ATLAS PUB Note
ATL-DAQ-PUB-2022-001
10th December 2022

FELIX Integrating the GBTx ASIC links into standard networks – the original proposal

Lorne Levinson[a], Niko Neufeld[b], Jos Vermeulen[c]

[a]*Weizmann Institute of Science, Rehovot, Israel*
[b]*CERN, Geneva, Switzerland*
[c]*Nikhef, Amsterdam, Netherlands*

This document reproduces the original August 2012 proposal for the device that became the FELIX (Front End LInk eXchange) element in the ATLAS Data Acquisition system.

This document reproduces, on the following pages, the original August 2012 proposal for the device that became the FELIX (Front End LInk eXchange) element in the ATLAS Data Acquisition system.

The Front End Link eXchange (FELIX) [1–5], developed by the ATLAS Trigger and DAQ project, replaces the previous ATLAS Front End readout architecture. FELIX separates data transport from data processing: data are transported by FELIX, a detector neutral custom hardware/software device; data are processed by detector-specific software running on servers. FELIX interfaces multi-gigabit per second optical links, from ASIC's that aggregate several slow serial copper "E-links" (GBTx [6, 7], lpGBT [8]) or from FPGA's, to an industry standard Ethernet network. These links carry readout, calibration and detector monitoring data from the Front End, and configuration and control data to the Front End. Acting similarly to a network switch, FELIX routes these links individually between Front End electronics and the relevant software processes on the network. It also distributes the TTC (Timing, Trigger and Control) signals [9], including the LHC Bunch Crossing clock, to the Front End electronics via dedicated E-links. FELIX provides a common platform for some ATLAS LHC Run 3 subsystems and will do so for all ATLAS LHC Run 4 subsystems. FELIX is built from custom PCIe FPGA cards [4] hosted in commercial Linux servers, each equipped with a high-performance Ethernet interface card. The use of commodity components and the sharing of a common platform by all sub-detectors reduces hardware, firmware and software effort.

An addition reference to the cited GLIB board may be found here [10].

References

[1] J. G. Panduro Vazquez, *FELIX and the SW ROD: commissioning the new detector interface for the ATLAS trigger and readout system*, PoS **EPS-HEP2021** (2022) 825 (cit. on p. 2).

[2] A. Paramonov, *FELIX: the Detector Interface for the ATLAS Experiment at CERN*, EPJ Web Conf. **251** (2021) 04006 (cit. on p. 2).

[3] J. G. Panduro Vazquez, *FELIX: the new detector interface for ATLAS*, EPJ Web Conf. **245** (2020) 01037, ed. by C. Doglioni et al. (cit. on p. 2).

[4] K. Chen et al., *A Generic High Bandwidth Data Acquisition Card for Physics Experiments*, IEEE Transactions on Instrumentation and Measurement **69** (2020) 4569 (cit. on p. 2).

[5] M. Trovato, *FELIX: The New Readout System for the ATLAS Detector*, 2019 IEEE Nuclear Science Symposium (NSS) and Medical Imaging Conference (MIC) (2019) 1 (cit. on p. 2).

[6] P. Moreira et al., *The GBT Project*, Topical Workshop on Electronics for Particle Physics (2009), URL: http://cds.cern.ch/record/1235836/files/p342.pdf (cit. on p. 2).

[7] K. Wyllie et al., *A Gigabit Transceiver for Data Transmission in Future High Energy Physics Experiments*, Phys. Procedia **37** (2012) 1561, ed. by T. Liu, URL: https://www.sciencedirect.com/science/article/pii/S1875389212018706?via%3Dihub (cit. on p. 2).

[8] CERN lpGBT Team, *The lpGBT: a radiation tolerant ASIC for Data, Timing, Trigger and Control Applications in HL-LHC*, (2019), URL: https://indico.cern.ch/event/799025/contributions/3486153/attachments/1901196/3138449/lpGBT20190903.pdf (cit. on p. 2).

[9] P. Gallno, *Modules development for the TTC system*, 5th Workshop on Electronics for the LHC Experiments (LEB 99) (1999) 327, URL: https://cds.cern.ch/record/433828/files/p327.pdf (cit. on p. 2).

[10] P. Vichoudis et al., *The Gigabit Link Interface Board (GLIB), a flexible system for the evaluation and use of GBT-based optical links*, JINST **5** (2010) C11007 (cit. on p. 2).

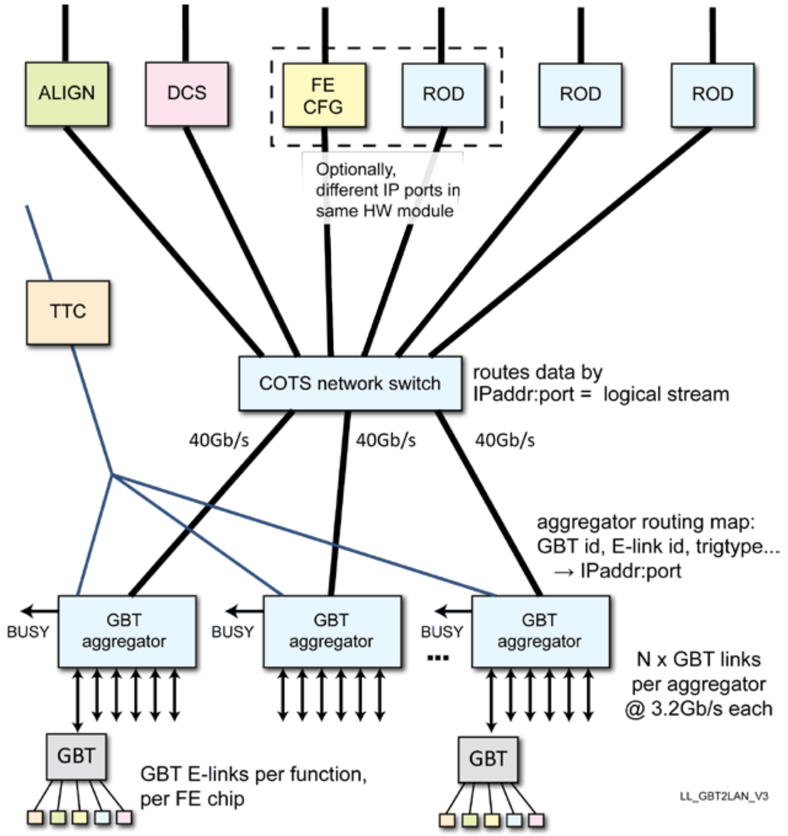
Integrating the GBT into standard networks – a proposal

Lorne Levinson (ATLAS), Weizmann Institute of Science
Niko Neufeld (LHCb), CERN
Jos Vermeulen (ATLAS), NIKHEF

August 2012

The CERN-developed GBT chipset has the advantage of allowing many logical data paths to be combined on a single bidirectional link. Example paths are the event data, DCS, TTC, configuration and alignment. This reduces the complexity of connections to the on-detector electronics. A disadvantage is that the different logical streams end up at a single off-detector point. This point is then required to understand which data belongs to which off-detector entity and to route it accordingly. Such an architecture encourages the bad practice of combining all the logically separate services into one off-detector hardware device, e.g. the ROD. It also means that the component with the highest availability and reliability requirements (probably DCS) forces its requirements on the off-detector end-point.

A generic GBT aggregator as described here solves these problems by aggregating several GBT links onto a higher bandwidth, industry standard network technology, e.g. Ethernet or Infiniband, from where standard network switches and protocols can be used to route data to and from the appropriate end-points. A well-defined device can be made to satisfy the requirements of many detectors and experiments, to be highly reliable and to be independent of experiment states and data formats. By separating the GBT implementation, it allows the end-point builder to concentrate more on the end-point functionality. The scheme is shown in the figure below:


ALIGN
DCS
FE CFG
ROD
ROD
ROD
Optionally, different IP ports in same HW module
TTC
COTS network switch
routes data by IPaddr:port = logical stream
40Gb/s
40Gb/s
40Gb/s
aggregator routing map: GBT id, E-link id, trigtype... → IPaddr:port
BUSY
GBT aggregator
BUSY
GBT aggregator
BUSY
...
GBT aggregator
N x GBT links per aggregator @ 3.2Gb/s each
GBT
GBT E-links per function, per FE chip
GBT
LL_GBT2LAN_V3

Some characteristics of the aggregator:

1. The LAN can be either Ethernet (10G, 40G…) or Infiniband (40G, 56G…) or both.
2. The aggregator has a configurable routing table that maps GBT semantics to network semantics, for example:
   GBT ID/E-link → Ethernet IP-address /port, or
   → Infiniband local identifier /queue pair
3. The aggregator includes a TTC interface in order to inject TTC info into each GBT link. A busy output level would be asserted if any of the logical channels' buffers were near overflow or if an off-detector endpoint so requested.
4. The aggregator has no internal states that depend on the state of the experiment's DAQ.
5. One can connect any GBT to any port of any aggregator in the LAN (with appropriate configuration of the aggregator).
6. Ideally GBTs have a unique id (i.e. a "MAC" address); otherwise aggregator#/connector# must be used to identify the GBT.
7. A very clear specification of its limited mission allows the aggregator to be adiabatically upgradeable:
   - Newer FPGA
   - Faster GBT
   - Faster COTS network link
   - Additional routing options
8. Quality of Service can be configurable for each logical connection:
   - Dedicated bandwidth for event data
   - "Less-than-best-effort" for data for monitoring
9. Broadcast to all GBTs can be supported.

Such an aggregator connected to a COTS network opens up many possibilities for the off-detector architecture:

- The off-detector endpoints need not implement GBT hardware interfaces. This means that they may not need to include FPGAs.
- The off-detector endpoints are free to be implemented as software on computers or as dedicated custom hardware (e.g. FPGAs). They may be dedicated computers in many formats (ATCA, VME, rack PCs), virtual machines, processes in a computer, or as threads in processes.
- Rerouting data from a failed off-detector endpoint to a spare or adding off-detector endpoints in order to share the load is easily done by reconfiguring the routing tables in the aggregator.
- Off-detector endpoints do not need to be mapped one-to-one to the geographical areas serviced by a GBT. A single off-detector endpoint may for example collect data from a complete ring of constant radius from many GBTs that each read out radially within a sector.

Two possible implementations are:

1. A complete FPGA implementation with GBT links, routing tables and Ethernet and/or Infiniband output, all in the FPGA. This may not be so easy for Infiniband. It is also more difficult to upgrade to a faster network interface. The GLIB board can be a platform for prototyping (https://espace.cern.ch/project-GBLIB/public/).

2. GBT links in a PCIe FPGA board. Data is shared with a host processor that does the routing and interfaces with a COTS Ethernet and/or an Infiniband PCIe network interface card. This option is slightly less compact, but more easily maintained and upgraded.

An aggregator as described here is not suitable for data paths requiring low or constant latency.