

Reviewers Comments, Questions, and Answers

12/06/2012 11:22

Comments and questions by the reviewers are listed here. Where possible, questions will be answered. In quite a few cases a simple answer won't be appropriate and the flagged issues should be discussed in the review session. Preferences by the designers might be highlighted in the text in green colour.

This document will be expanded as comments come in.

=====
Architecture issues:

--

Input links: In the documentation, it appears that there is room for up to fourteen 12-fiber receivers. This translates to 168 input fibers. But it is stated in the documentation that the FPGAs in the final design may have up to ~80 MGTs each (up to total 160 input links), while those for the prototype will have 56 MGTs (up to 112 input links). This leads to a couple of issues:

1. For the production version, how do we distribute the unassigned 8 fibers (168-160) in the best possible way? Do we have **13 fully occupied receivers, plus one with four fibers**, or multiple partially-used receivers? This probably can't be decided in the review, but we should look ahead towards how we want to do fiber mapping from the various systems in Phase 2.

tbd

2. How many MiniPods will the prototype support? For simplicity, it has been proposed to send a full 12-fiber ribbon from each CMX to L1Topo in 2014, not to mention the possible addition of L1Muon. But 2x56 MGTs falls well short of the bandwidth necessary for 12 L1Calo CMXs, so if we wanted the prototype to be usable one would need 13-14 partially-utilized minipods. Alternatively, we could say that the prototype is a reduced-channel-count module for development purposes only.

We were planning to do a total of 3 prototype modules based on the smaller chip. In fact the very first one would need to be 7V485T due to component availability. However, for any further modules that is still open for discussion. Alternatively for 2nd copy, 690T might be possible, though with engineering samples rather than production silicon. Will try to find out about lead time...

--

G-link readout to DAQ and ROI: The specification includes one G-link type readout each to DAQ and the RoI builder, utilizing L1Calo RODs. Leaving out the decision to use L1Calo RODs (which I will bring up again below), the input and output bandwidth of the proposed L1Topo is too large to be read out by a single Glink. At 100 kHz L1A rate, the maximum readout bandwidth is 20 bits data width times 399 bunch crossings, or 7980 bits. This corresponds to the data payload for only about 62 fiber links running at 6.4 Gbit/s. So to read out up to 160

input links up to 10 Gbit/s plus outputs, BCnum and checksums requires either more/faster readout links and/or readout formats with compression/truncation.

We are planning for spare miniPOD connected to the control FPGA. That could handle additional channels. However, if we think we'll require more than one DAQ and ROI channel anyway, we could drop the proposed SFP devices and go for miniPOD right away. A battery of SFPs on the front panel doesn't seem sensible. It should be pointed out that the data need to get into the control FPGA first. Therefore we might need to think about MGT output right from the processor FPGAs...

--

SystemACE: Xilinx announced in late 2011 that SystemAce is obsolete and will be discontinued. has been rendered obsolete (in favor of Platform Flash). For a system that we may want/need to upgrade or expand through at least 2017 (and possibly into Phase 2), I am nervous about committing to an already-deprecated device. Platform Flash has some attractive features; it is a read/writable NOR flash memory with a simple peripheral bus interface (for e.g. interfacing with a local microprocessor) that can directly configure FPGAs without the need for glue logic.

Unfortunately PlatformFlash XL seems to offer 128Mbit per device only (maybe someone else knows more on that). So we would not be talking about just a single chip then. If we want to be able to store more than one configuration concurrently we'd rather need of the order of 10 of them. We are currently experiencing horribly long write sequences for even moderate bitfile sizes on industry standard flash devices and would like to avoid that on future modules.

--

Module control: The specification calls for the module to be initially controlled by a serially extended VME bus and then later by ethernet. But little is written about the high-level protocol or the device that will handle the interface. I would very much like to consider having a local microprocessor on board that can provide a ready-made TCP/IP interface. There are a number of inexpensive SOC devices that support Linux, and an embedded system would provide both a convenient way to provide control and diagnostics remotely (i.e. down in the pit) instead of requiring a USB connection, as well as a good interface to PlatformFlash if we go that route.

There is no expertise nor effort available at MZ for microcontroller developments. We should go that direction only if others (within L1Calo?) require similar circuitry and common designs & developments are possible. Anyway I wouldn't believe in any control that's free from programmable logic devices. If we think we need a processor, wouldn't Xilinx Zynq be an appropriate choice ?

Can we solve that problem with help of a mezzanine ?

--

Parallel I/O between FPGAs: In the specification there are 'only' 238 parallel i/o lines between the two FPGAs. Even if we run them at 960 Mbit/s, the maximum available data payload per BC is only 5712 bits, corresponding to the data from less than 45 input links at 6.4 Gbit/s. So the boundary between FPGAs is not transparent, and we will need to be selective about which L1 system links come to which of the two FPGAs, and what/how much information received in one FPGA can be made available to the other.

Agreed. There is probably nothing we can do about that at h/w level.. Planning to use the largest FPGAs anyway, limiting to two of them, any other scheme would probably be worse.

--

Output to CTP: The specification calls for optical outputs to CTP, based on MGT links. But what will be the link speed/protocol/latency?

Link speed / protocol assumed 6.4Gb/s 8b/10b, but as CTP planning to use same device family, there is plenty of options. Latency assumed to be below 2 ticks per end. To be confirmed.

=====

Physical hardware issues:

--

There are several places where it is pointed out that the ATCA specification isn't fully available to us (it is an expensive thing to get). Can we get access to the specification through e.g. CERN? This would be especially important for the electrical specification of the ethernet port with the ATCA base interface.

Thanks for suggesting that.

--

PCB thickness: One thing I do know about the specification is that board thickness is up to 2.4 mm. This is good for many-layer boards with impedance-controlled lines and good mechanical stability.

Thanks

--

Optical backplane connectors: The specification mentions either four or five optical feed through connections in zone 3, with a baseline of 48 fibers per connection. But the feed-through connectors indicated have a vertical height of 21.35mm each, and the ATCA mechanical specification only has 90mm of usable card edge in zone 3 after the guide pins above and below. So I think the practical "safe" limit in Zone 3 is four connectors, not five.

Agreed, we would probably need to omit some alignment h/w if going for 5 connectors. 4 should do for 48-way connectors.

--

A suggestion for the detailed design checklist: Attila has been working on FPGA-based timing boards for XFEL with multiple high-speed and low-jitter links. He has found it useful to break out individual power planes for each receiver, isolating each from the main power plane with a ferrite bead. I think it is worth considering for L1Topo as well.

We will consider that and will check against Xilinx docs and design density

=====

Organizational issues:

Not all of these may be answerable by Mainz, but we do need some clarity on these given the recent decision to separate L1Topo from L1Calo:

--

RODs: Who is responsible for developing/providing these, and which readout partition do they sit in? This has consequences for the actual

physical link interface and protocol, as well as readout software and firmware development.

thanks for flagging

--

Run-control software: Which software partition does this belong to?. For instance, to support the algorithms under study, L1Calo will need to apply threshold cuts for different e/tau/jet RoI definitions after they are received in L1Topo, and not before. How do we reach an understanding on how we will control and configure the L1Topo algorithm firmware?

--

Test environment at CERN: Does it belong in the CTP or L1Calo test areas? The most challenging interfaces for initial integration in 2014 will be the input links between the various L1Calo CMX modules, and the readout links to the RODs. If these are in different places, this could pose challenges.

Only option for any integration tests will probably be L1Calo test lab at CERN ?

--

Finally, there is the issue of the actual algorithms and data that L1Topo will process. I have assembled an initial draft outlining the kinds of algorithms under consideration, and what data contents and bandwidth might be expected from each L1 subsystem (L1Calo, eFEX, jFEX, L1Muon). I have received a little feedback on these now, and am attaching a slightly updated version.

Thanks

As a general comment, I think that the interface to the CTP needs to be better specified. This obviously applies to the physical layer of the connection, but more importantly also to the data content. For instance the current design foresees two ribbon fiber outputs to the CTP, assuming a link speed of 6.4 GBd (128 bit/BC) this translates to about 3000 bit per BC. This is clearly overkill, given that the upgraded CTP will be able to handle 320 trigger inputs (this includes both the optical and electrical inputs. I would therefore like to arrive at a better understanding of what kind of trigger information L1Topo is likely to send to the CTP. My assumption was that that a small number of bits (minimum one) per topological algorithm would be sufficient (this is even mentioned in the L1Topo specification on page 14). Given that the number of algorithms is probably not very big either (<10?), and that the multiplicity signals can continue to be fed to the CTP via the existing electrical connections, a relatively small number of bits should be sufficient. In this case I don't see why we shouldn't go for an electrical connection between L1Topo and the CTP as well, given the obvious savings in latency. In this respect, we could even foresee an

electrical input for a small number of bits (~32) directly on the new CTPCORE in order to further reduce the latency. However one would need to come to an agreement on the implementation soon.

In addition I think that installation issues should also be considered early on. In particular it would make sense to install L1Topo close to the CTP, in order to minimize the latency for the muon inputs in phase-1 (actually for the reduced number of signals in phase-0 as well). Otherwise a latency penalty of 3-4 BC would probably result just from the cabling required between the two levels in USA15. We thought there may be space in the CTP or MUCTPI racks, however I'm not so sure anymore. In particular the problem of the direction airflow for the ATCA chassis and the requirement for a probably fairly large optical patch panel could be problematic.

Thanks. No, we do not require the large bandwidth. However, we are concerned about latency and would therefore probably appreciate some flexibility to transmit latency critical data early on in the data stream. Also, there has been some effort been made at MZ quite a while ago to bypass large fractions of the data formatting stages (in Virtex5 and 6) and run with optimized encoding reducing latency. It is assumed that both L1Topo and CTP are powered by the same devices and therefore compatibility for any conceivable encoding is guaranteed. This is not something that could possibly be agreed on here and now. Any latency reduced transmission scheme would only be employed if absolutely required. Otherwise it's just simple 8b/10b encoding, possibly at highest sensible data rate (not for bandwidth but for latency).

Location of crates and electrical formats should indeed be agreed on soon, if possible. Unfortunately I am myself not quite aware of what information the various explored algorithms would yield.

Below are more detailed comments on the documents provided.

L1Topo user requirements document

- Page 2, section 1, paragraph 2:

A "standard ATCA shelf" is referred to, but there are any flavours of shelves. In particular no detail on the number of slots or the orientation of the boards or the directions of the airflow is provided.

Agreed. It would be sensible to design the module for a "standard" ATLAS ATCA crate if that were defined in time.

- Page 2, section 2, list item 2:

How will the TTC clock pair in zone 2 be driven? Will this require a custom backplane?

The idea to run certain signals on the backplane had once been considered as a general scheme for future L1Calo modules. That hasn't been followed recently. We would certainly make optical front panel connectivity available, in case no backplane based scheme can be devised. Please note that we are prepared to run IP connectivity through the front panel as well. In this case we wouldn't require any Zone 2 backplane at all. It could be there, though, just unused.

- Page 3, list item 8:

At least the number and speed of the electrical signals to the CTP should be specified. I assume they are equally split over both processing FPGAs?

That's the intention.

- Page 3, list item 11:

Is the jitter cleaner chip chosen guaranteed to have deterministic phase relation between input and output clocks? We had some bad

surprises with a chip from TI.

That's not known. However, the jitter cleaner is meant to be used on the MGT clocks only. That would be of the order of 320Mhz. Due to the multiplying up and dividing down of that clock in the MGT tiles we would expect an unknown phase offset of that order anyway, but we would appreciate input from others...

It's also not clear to me how a clock extracted from an optical output is used as a bunch clock source, does the FPGA recover the clock? Does it also go via the jitter cleaner?

We would indeed expect the FPGA to be able to recover clock and data from any reasonably formatted stream received on optical inputs, in a MGT. For that end we would be ready to supply the respective MGT with a separate, crystal based reference clock (multiple of 40.08). The recovered clock would then be routed through the jitter cleaner. The details hadn't actually been thought through in detail yet and therefore we appreciate that comment.

- Page 3, table 1:

A bandwidth of 13 GBd is quoted, however the optical receivers are only specified to 10.3 GBd. In fact since the bit rate must be a multiple of the bunch clock and the number of bits per BC must be divisible by 32, 9.6 GBd is a more realistic figure.

Yes, so far we have to rely on preliminary data sheets on miniPODs and that suggests 9.6 Gb/s line rate maximum.

The number of electrical outputs is 48, however the specification mentions 44 bits to the CTP, which is correct?

For electrical transmission we were planning to use exactly 1 bank per device, and since there are some multi use pins on some banks, that's not yet finally defined. However, we should seize the opportunity to agree on an exact format.

- Page 4, item 6:

How many bits per BC are sent from L1Topo to the DAQ and RoIB? What is the expected data content? Is a single G-LINK really sufficiently fast?

We need to define a requirement there. Large numbers of SFP links are probably a bad idea, we should go for miniPod, if one or two SFPs is considered insufficient.

Project specification

Page 3, section 1.2.1:

It's not clear to me whether the FPGAs operate receive the same data via their optical links (which would require duplication at the source) or if they exchange the missing data via the on-board connections.

Even though we tried to widen the processor-to-processor RTDP to the extent possible (at the expense of rather narrow electrical CTP and control paths), the inter-processor path is expected to be a bottleneck. There is nothing we can do about that, and, in fact, we might require to duplicate at source. The CMX is built to allow for upstream duplication.

Page 4, paragraph 4:

If the data duplication is expected to be done at the source, then the required fan-out should be specified, given that it impacts the design of all the modules sending data to L1Topo. In particular it should be clear if both processor FPGAs on an L1Topo module are expected to receive identical copies of the trigger data via their optical inputs. The maximum number of L1Topo slices also needs to be defined for this.

See above. We expect to see no more than two processor modules. We are open to either build additional copies or redo the modules, if after pre-phase1 data or algorithms don't fit the processors. However, the chosen 2-processor scheme using the largest devices on the market is probably best we can do. Any improvement over that would probably depend on higher bandwidth devices coming to the markets. Would appreciate any better suggestions.

Page 6, paragraph 3:

How will the on-board IPMC (Intelligent Platform Management Controller) be implemented on the L1Topo module? This is mandatory and does a lot more than just environmental monitoring. Normally the IPMC function is performed by an on-board microprocessor with the associated software.

No effort went into that in MZ. We would need to take over any design possibly available within L1Calo (RAL) or CERN. The IPMC connectivity is meant to go onto a mezzanine module and full ATCA functionality would be available only once that's in operation. For initial tests on the bench that functionality is not required. Open to suggestions.

Page 7, section 2.2:

Throughout the documents sometime 4 and sometimes 5 optical backplane connectors are mentioned. As Sam pointed out the allocated space in zone 3 (95.1 mm) only allows for 4 optical connectors.

5 connectors would be possible only if either sacrificing alignment pins or milling down the CPI connectors. Baseline is 4 connectors.

Page 7, section 2.3, last item:

As pointed out above the 24 GB/s to the CTP is complete overkill, so I would say that this is a functional requirement, in particular since it's not clear what the data content would be.

Agreed. It's not a requirement. Should anyone come up with a good reason not to route all 24fibres to the CTP we will happily consider some of them spares, which could actually quite useful on the DAQ/ROI path. Routing data right off the processors will free bandwidth on the processor-to-control FPGA path.

Page 8, bullet list:

I think the potential GbE or PCIe connectivity for controlling the module would probably require a 125 MHz reference clock in addition to the other clock mentioned.

Is being made available.

Page 12, last paragraph:

Clearly the module should be made compatible with the ATCA base interface, unless there are strong reasons not to do so. What documentation exactly is missing?

Since you provided the link, we will probably be able to manage it. See Sam's comments on implementation of the links. Since base interface is redundant, we consider implementing one of the links via a processor, one of them VHDL coded in an FPGA. Open for suggestions.

PCIe was also mentioned as a possibility for module control (page 6, paragraph 4), it's not clear how this ties in with the other possibilities and what the preferred option for module control is.

PCIe has briefly been discussed within L1Calo. I believe that's not a favoured option any more. We really need to be concerned about what might once be supported by online software.

Uses requirements: 2.2, 2.11, 3.5

I am not sure what to conclude about the clock discussions here and elsewhere; it appears that we have not converged on a clocking scheme, and are building in multiple options. Aren't there constraints from the neighboring modules, in particular cmx and ctp which define at least the 2014 clocking requirements? What about the muon system 2014 inputs?

There has been agreement that all transmitting modules will do so at multiples of the LHC bunch clock. There has been no decision yet what reference clocks should be used on the receiving end. The receive reference clock is used only so as to tune the receiver PLLs to somewhere near the embedded clock in the data stream. A 40.00 multiple would not be within the specs of reference clock error for Xilinx MGTs. 40.08 would be.

I also don't understand how the small changes in LHC frequency during acceleration will be handled, but presumably this is hardly a new issue, and will be handled in the same way L1Calo and CTP do.

If frequency variations were too large, lock could be lost. However, nothing is known (and there will certainly never be any specs by Xilinx on that) what will happen if the reference clock isn't stable. We will need to be prepared to do a resync after ramping, I would guess. BTW I am wondering whether embedded comma characters used as representation of "zero" would deal with such problems in an elegant way...

2.17, 3.6

I do not understand the user requirements here. Elsewhere in the L1Calo, we typically say something like "we will send to the ROD all the inputs to this module, plus all the new output information it creates". Are we able to make such a statement with only a single G-link? (see below)

No. happy to discuss how to deal with that. The specs talk about spare connectivity on miniPODs. We will implement what's required. Unfortunately I didn't get a lot of feedback on the users requirements, which haven't changed a lot since GOLD times...

Some big-picture things I missed seeing was something like:

one L1Topo Card will receive xxx fibers
each L1Topo FPGA will receive yyy fibers
one copy of the present L1 CMX card output would arrive on zzz fibers
the inputs expected in 2014 from muons would arrive on aaa fibers, or on bbb lvds inputs
the currently projected eFex outputs would arrive on uuu fibers
the currently projected jFex outputs would arrive on www fibers
the current CMM outputs to CTP occupy kkk lvds lines
Therefore one Card would handle nnn copies of CMX outputs and all expected muon 2014 inputs, plus send to CTP all current CMM outputs plus xxx more
However handling the eFex and jFex would require mmm more cards.

Table 1 never mentions zone 3

Sorry, an oversight. Should have stated that the real-time input is in zone 3.

At one point we discussed local electrical duplication of inputs to L1Topo by a trick with the fpga inputs Sam mentioned, I believe. What is the status? Has this been discussed and excluded, or is it still on the table?

Well, let's discuss that. My point of view has always been that, while useful for fixed algorithms (eg. Sliding window) it is not an option for topo. We would have to decide at time of board design (ie. NOW) what fraction of connectivity is used up for duplication. Any link used for duplication will be unavailable for data input...

3. For a board about to go into a prototype construction felt to be nearly identical to the final, there a surprising number of items classified as undecided, or "implementors lack ACTA documentation".

Surely there must be some mechanism to obtain ACTA documentation for the implementors.

That problem might be solved now. However, it was not clear from the beginning that ATCA compliance is of any use for this module. We are not relying on ATCA backplane connectivity. It's bare form factor and power scheme. We are happy to do fully compliant module if some level of support is available. It's the first ATCA module being built in Mainz and within L1Calo, I would think...

What are the plans for firming up the other questions in the design? Is that still to happen before first prototype, or between proto and final production?

We are hoping to sort out a large fraction of questions between now and prototype production. However, that might require a larger fraction than anticipated, of functionality being located on the mezzanine. That shouldn't pose any real problem: The JEM is equipped with 7 mezzanine modules and is successfully running in 32 copies in the pit for years now...

3.4 The policy here is fervently bleeding-edge; it is mentioned elsewhere that the path to actually using 13GB/s remains unclear.

Is 13 GB really the right target for 2014? Or should it wait for 2018?

We do not expect to use 13Gb/s in 2014. By time of module production we will have to decide on components to be mounted. We will buy the more expensive 13Gb/s types only if by that time we expect to see input streams of that rate, and if we see 13Gb/s miniPODs on the horizon...

3.8 USB

Comment: one of my colleagues on another experiment (however, less experienced with firmware than you) abandoned USB and went to a TCP/IP emulator followed by VJTag and a soft-core processor as a control and readout interface...this was also on alterra, so usb device firmware or the usb driver on the pc might have been the issue... are we confident that the Xilinx USB implementation is bulletproof enough to depend on for pre-configuration access?

Preconfiguration acces is already a belt and braces thing. And Xilinx USB access scheme has so far worked flawlessly in the lab.

Project specification

1.1 no muon document

well, yes,...

1.2.1 I have heard rumors of a muon format in 2014 of perhaps 1 eta and 3 phi bits per RoI, but I have no idea of the number of RoI's, nor the electrical/optical format. What provisions are we making for such a proposed input?

So far we have in (rare) discussions with muons talked about optical connectivity. There is plenty of fibres available for that. Limited electrical connectivity is available via the mezzanine module, as recently discussed with the CTP. Obviously that would have to be shared between the two modules then. Dedicating further electrical bandwidth to external In/Out would be dangerous since this would be taken off processor-to-processor bandwidth, which is a bottleneck already now (see Sam's comment)

Page 4

No drawing of a L1Topo crate with > 1 Card, no discussion or otherwise of backplane in ACTA crate

Wouldn't mind adding some further drawing. However, topo modules wouldn't be talking to one another via backplane...

Data Processing

it sounds like the only outputs from L1Topo will be a few yes/no bits. Not even the list of candidates presented to an algorithm? This sounds like a tough way to verify/debug.

Not quite sure what you mean. Here we are talking real-time path and there we'd in fact send only a small data volume to the CTP. Any debug related stuff would run either/or to DAQ / crate controller CPU.

top page 4: direct connection to CTP? when will this decision be made?

I would happily make that decision *now*

bottom of page 5: "some application software would have to be written" for JTAG access to FPGA

I think it would be useful to have a list of the various pieces of software that need to be written to support the LITopo, and if possible a preliminary list of those responsible. I realize that this is a preliminary review of a card, but I am somehow missing the feeling that we are reviewing a project, not a card. Have we assigned the necessary personnel to carry out the project?

I appreciate that comment. We are likely to get in deep water and we should include our current L1Calo online software people in that discussion !

Top page 6: USB preconfiguration/system ace/ external flash card writer. Will this be resolved before the prototype?

We should have several optional configuration paths. The added cost is low. ACE is obsolete . Some recent schemes are horribly slow to update. No golden scheme seems to exist.

Module Control: there are many possible solutions listed; is the optical VME extender the selected solution? when decided?

VME extender is being used only until L1Calo / L1 online software supports modules controlled via Ethernet. L1Calo needs to sort that out for FEXes and their R&D anyway. Choice of data link doesn't affect LITopo hardware design.

2.3 I would have thought "interface with CMX outputs of 6.4Gb/s" would be a functional requirement; similarly for CTP; or eFex or jFex...along with the technical means to make that so.

... that makes sense

2.4 the clock discussion cycles back yet again. It is confusing to have it spread over so many locations.

... sorry

2.7 see earlier question on whether single Glink is enough

3.1 some of this would have been nice to have in an introductory overview (eg near a crate diagram)

P 12 bottom

now it appears that IP control is considered after all--I thought the earlier part was tending to use extended VME and ignore IP?

See above

again, complaints about lack of atca doc. how and when will this be resolved?

3.4

the text mentions a half dozen signal standards. are you implementing all of them? if not, when deciding?

Well, that's a wild mixture of standards indeed. Each for its specific purpose...

MGT here described as 6.4 or 10.0 (would have been welcome earlier); though unclear how that meshes with 13GB mentioned earlier

4 software at last

as mentioned earlier, need to flesh out project and its software and firmware parts, effort available, schedule, for this to grow from a board to a project

4.1 playback and spy; could (as mentioned above) use more on how fully these are useable with this hardware: what are the requirements, and does this hardware meet them

Well, might expand on that a bit. Should perhaps say that playback and spy are implemented in firmware and certainly required in modules early life. Might be kicked out at a later stage if more resources required on real-time path.

I might have expected comments about firmware structure segregating data movement and control, from algorithmic configuration and execution, with the former being closer to hardware and thus required to be Mainz-dominated

4.2 hmm, what is xx%?

Oh, sorry that's 4% of a XC6V550T as far as we can say right now.

yes, algorithmic firmware development and organization needs fleshing out

5.1 would have appreciated early on a diagram showing these modules on the card

5.4 13Gb: still a requirement to support 6.4 and 10.0, however?

Yes. For clarification: the FPGAs to be mounted eventually are supposed to support all rates without any rate gaps.

"optical wavelength of these devices is not compatible to the current TTC system and an external converter needs to be employed". How vital is this? Who will do it?

Mainz would have to supply a tiny box doing the conversion. That seems to be more sensible than building some outdated lowtech thing on a new module, knowing that it will be replaced at some stage anyway. Open for discussion...

5.5 need to agree with Sam for a web home for document under development and insert reference here and at head of document

Note on Topology Processor Development

I am confused; mostly sounds like Gold document or early draft? but also gives requirements for CMX which I think were not adopted (10 Gb/s); so I did not read and comment on this document in detail.

Sorry that wasn't meant to be part of reviewed documents. None of the subdirectories are...