needed, but requires a detailed understanding of the detector. Simulation of high pile-up events, with various data-reduction techniques applied will be described. Two scenarios are envisaged: (a) regional readout - calorimeter and muon triggers are used to identify portions of the tracker; and (b) track-stub finding using special trigger layers. A proposed hardware system, including data reduction on the front-end ASICs, readout within a super-module and integrating regional triggering into all levels of the readout system, will be discussed.

I. Introduction

A tracking trigger is a relatively new proposal for the ATLAS upgrade, which already has a well established tracker project. A re-design would be ideal, but without a full physics study to support the case, and with viable possibilities to adapt the design as it stands, the additional effort and time required is likely too costly. To this end the work presented here has used the current state-of-the-art Pixel and Strip upgrade projects as a foundation. We have attempted to work within the architectural and technological constraints of the existing design. For most of the sub-systems we seek extensions of existing capabilities, but little in complete re-design. In areas less defined (e.g. most of the off-detector electronics), we use the current ATLAS SCT and Pixel topology as the baseline.

A track-trigger straddles two distinct components of ATLAS - detector (including readout) and trigger. These two groups have agreed parameters in-which to operate (trigger rates, latency etc.) and we attempt to retain these where possible.

Various options for tracking readout exist, falling into 3 areas:

1) Bunch-crossing (BC) rate readout of the whole detector. This increases data-volume/bandwidth by a factor of order 300, and is deemed infeasible.

2) Auto/local event selection with special layers. On-detector logic selects good track-stubs autonomous of any triggers, which are "pushed" out as needed. In the case of Strips, both sides of a module will be connected to each other. These connections could be at the chip, module, or super-module level, with increasing bandwidth requirements respectively. Early studies show that high readout rates are required as it is difficult to distinguish between low- and high-pT tracks (influenced by the magnetic field). Options for on-detector track-finding are also being investigated, although this require grouping data from modules spread over multiple layers/discs with difficult readout challenges. These ideas are in their infancy and not covered in this paper.

3) Readout only regions of the detector prior to an L1A being issued, making use of seeding from early stages of the L1 trigger system. This is the focus of this paper.

II. Regional Readout

Regional readout uses L1Calo and L1Muon to identify potentially interesting features at a few hundred kHz. They issue fast readout requests to specific regions in the tracker at this rate, providing the (η, ϕ) position of the objects identified as interesting. In this way, only a small fraction of the detector is read out, and only at a reduced rate such that the required additional bandwidth will be modest.

Several variations are possible with this approach, depending on how fast the regional detector data can be read out and processed, and on the overall Level-1 Trigger latency envelope. Ideally, tracking information should be used directly within the Level-1 Trigger. However, ATLAS has also discussed an option for a two-stage Level-1 trigger, for use if the Inner Detector readout is too slow. This would require additional buffers on all ATLAS detector front-end ASICs (FEICs), in which data would be held until the slower, definitive hardware trigger decision is available.

A. Regional Readout System Overview

The track-trigger builds on the existing Level-1 Trigger architecture, in which a potentially interesting event is identified, and a signal synchronous with that event is sent to the detector front-end (FE) modules. The FEICs transfer the event data from their pipelines to a readout buffer where the data are queued until they can be transferred off-detector.

For regional data readout, the process has two important differences:

- The trigger in this case is a regional-readout-request (R3) which is not broadcast to all FE modules. Instead it is sent only to the Inner Detector modules that fall within the region-of-interest.

- Readout is minimally buffered - when an FEIC receives an R3, it must return the data as fast as possible employing prioritised multiplexing or a separate data path.

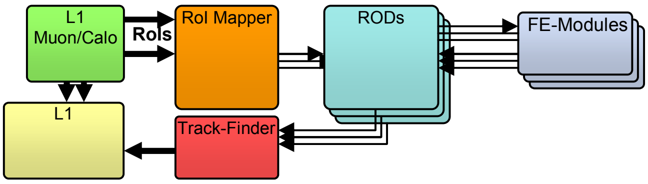

Figure 1: Conceptual Regional Readout System within ATLAS DAQ.

Figure 1 shows the system layout. The track-trigger process begins with the receipt of one or several RoIs from the L1Calo or L1Muon system by the RoI mapping hardware. The information is decoded and synchronised, generating readout requests to be sent to the modules within the RoI. At this stage the physical geometry of the detector can be used to send targeted RoI/R3 signals to the Readout Drivers (RODs) which map and forward them to the desired super-module.

The Super-Module Controller (SMC) ASIC (that resides at the edge of a super-module) decodes the signal for inclusion in the trigger, timing and control (TTC) signal distribution, using special lines/protocol to identify which modules should be read out.

The FE modules comprise a Module Control Chip (MCC) and many FEICs. Upon receiving an R3, the MCC prepares for readout of track-trigger data while forwarding the R3 signal to the FEICs, which copy the raw-event from mid-pipeline, process and insert it at the front of any queues. The data are then sent off-detector on a prioritised channel.

In the case of dedicated track-trigger links, these data would travel directly to the Track-Trigger Processor (TTP). It is more likely, though, that track-trigger data will be multiplexed with normal event data on the same links and be intercepted on the ROD for forwarding to the TTP.

B. Rates and Expectations

Some estimates need to be made of the trigger rates we expect. We presume that the Level-1 rate remains at 100kHz, and the R3 rate somewhere between the bunch-crossing and L1 rates at 400-500kHz.

The detector will likely contain of order 4000 RoIs. Guesstimate from current detector expectations indicate that an RoI encompasses ~1% of modules on the detector, and that ~4 RoI are expected per event [1].

Figure **2** shows pictorially the scale of an RoI.

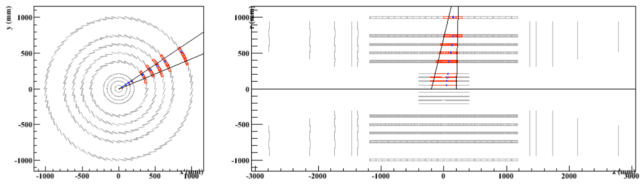

Figure **2**: Event display showing RoI geometry (RoI: $\Delta\varphi$=0.2, $\Delta\eta$=0.2 at Calo Δz=40cm at beam line).

III. IMPLEMENTATION

Incorporating a track-trigger, particularly as part of the Level-1 Trigger into the ATLAS upgrade involves changes to many sub-detectors and almost all sub-systems of the inner-detector. As the overall architecture of the detector is affected, and will need to be re-evaluated, the constraints and requirements need to be examined:

- Trigger latency – Latency affects almost all aspects of the design, but in terms of trigger it defines the FE pipeline length – longer pipelines need more resources.

- Data volume – Bandwidth affects readout rate, dead-time and latency.

- Data transfer and synchronisation – Transferring different data types with differing constraints is difficult.

- Regional-readout-request distribution – Targeted R3s need more infrastructure.

- Off-detector readout and track-finder – This is a new sub-system where a fast and synchronous path to Level-1 Trigger is required.

A. Overall Latency

FEICs have finite pipelines, defining the Level-1 trigger latency. The current ATLAS has a maximum latency of ~3.2µs (128 BC). The upgrade already prefers more (6.4µs is a common assumption) [2], but this needs to be evaluated against cost and complexity – in both new hardware and increased power.

Much of the trigger latency is consumed by cable lengths between the counting room and the detector – a round-trip time is 1µs. As the track-trigger system needs to readout the RoI data prior to a level-l decision an additional 1µs round trip is required. The track-finding efficiency increases with processing time. An initial estimate, based on D0 [3] indicates a minimum of 2µs.

Initial estimates:

BC → RoI	1200ns + 500ns fibre
Decode RoI/R3	650ns + 500ns fibre
Data Volume	2375ns
Readout	325ns + 500ns fibre
Track-Finder + L1	2000ns + N + 500ns fibre
Total	**8550ns + N**

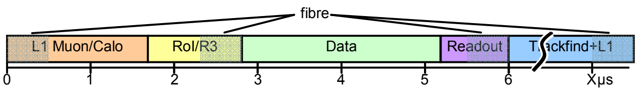

Figure 3: Chart showing contributions to latency.

B. *Data Volume and Dead-time*

Event data is the largest contributor to latency on-detector. Although queuing regional data in the FEICs would only slightly increase latency due to the low R3 rate per module, the peak latency would be much higher. It follows, therefore, that a module cannot accept a second R3 while busy with readout of the previous, and data-volume equates to dead-time.

To reduce data-volume (and latency) data compression on the FE module is desirable. For track-finding not all hit data is useful - in general, if a module (or FEIC) has too many hits, or wide clusters, there will be little opportunity of a track-finder to identify un-ambiguous tracks.

To effect this, simulations have been carried out where the cluster width is restricted to <3 strips and the number of clusters per FEIC and per MCC are capped. Using SLHC-like events (400 pile-up) it can be shown that <5% of high-pT track derived hits are lost [1]. See Figure 4.

To further reduce data-volume, only the first strip of 2 strip hits are used. Combining the low number of hits with known hit-count maxima allows for an efficient packing algorithm that will improve further with larger (more strip channel) chips.

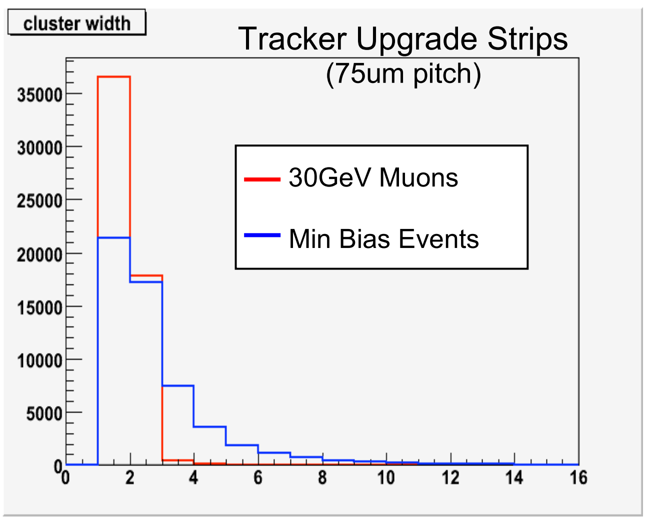

Figure 4: Plot showing cluster width differences between higher pT and min-bias events.

C. *Data Transfer and Synchronisation*

Ideally regional data would have a dedicated path off-detector allowing for fixed latency and no congestion. This introduces many new readout paths, and could double the number optical links between the detector and counting room. This is obviously not desirable.

Sharing a readout "channel" with event-data makes sense (especially when considering the low data-volume), but this both de-synchronises the data and increases latency: Event data will most-likely be transferred in packets [4] with headers, trailers, bunch-crossing IDs, event IDs, chip IDs etc. A packet might be broken into frames allowing it to be transferred non-continuously. Regional data will need to wait for any in-progress packets or frames to finish transferring before initiating readout.

Smaller frames will have less impact on regional-data synchronisation, but will also decreases data-volume efficiency. Ideally a frame of the order 10 bits would be a compromise worth investigating: 1 start bit, 1 normal/regional event select bit, and 8 bits data.

D. *Off-detector Readout*

Data from the detector are transferred, via optical links, to RODs in the counting-room. Regional data does not need to be processed by the ROD in a significant way. Here the ROD acts as a router diverting the incoming data out to the track-finder hardware.

As track-trigger data-volume is low, the number of links to the track-finder can be optimised and data concentrated (although queuing during times of peak volume needs to be taken into account). Tags will need to be added to the data to identify which link (or module ID) it belongs to. As the data will arrive relatively slowly from the front-end (a single optical link is shared by 12 modules) it might be fragmented when sent to the track-finder and require more tagging. The additional latency incurred while queuing can be reduced, on average, by prioritising older data (i.e. that with earlier bunch-crossing IDs).

Detector layout plays a part in level readout latency too. As an RoI will encompass adjacent super-modules, data should be routed to different RODs. For example, in the barrel, only every 3rd super-module, radially, should be connected to the same ROD.

E. *Track-Finder*

Due to the distinct differences in layout between barrel and end-cap, the track-finder will have optimised configurations divided geographically along the length of the detector: barrel, end-cap and both. The detector will also be divided into quadrants, with overlap. This motivates independent track-finder units servicing the 24 zones.

To allow for asynchronous data, the track-finder unit will assign a processor to an individual event (BCID). Incoming data from the RODs will need to be routed first to its' zonal unit (and duplicated in the case of overlaps) and then routed to the processor assigned to that event.

The processor is expected to operate using a "bingo" technique – as data arrives it is used and if tracks are found they are logged. This means tracks can be found even with incomplete data-sets.

By determining a processing cut-off time synchronous to the event being processed, all tracks found can be passed to the next stage of the trigger system synchronously if needed, with outstanding data discarded.

F. Regional-Readout-Request Distribution

The regional-readout-request signal operates similarly to the L1-Accept (L1A) signal – it is synchronous to the BC it acts on, used to copy data from the front-end pipelines, generated by the Level-1 Trigger, and is desired to be low-latency.

However, unlike the L1A, the R3 is not broadcast, but instead targeted at specific modules. There are of the order 50000 modules in the tracker alone, so this is a large-scale system.

With ~4000 RoIs it will be most efficient to distribute RoI-IDs as opposed to R3 signal where possible. Using CERN Giga-Bit Transceivers (GBTs) in the counting room, we can distribute 6 to 10 RoIs/BC [5], allowing RoIs to be broadcast to all ROD-Crates (containing ~10 RODs each) via the TIM, or directly to each ROD (of order 200 in the SCT+Pixels).

Each ROD identifies which of its connected super-modules are inside the RoI and generates an R3 map for these modules. This requires a custom look-up table on each ROD which will need uploading at configuration.

The R3 signals are transferred using a special GBT word to the super-module where the SMC decodes the signal and forwards to the modules.

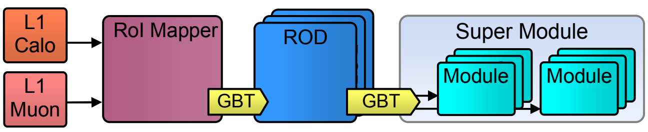

Figure 5: Schematic of R3 generation and distribution system.

As each module needs to be identified individually, point-to-point links between the SMC and the module would be ideal, but resources on-detector are limited. Sending the signal serially (at 40Mb/s) is slow and introduces latency (300-600ns). Latency can obviously be improved by broadcasting at higher rates.

A compromise between signalling and latency on-detector would be to split the super-modules into 'zones' allowing simultaneous short bitmaps to be sent to each group of modules.

Other options include broadcasting just the central module ID and let the modules decide if they are inside the RoI or not.

IV. Conclusion

Although a track-trigger has only recently been applied to the ATLAS upgrade design, options have been found for its incorporation. There are many outstanding issues, not least of which is the latency requirement, but all of the sub-systems involved seem capable of the modifications required.

V. References

[1] ATLAS Track-trigger webpage, https://twiki.cern.ch/twiki/bin/view/Atlas/L1TrackTrigger
[2] N. Gee, private communication
[3] H. Evans, meeting slides http://indico.cern.ch/getFile.py/access?contribId=5&resId=0&materialId=slides&confId=65243
[4] Strips Readout Working Group, private communication
[5] CERN GBT Project, http://cern.ch/proj-gbt