CRU User Requirements

ALICE Common Read-out Unit (CRU)

Engineering Design Review, CERN, 20 June, 2016

Ernő DÁVID, József IMREK, Tivadar KISS – Wigner RCP, Budapest; *Tuan Máté Guyen* – TU Budapest

Filippo COSTA – CERN

Jubin MITRA, Sanjoy MUKHERJEE, Shuaib Ahmad KHAN – VECC, Kolkata

Requirements from the Detectors – I.

- **General**
  - The detectors will have to be connected to the CRUs with CERN GBT links.
  - GBT links shall work in GBT mode (multiple detectors) or in Widebus mode (TPC)
  - In addition, CRU shall be compatible with optical links with custom protocol. (e.g. TRD.)

- **Data streaming**
  - A single CRU shall support different number of detector links. (e.g.: MFT: 36, TPC: 20, TRD: 30) Some detectors may have different no. of uplinks and downliks.
  - The CRU shall able to receive the incoming detector data streams and pass them to the User Logic for further handling
  - The CRU's PCIe DMA stage shall accept the outgoing traffic from the User Logic and deliver the output data stream to the RAM of the FLP servers.
  - The CRU shall have the ability to buffer up a certain amount of detector data. (e.g. 1.5 MB (12 Mbit) for TRD).

- **Configuration Download**
  - Some detectors will utilize the GBT downlink for packet based communication to deliver large amount of configuration data (e.g. ~ 10 MBytes) to the front-end cards. This packet based slow control must be supported in both direction between the front-end card and the CRU.

Requirements from the Detectors – II.

- **Trigger and Timing Distribution**
  - **LHC clock distribution over GBTs with constant and deterministic phase relations between GBT links, CRUs, power cycles, and firmware updates.**
  - **Trigger (TTS) information distribution with constant latency over the GBT downlink data channel.**
  - **The CRU in case of an absence of signal (i.e. clock) from the LTU shall be able to automatically switch to a local clock of 40 MHz or LHC frequency not to loose communication and control to the detectors.**
  - **Even if not connected to the trigger system, CRUs itself shall be able to generate programmable triggers with fixed or random rates.**

- **Fast Read-out control**
  - **For some detectors, the CRU shall be able to generate and distribute read-out control signals to the FE ASICs over GBT downlink data channels with constant latency (e.g. TPC, MCH)**

- **Detector Slow Control**
  - **The CRU shall provide access to the GBTx ASIC registers (GBT control) for GBT/CRU control applications**
  - **The CRU shall provide access to the GBT-SCA features on the FE cards (detector control) for detector specific DCS applications.**
  - **The CRU shall support the response time requirements of the FE cards safety modules.**
  - **Some detectors (e.g. ITS) requires a broadcast ability for sending packets to the front-end cards.**

Requirements from the Detectors – III.

- **Data processing**
  - **The CRU firmware shall support a detector specific User Logic for data processing. (TPC, ITS?, TRD, MFT?)**
  - **The CRU shall be able to send the FLP multiple kinds of data streams (i.e. with different level of processing)**

- **Busy/throttle**
  - **The BUSY signal will be embedded into the incoming detector data packages, it must be extracted by the CRU and delivered to the trigger and timing system (to the LTU).**
  - **When the detectors send a BUSY condition the CRUs shall aggregate them and propagate them up to the CTP. The CTP will then see that part of the detector is busy. The CRU shall provide information which particular link had set the busy.**
  - **CRU shall be able to recognize busy/throttle states of data flow and deliver it to the trigger and timing system (to the LTU).**

Requirements from O2 and DCS Systems

- **O2 requirements**
  - **CRUs shall provide formatted data packets to the FLP computers with standardized common data headers called SDH.**
  - **Continuous data strems (data streams from detectors running in continuous read-out mode) shall be segmented by heartbeats.**
  - **Busy states and packet drops in data read-out shall be recognized and communicated to the trigger system to build drop maps.**
  - **CRU shall support the throttle mechanism specified as a system level requirement in ALICE Tech Note ALICE-TECH-2016-00, „*The detector read-out in ALICE during Run 3 and 4*”.**

- **DCS requirements**
  - **The CRU firmware and software shall provide a generic interface which is accessible on the FLP server on Linux application level, which allows any DCS related application to access the DCS related front-end functionality in a transparent way.**

General Requirements – I.

- **General Firmware Related Requirements**
  - **The CRU firmware shall be based around a modular and extendable architecture, instead of a closed monolithic architecture. The CRU firmware shall be extendable by the detector teams with their detector specific code for the data processing.**
  - **The CRU team shall provide a basic, generic firmware implementation which supports as many detectors as possible (detectors without special requirements for the CRU)**
  - **The CRU firmware shall support the remote firmware update and functionality control**
  - **All CRUs shall provide a „fall-back" solution bypassing the detector specific data processing, which moves unmodified raw data (a.k.a. link data) in standard ALICE data packets to the FLP memory.**
- **General Software Related Requirements**
  - **The CRU team shall provide two levels of software support:**
    - Low level libraries with an API
    - Command line tools
  - **The CRU driver and API layer shall provide support for concurrent access of the underlying features. The API shall guarantee that the critical resources shared correctly between parallel users.**
  - **The CRU software shall support remote firmware update.**

General Requirements – II.

ALICE system level requirements on implementing responsibilities have to be addressed on two levels:

1. Procedural level

- **The FW development process, e.g the firmware integration workflow (testing, verification, release process and versioning) shall be specified and followed.**
  - Presently we have a proposal and we are collecting feedback from detector teams.
- **CRU team shall deploy firmware in CRUs**
- **The capability of switching back („fall back") to a raw data taking mode be always available without firmware up/downgrade procedure**

2. Architectural level

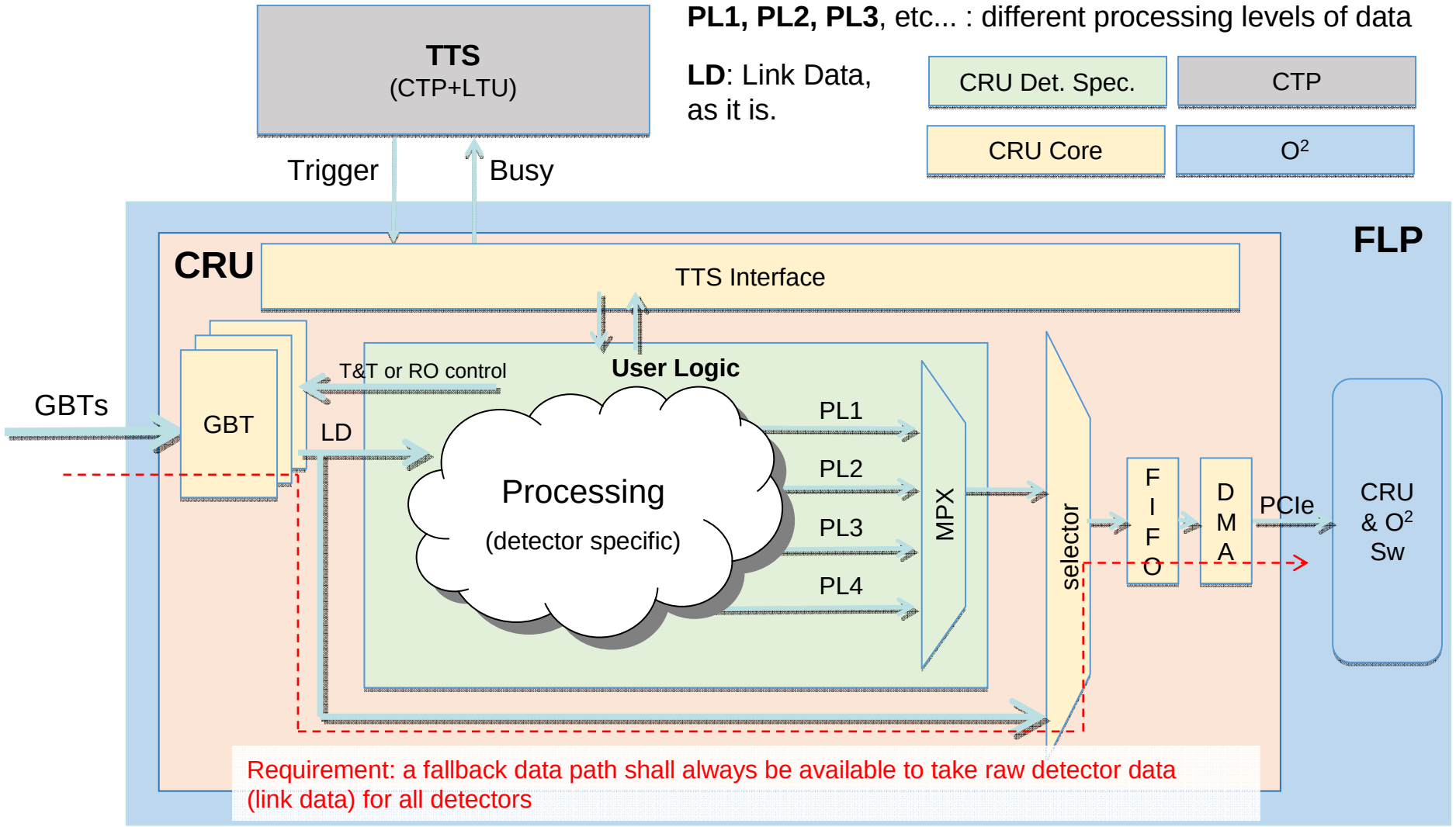
- **A data path shall be implemented that makes raw data taking possible independently of the detector User Logic, and low level softare tools shall be available to do it independently of the O2 software.**

Design goals:

- to avoid duplicating of development work
- to minimize duplication of FPGA resources
- → to be evaluated with the detector teams

Example 2 – User Logic with Processing

(developed by detector CRU teams)



Detector Specific Requirements (v0.7)

Det.	Link type	Device on sending card	No. of links		No. of CRU	Links per CRU	T&T through CRU	Busy by CRU	Using of GBT Downlink (e-links)	Using of GBT Uplink (e-links)	Proces-sing in CRU	SDH by CRU	DCS through CRU
			bidir	unidir									
TPC	**GBT** (GBTx)	**ASIC** (SAMPA)	7200	7200	**360**	20	yes	yes	RO-CTRL: 1x (or 5x) 4 signals @ 80 Mb/s (trig, HB, sync, rst)	28 x 160 Mb/s serial links (SAMPA raw data streams)	CF	yes	**GBT ctrl. ch. + SCA** (I2C)
MCH	**GBT** (GBTx)	**ASIC** (SAMPA)	550	-	**25**	24?	yes	yes	RO-CTRL: 8 x 5 signals @ 80 Mb/s (HRst, HB, Sync, Phys, x)	40 x 80 Mb/s serial links (SAMPA packets)	-	yes	**GBT ctrl. ch. + SCA** (I2C)
MID	**GBT** (GBTx)	**FPGA** (8x Max10, 2x CyclonV)	32	-	**2**	24?	yes	yes	RO-CTRL: 10 x 1 bit stream @ 320 Mb/s (RO commands (8 bit serially)	10 x 320 Mb/s serial links	-	yes	**GBT ctrl. ch. +SCA** (I2C, JTAG, ADC)
TOF	**GBT** (GBTx)	**FPGA** (Igloo2)	72	-	**3**	24?	yes	yes	80 bits of TTS @ 40 MHz	parallel mode, packets (w/ SDH + data) (or single words for control)	-	-	**NO** (Custom optical link)
FIT	**GBT** (?)	**FPGA** (Virtex6)	22	-	**1**	24?	yes clock?)	?	80 bits of TTS @ 40 MHz (?)	parallel mode, packets (w/ SDH + data) (or single words for control)	-	-	?
ZDC	**GBT ?** (?)	**FPGA** (Virtex5,6)	1	-	**1**	24?	yes	?	80 bits of TTS @ 40 MHz (?)	?	-	-	?
ITS	**GBT** (GBTx, FPGA)	**FPGA** (Kintex7)	192	384	**24**	24?	clock only	yes	parallel mode, packets (w/ DCS payload) (or single words for control)	parallel mode, packets (w/ SDH + data) (or single words for control)	CF (?)	yes	**GBT ctr. ch.** (I2C, JTAG) **+ GBT down- and uplinks**
MFT	**GBT** (GBTx, FPGA)	**FPGA** (Kintex7)	80	80	**10**	36	clock only	yes	parallel mode, packets (w/ DCS payload) (or single words for control)	parallel mode, packets (w/ SDH + data) (or single words for control)	?	?	**GBT ctr. ch.** (I2C, JTAG) **+ GBT down and uplinks**
CTP	**GBT** (GBTx, FPGA)	**FPGA** (Kintex7)	2	-	**1**	24?	-	-	parallel mode, packets (w/ DCS payload) (or single words for control)	parallel mode, packets (w/ SDH + data) (or single words for control)	-	-	**NO** (Eth link)
TRD	**custom** 8B/10B	**ASIC**	-	1044	**54**	30	-	yes	No downlinks!	No GBT - custom 2.5 Gb/s 8B/10B optical links, raw data stream	TK	yes	**NO** (Eth link)

How Many Different CRUs Will We Have?

- There is an effort to identify the possibility of common solutions, in order to reduce the number of different FW variants.

- Grouping / merging of detector requirements is not always possible
  - Some of the different detector specific interface and protocol requirements can be served by one common firmware, others need dedicated functionality.
  - High level of processing of data anyhow needs very detector specific and uses up most of the FPGA resources (esp. for TPC), so it needs dedicated user FW.

- Conclusion: We need multiple FW in the same HW, but we have to minimize the diversity
  - The *core CRU firmware*, developed by the central CRU team, and which contains the low-level interfaces and control functionalities shall be common for all CRU variants
  - All detector specific features shall be implemented inside the *User Logic*
  - There shall be a common, clearly defined interface between the two.

Thank you

Reserved slides

How Many Different CRUs Will We Have? (present understanding)

CRU	May be used by	Responsible team	Specialties	Main parts provided by CRU team
Firmware 1 (default, basic CRU)	TOF, FIT, ZDC (?), CTP, + ITS w/o CF, + MFT w/o CF	CRU	-	all
Firmware 2	TPC (with CF)	Detector + CRU	FE Read-out CTRL, Data packets receiving and decoding, Cluster Finding, SDH	GBT, PCI w/ DMA, TTS, DCS
Firmware 3	ITS (with CF)	Detector + CRU	Cluster Finding, SDH	GBT, PCI w/DMA, DCS,
Firmware 4	MFT (with CF)	Detector + CRU	Cluster Finding, SDH	GBT, PCI w/DMA, DCS,
Firmware 5	TRD (with tracking)	Detector + CRU	Different Detector Link, Tracking, SDH	PCI w/ DMA,
Firmware 6	MCH	Detector + CRU	FE Read-out CTRL, Data packets receiving and decoding, SDH	GBT, PCI w/DMA, TTS, DCS
Firmware 7	MID	Detector + CRU	FE Read-out CTRL, data packets receiving and decoding, SDH	GBT, PCI w/DMA, TTS, DCS