Critical Systems Labs

Strategic Insight for Safety

Beam Loss Monitor System – Preliminary Feedback

November 11th, 2010

Purpose and scope

- This review will seek to:
  - assess the adequacy of the overall BLM system design with a focus on the programmable parts
  - identify possible weaknesses in the programmable parts of the mission-critical BLM
  - suggest activities that could increase the level of confidence that the programmable parts of BLM system performs as intended
  - suggest potential improvements of the BLM
  - provide a general comparison of the BLM with approaches in industrial systems.
- The scope of this review is also limited to a consideration of:
  - Potential sources of unsafety within the BLM, where the detection of an amount of particle losses that has the potential to quench the magnet is not relayed to the Beam Interlock System, resulting in a 'missed generation of beam dump trigger' and potentially machine damage.
  - Potential sources of unavailability, where failure of the BLM leads to a request to the Beam Interlock System to dump the beam, resulting in a 'false dump trigger'and some machine downtime.

Novel solutions in the design

- Dynamic range of detectors is broad and required novel approach to capture and monitor this information
- Has introduced some novel solutions to solve problems, e.g., use of ADC to increase dynamic range. Compared to BIS which is very much all just proven technology.
- Novelty is usually avoided in critical systems but in this case it is a necessary novelty – not just novelty for the sake of being novel

Fault Tolerance is substantial

- Fault tolerance of critical data path
  - Sensors
  - Communication links
  - Computation and decision making components
- Many strong and impressive aspects to the fault tolerance, eg.,:
  - Using CRC check + 8b/10b protocol for optical links, redundant of optical links
  - reading out threshold values from FPGA every minute and checking CRC
  - ~4000 sensors, infrequent cases of high loss detected by only one sensors
  - Dozens of other examples…

Critical data path only partially redundant

- Substantial, but not “end to end” as per other finding
- Some computation and decision making components (such as threshold comparison in the surface FPGA) is not redundant

BLMS has mixed purposes

- This system seems to have started its life as a measurement system; it has evolved towards being a critical system
- Possibly this affected some early decisions about budget and design
- Going forward, does management regard the BLMS to primarily be a measurement system, with a secondary role as a contribution to machine protection?
  - Seems to be some uncertainty or mixed messages in this regard

Critical and non-critical function mixed

- Mix of critical and non-critical functionality in design
  - 50% (?) of the logic on the surface card is not related to machine protection
- A common practice in industry is to separate the critical and non-critical functionalities
  - To guarantee high availability of resources required for critical functions
  - To avoid the possibility of negative effects on the critical functionality by non-critical functionality

% Logic Used on FGPAs is Very High

- At first glance, the % logic used is a very big concern.
- Non-technical constraints appear to have forced this threshold to be exceeded, which may be regrettable in the long term for various reasons
- The designers used ingenuity to find a solution. Workarounds (i.e., squeezing more logic than desired into the FPGA) appears to be done carefully with attention to risks
- But could be a problem in the long run, with respect to maintainability, e.g., changes to the design, re-use and porting to new hardware.
- Now “boxed in” to a very small corner. Hard to imagine going beyond 95% and yet they seems to what to add more

No (Current) Functional Specification

- There is no one document that specifies the functional behaviour for the current system, as now deployed
- Various documents (e.g., published articles) capture the intent for the functional behaviour but this is fragmented and scattered
- What is the basis of design testing in the absence of a functional specification? (It depends too much on knowledge stored in people's head)
- Even a modest effort could produce a comprehensive and up-to-date functional specification that would provide linkage between intent and design

Design testing is impressive

- Very impressive effort to test the design using complementary levels of testing, e.g.,:
  - Radiation testing of Tunnel cards
  - Surface Board uses Simulation, Hardware based-testing, Software-Based testing
  - Combiner card

Design Testing is not sustainable

- While the testing is impressive, it is not always documented and therefore not sustainable
- There is no master test plan
- The group's philosophy seems to be automation over documentation but will anyone know what to do when there is a change to the design

“Proof Testing” is impressive

- Very impressed with the approach taken in test the proper operation of the detectors each 12 hours*
- For example, basic connectivity test has evolved into a much more comprehensive test for proper function that has already been beneficial
- Importance of this testing is amplified by fact that detectors are a single point of failure (in the worst case) and some failures will mask a dangerous loss, i.e., a possible failure mode is indistinguishable from low count
- Will proof testing with CS source in tunnel be performed on a regular basis??

* To be precise, a new fill is not allowed if this proof testing has not been performed within the previous 12 hours

Some known limitations

- Limit of dynamic range close to the Injection Points
- Others?

Implications of Operating at Higher Energy Levels

- Noise
  - The current dynamic range for ADC may result in detecting noise when operating in nominal energy
  - One possible solution is using ASIC instead of ADC
  - The group is aware of these problems and is looking at ways to address them

Human Error

- Typically there would be limits on how much a critical value can change in one interaction
- For the BLMS this is not true for threshold values in the Master table

Maintainability Aspects

- Many safety problems are introduced by maintenance actions
- For example, US FDA analysis of 3140 medical device problems between 1992 and 1998 showed that 7.7% were due to software failures and of these, 192 (79%) were caused by defects introduced after initial production and distribution

General Principles of Software Validation;
Final Guidance for Industry and FDA Staff – January 11, 2002

Maintainability Aspects

- Over the long term (the next 20 years?), it could be difficult to maintain the system
- Currently, depends far too much on knowledge in people's head such as:
  - smart optimizations (in VHDL) not easily understandable,
  - various levels of testing
- BLMS team shares this concern:
  - seems to favor automation over documentation as a means of addressing this concern
  - However some documents produced such as "Management Procedures of the BLM System Settings"

Maintainability Aspects

- BLMS already close to the physical limits of some components (Surface FPGA). In this context, any change to the design carries some significant risk and will be challenging for the maintainers

Backup SLides

Percentage of Logic used on FPGAs

- Approx. 85% for the Tunnel card (BL…)
- Approx 95% for the Surface card (BLETC)

Percentage of Logic used on FPGAs

- Using too much of the logic on an FPGA is not recommended because:
  - Heat related issues
  - Placing and routing become more difficult and performance
  - ‘Performance’ degradation: the output may become unreliable at the intended clock rate
- The suggested upper limit we have heard is 70%