Processing algorithms development

- Aims
- People/Tasks
- Documentation
- Roadmap

Aims

This is the first meeting – just to spark the discussion

For sure we are not going to solve/answer all the problems today

We must define what we need/expect to have as the outcome of the work done on DSP

NOTE – this is my personal view derived from the work I've done on the Tell1 DSP, any suggestions are warmly welcome!

Aims

New ASIC is being developed as a common part for TT, VELO (strip option) and IT (strip option)

Back-end of the chip can be used for SciFi (and possibly RICH)

The ASIC is an example of SoC (System on-Chip) device: each channel is individually instrumented – can perform digitisation, digital data processing (DSP) and zero-suppression (ZS)

Raw ZS data from the chip will be sent to the Tell40 for further processing – need to study and optimize the ZS data transport protocol then

The big picture – we want to have a product (chip) that will perform signal processing and send the ZS data down the DAQ line to LLT

The processing chain:

- Base line correction (aka pedestal subtraction)
- Common mode suppression (MCMS or/and LCMS)
- Spill-over rejection
- Zero suppression
- Data encoding (Raw ZS data buffer creation)

Note, the functionality will also include switching on/off each individual processing stage, masking bad channels, calibration constants, NZS production,… etc

This is a bit of a trivia by now but I will repeat this anyway:

Once diffused the chip cannot be changed

We cannot afford mistakes – cannot make corrections on the Tell40 ZS must be done on-chip

This requires very detailed studies of the DSP on both high and low level – EMULATORS

Can exploit the experience gained with the present framework we use to run VELO and ST - VETRA

The model of collaboration and development process

- We should have a common project to share C/HDL code
- Can use the existing LHCb SVN structure
- This should be easy to use and stand-alone (dont want to give the Gaudi tutorial for each electronic engineer in my group…)
- Re-use (some of) the tell1Lib legacy
- Extremely easy for people to use/update the code
- The core part – C/HDL engines (DSP algorithms models)
- Tools for configuration, IO, simple framework to run the processing, monitoring etc. (dont want to get into technical details here)

In the future this can be used to build VETRA++ to run in the pit

The processing much more complex but we have more flexibility

We can re-load firmware (whenever) we want

I believe we should make it a part of the ASIC software project

- we will need high level models of the Tell40 algorithms
- we should have the proper emulation as well

The main outcome of this meeting should be to identify manpower

Then we can assign tasks and start discussing technical details

How soon can we provide information on people that can start developing C/HDL

We should also have another iteration on the data processing itself – be more specific what each sub-detector wants in the chain

Shall we try to crate a document (similar to our chip spec note) regarding data processing on the Tell40?

Twiki (thanks to Massi) :

https://twiki.cern.ch/twiki/bin/view/LHCb/StripAsic

The next meeting regarding the processing should be much more technical

Remember that this project is going to have life-time much longer that the electronic design

- Emulation will be needed for daily running
- Monitoring, calibration, data quality
- We need to think now about integrating the highl level emulation into the LHCb software framework
- The first version of the ASIC related data processing should be ready by 2014