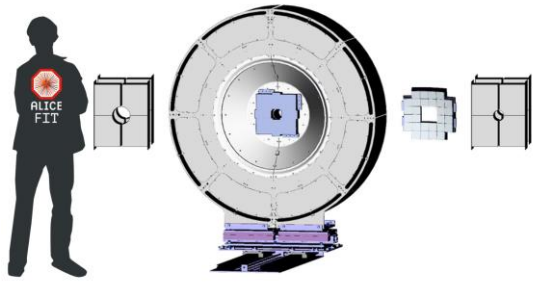


FIT Readout

D. Finogeev

Institute for Nuclear Research, Russian Academy of Sciences, Moscow

on behalf of the FIT Collaboration

Production Readiness Review, December, 14, 2020

I. FIT Readout Unit (description)

II. FIT Readout Software simulation

FIT GBT readout project

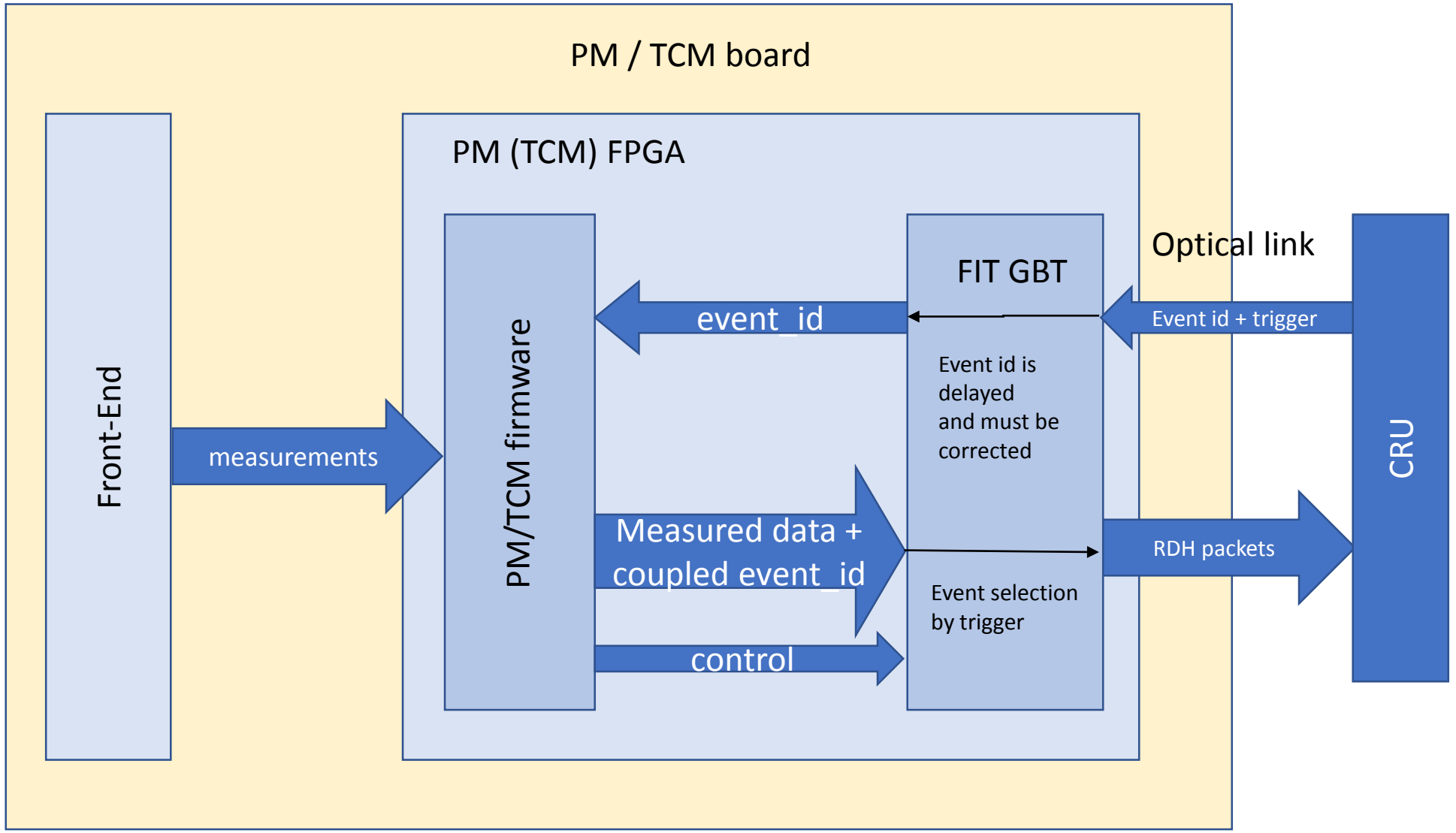
FIT GBT readout project :

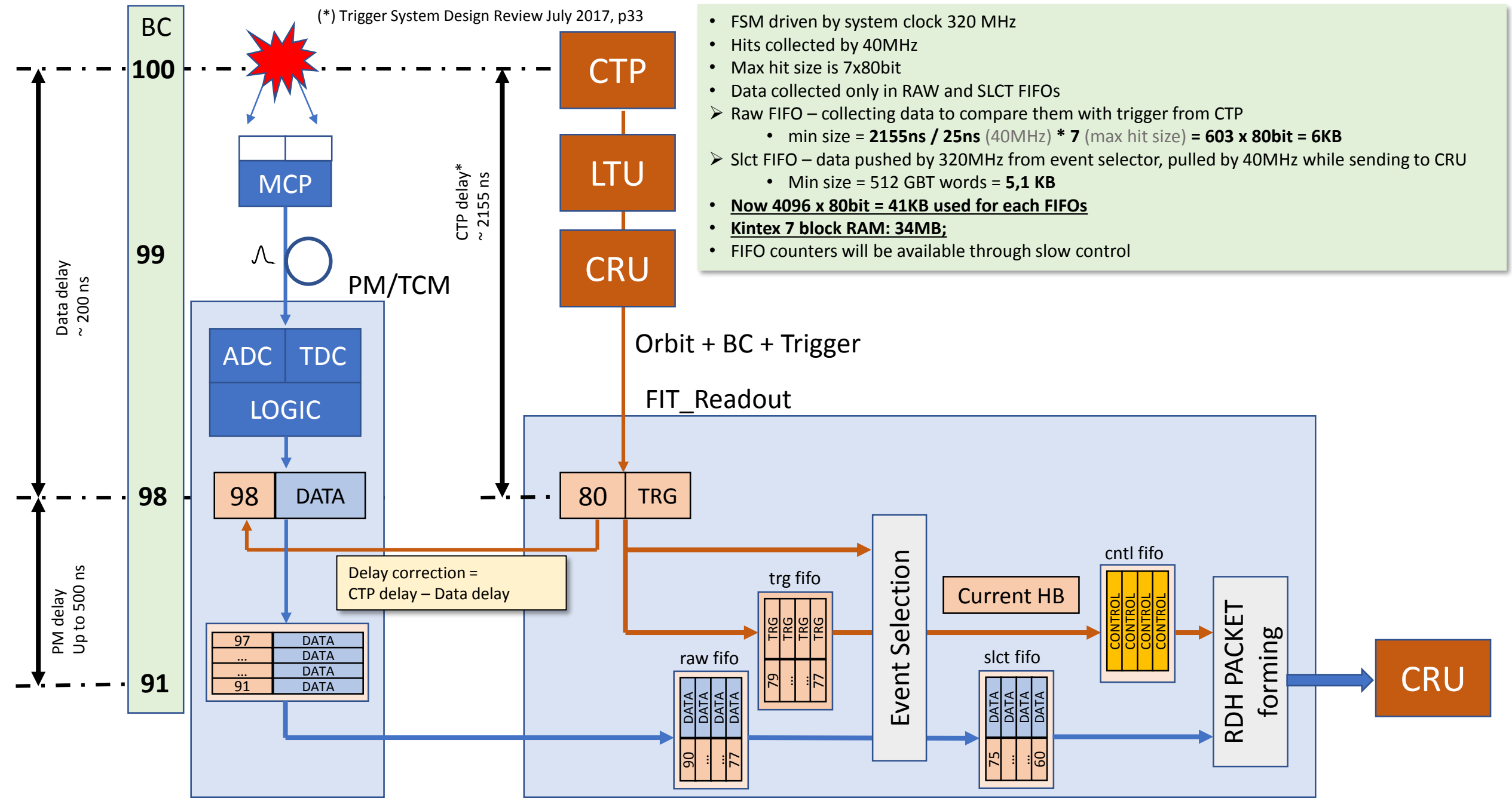
- Is a part of PM and TCM FPGA firmware, provides data exchange between PM/TCM modules and CRU

- Receives BCID and Trigger information from LTU via CRU trough GBT link
- Corrects event ID to compensate trigger data latency (near 2us)

- Receives data from PM or TCM logic, build RDH packet
- Selects detector data frames according to readout mode, and send readout data to CRU

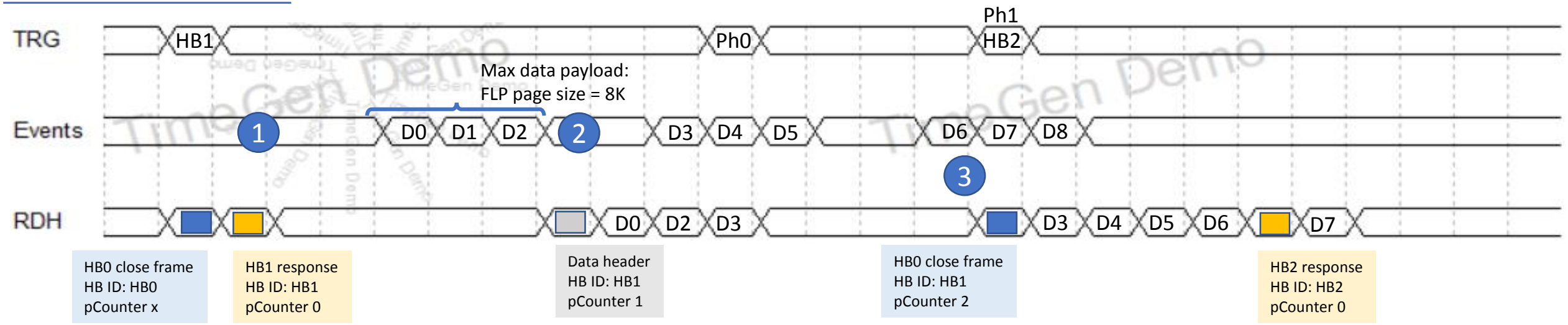
FIT GBT readout structure





- FSM driven by system clock 320 MHz
- Hits collected by 40MHz
- Max hit size is 7x80bit
- Data collected only in RAW and SLCT FIFOs
- ➢ Raw FIFO – collecting data to compare them with trigger from CTP
  - min size = **2155ns / 25ns** (40MHz) *** 7** (max hit size) **= 603 x 80bit = 6KB**
- ➢ Slct FIFO – data pushed by 320MHz from event selector, pulled by 40MHz while sending to CRU
  - Min size = 512 GBT words = **5,1 KB**
- **Now 4096 x 80bit = 41KB used for each FIFOs**
- **Kintex 7 block RAM: 34MB;**
- FIFO counters will be available through slow control

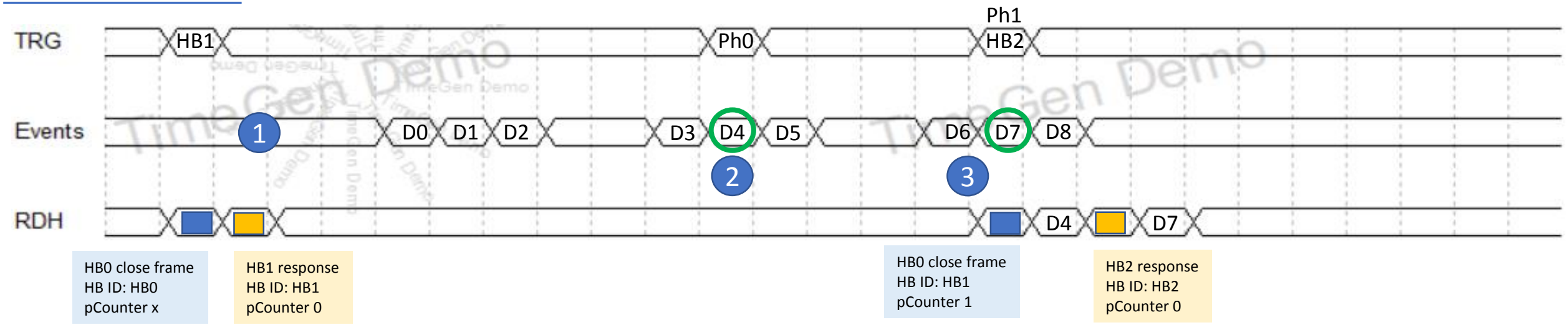
CONTINIOUS READOUT



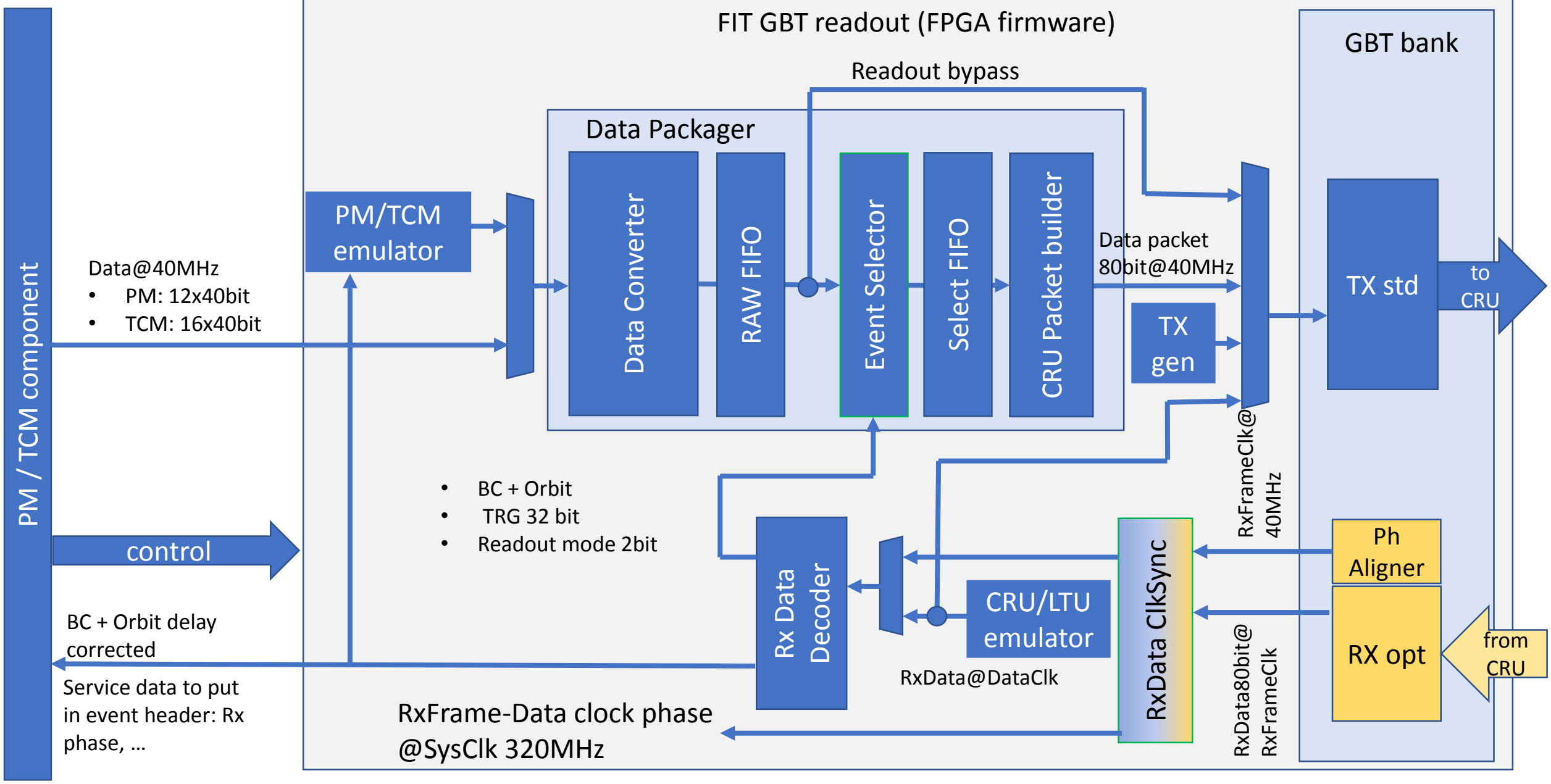
1) Closing frame for HB0; sending HB1 response
2) Data collected more than 8K send with out trigger
3) Then HB2 received frame with HB1 is closed sending remains data; new HB2 response send

- Upper data throughput estimation
  - Max packet payload = 512 – 7(RDH,trlr,xOP) = 505
  - Orbit = 3563 bc = 7 packets = 3535 GBT payload words
  - 1 PM hit (12 channels) = 7 GBT words; orbit = 505 hits x 11kHz orbit rate = 5.5 MHz hit rate

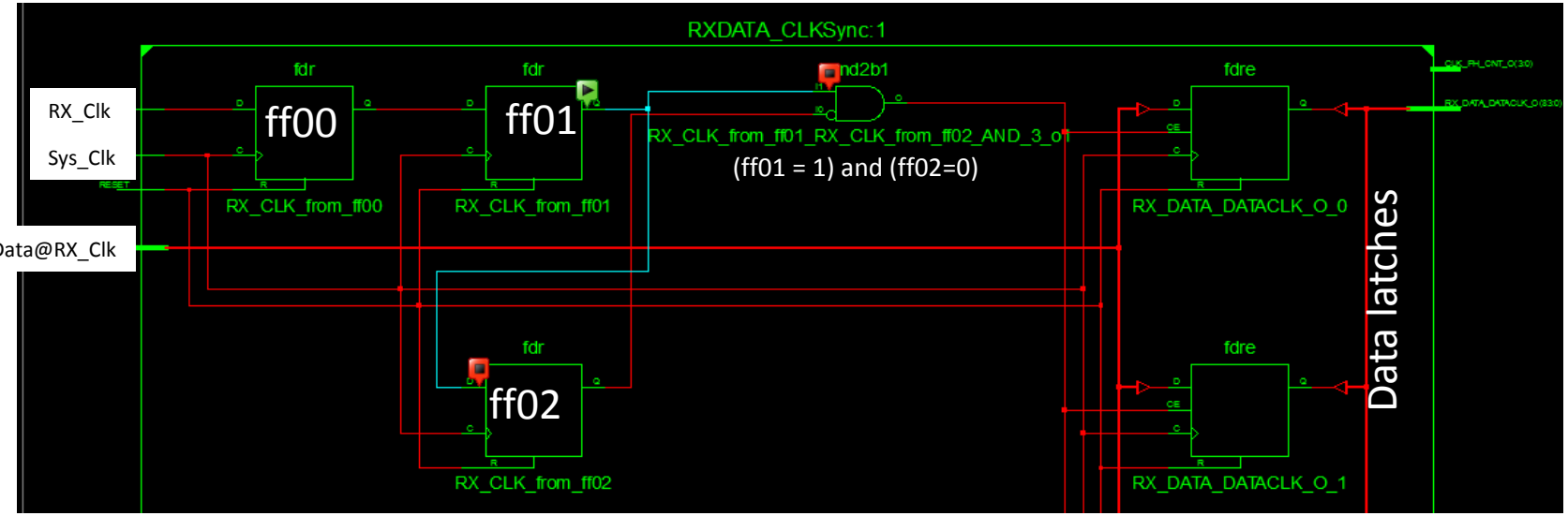
TRIGGER READOUT



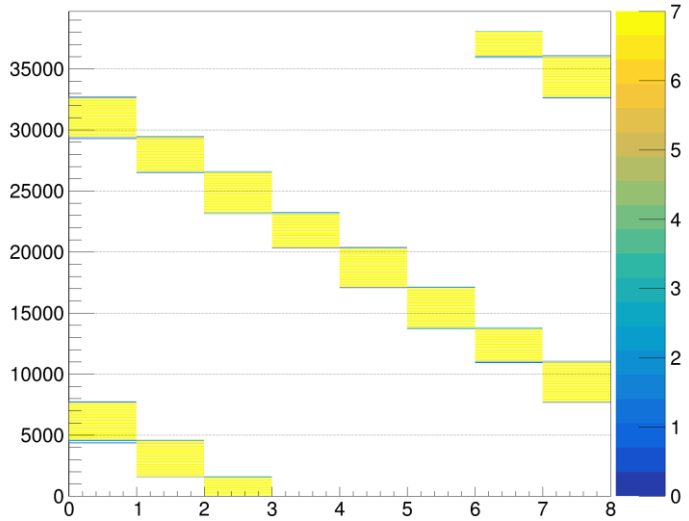
1) Closing frame for HB0; sending HB1 response
2) Data selected by trigger
3) Then HB2 received frame with HB1 is closed sending remains data; new HB2 response send


FIT GBT readout (FPGA firmware)
GBT bank
Readout bypass
Data Packager
PM / TCM component
PM/TCM emulator
Data Converter
RAW FIFO
Event Selector
Select FIFO
CRU Packet builder
Data packet 80bit@40MHz
TX gen
TX std
to CRU
Data@40MHz
• PM: 12x40bit
• TCM: 16x40bit
control
• BC + Orbit
• TRG 32 bit
• Readout mode 2bit
RxFrameClk@ 40MHz
Ph Aligner
Rx Data Decoder
CRU/LTU emulator
RxData ClkSync
RX opt
from CRU
BC + Orbit delay corrected
Service data to put in event header: Rx phase, …
RxData@DataClk
RxData80bit@ RxFrameClk
RxFrame-Data clock phase @SysClk 320MHz

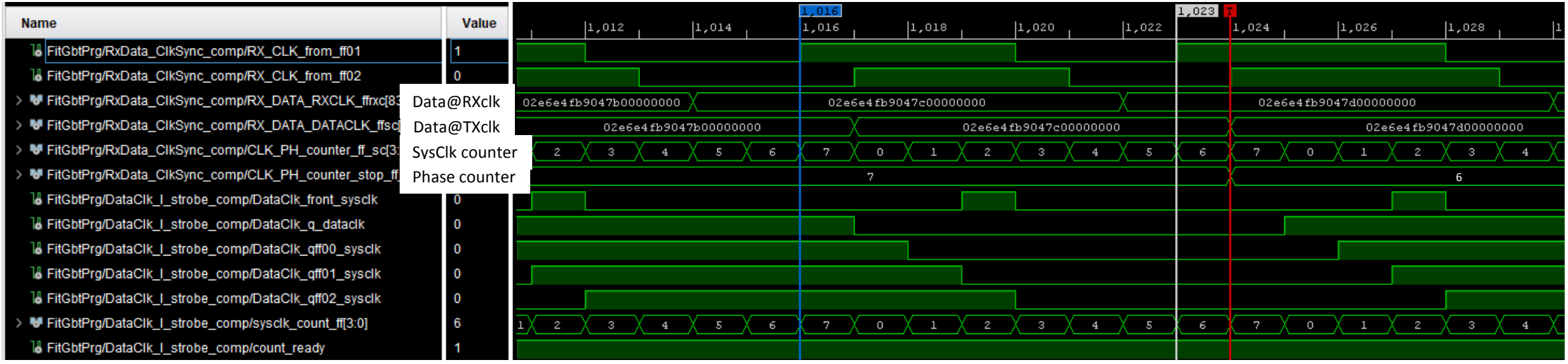
RX – TX domain crossing



Clock phase [ps] vs Ph counter

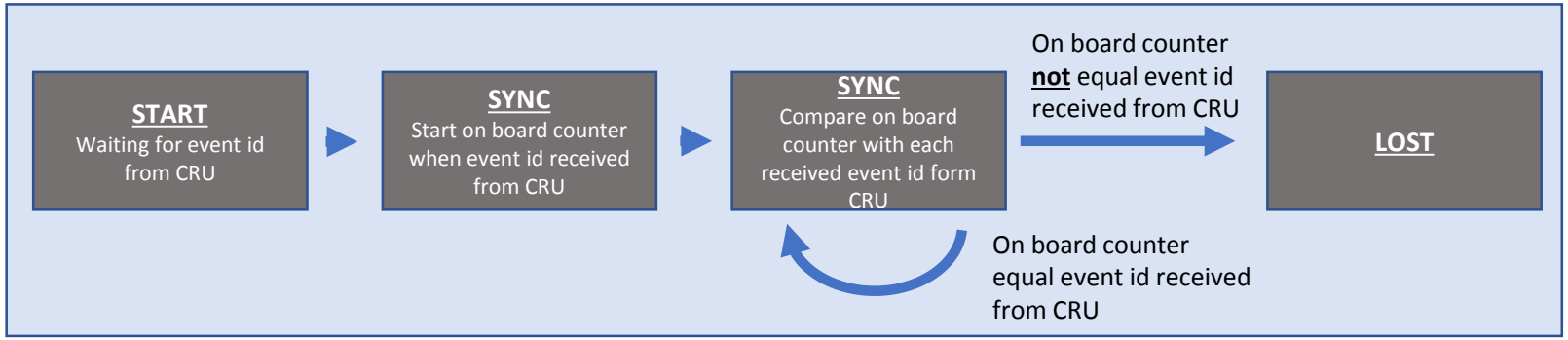


Chipscope @320MHz, triggered while phase crossing



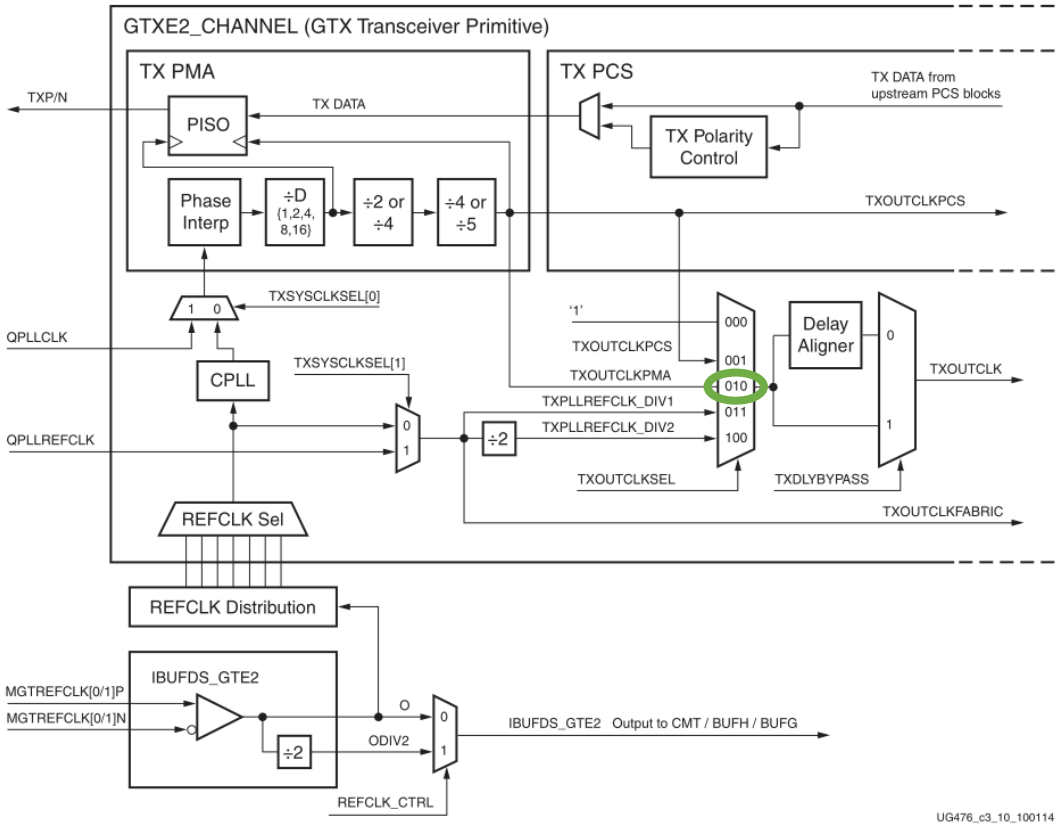
BC & Orbit Id Synchronization

- ~~CTP could not send event id each BC (CTP requirements 2017)~~
- Internal counter starts with first received event id (isData='1')
- Each time next event ID (orbit + bc) is received, the internal counter is compared to received ID
- In case of discrepancy the system goes to "lost" state and generates an error status



Changes in GBT project version 5_0:

1. MGT RefClk changed from 120 to 200 MHz
   1. CPLL_FBDIV changed from 4 to 3
   2. CPLL_FBDIV_45 changed from 5 to 4
   3. 120*4*5 = 200*3*4 = 2400 SerialClk
2. Project option changed to latency optimized only for RX
3. TX manually changed to standard version
   1. TX changed to STD version
   2. TX buffer enabled
   3. TXOUTCLKSEL changed from 011b to 010b
   4. PHASE_ALIGNMENT_MANUAL changed from TX_GTX_BUFFBYPASS_MANUAL_MULTILINK to false



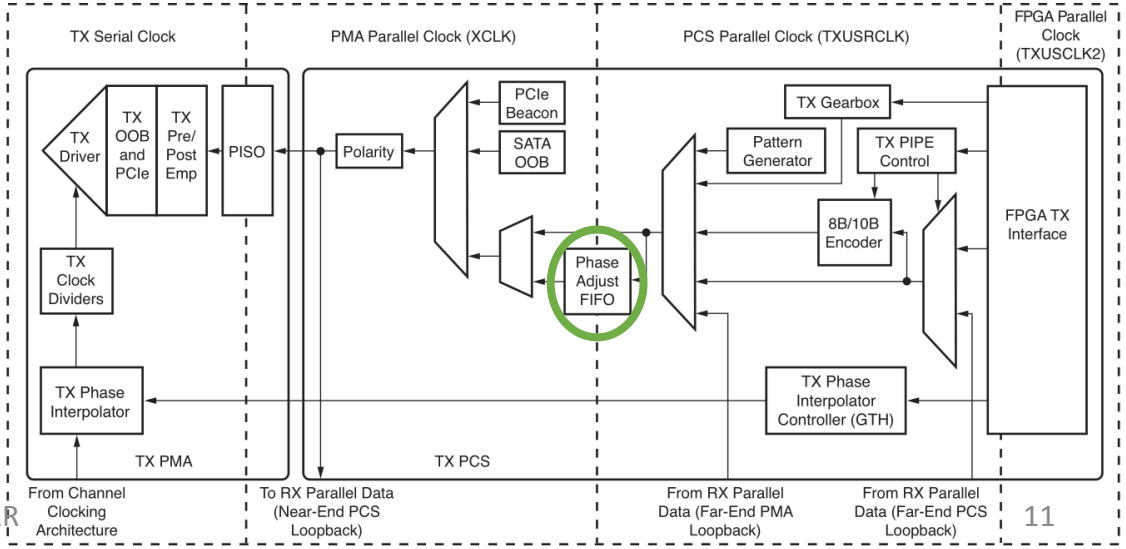


Table 2-8: **CPLL Divider Settings**

Factor	Attribute	Valid Settings
M	CPLL_REFCLK_DIV	1, 2
N2	CPLL_FBDIV	1, 2, 3, 4, 5
N1	CPLL_FBDIV_45	4, 5
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8, 16(1)

1. TX/RXOUT_DIV = 16 is not supported when using CPLL.

FIT firmware repository

- https://github.com/AliceO2Group/alice-fit-fpga
- Vivado projects compilation from tcl script
- Auto verification by Jenkinsfile
- ***Many thanks to Christoph Mayer***

AliceO2Group / alice-fit-fpga

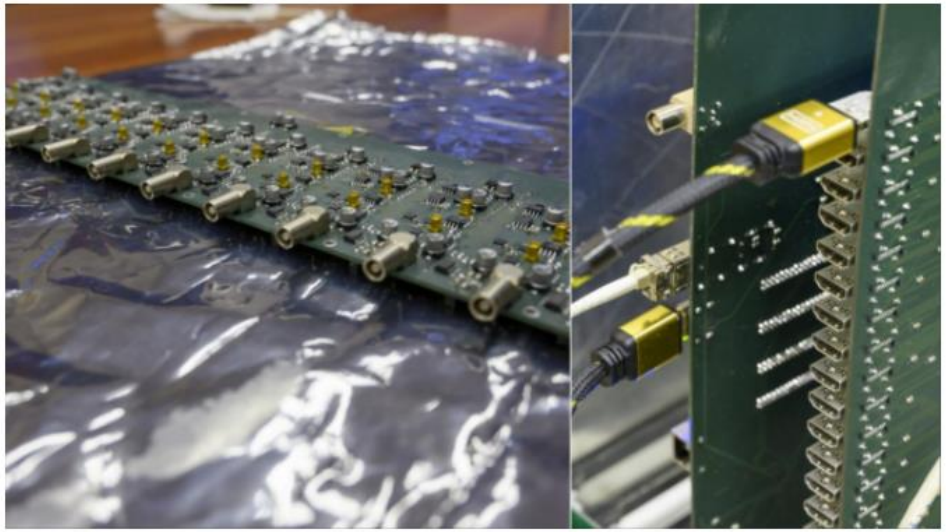
Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights

master | 2 branches | 0 tags | Go to file | Add file | Code

dfinogee Significant updates (#10) ... ca341bd 4 days ago 55 commits

firmware	Significant updates (#10)	4 days ago
software	Significant updates (#10)	4 days ago
.gitattributes	FIT FT0/PM 1st commit	10 months ago
.gitignore	Significant updates (#10)	4 days ago
.gitmodules	GBT-FPGA submodule added	10 months ago
Jenkinsfile	Significant updates (#10)	4 days ago
README.md	PM project updated	5 months ago

README.md



About

ALICE Fast Interaction Trigger (FIT) FPGA code

fpga-firmware | vhdl-code | vivado | gbt | cern | alice-experiment

Readme

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 2

dfinogee Dmitry Finogeev

hcab14 Christoph Mayer

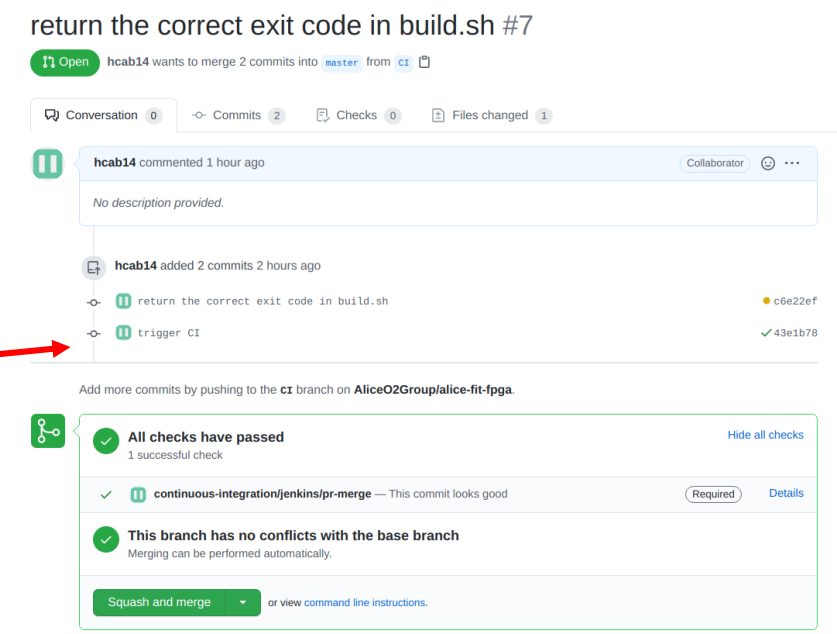
Languages

VHDL 87.1% | Tcl 8.0% | Python 3.2% | Verilog 1.7%

Continuous Integration - FIT FPGA

Slide from Christoph Mayer

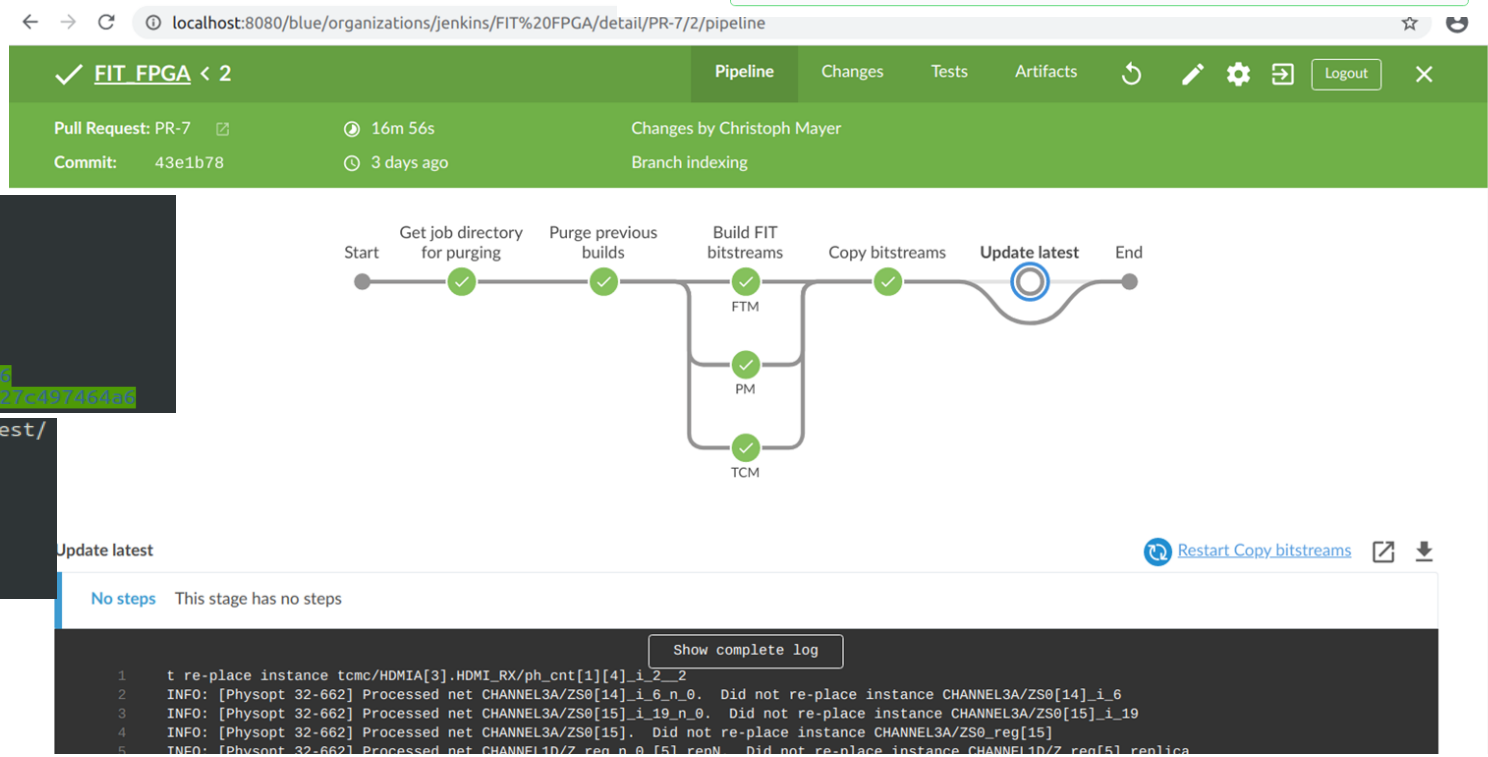
- **Jenkins** is running on FITSERVER2:8080
- Pull requests can only be merged if CI passes
- Whenever a new commit appears in GitHub, a complete rebuild is triggered
- Bitstreams and log files are put in a common directory



Branch name Git commit hash

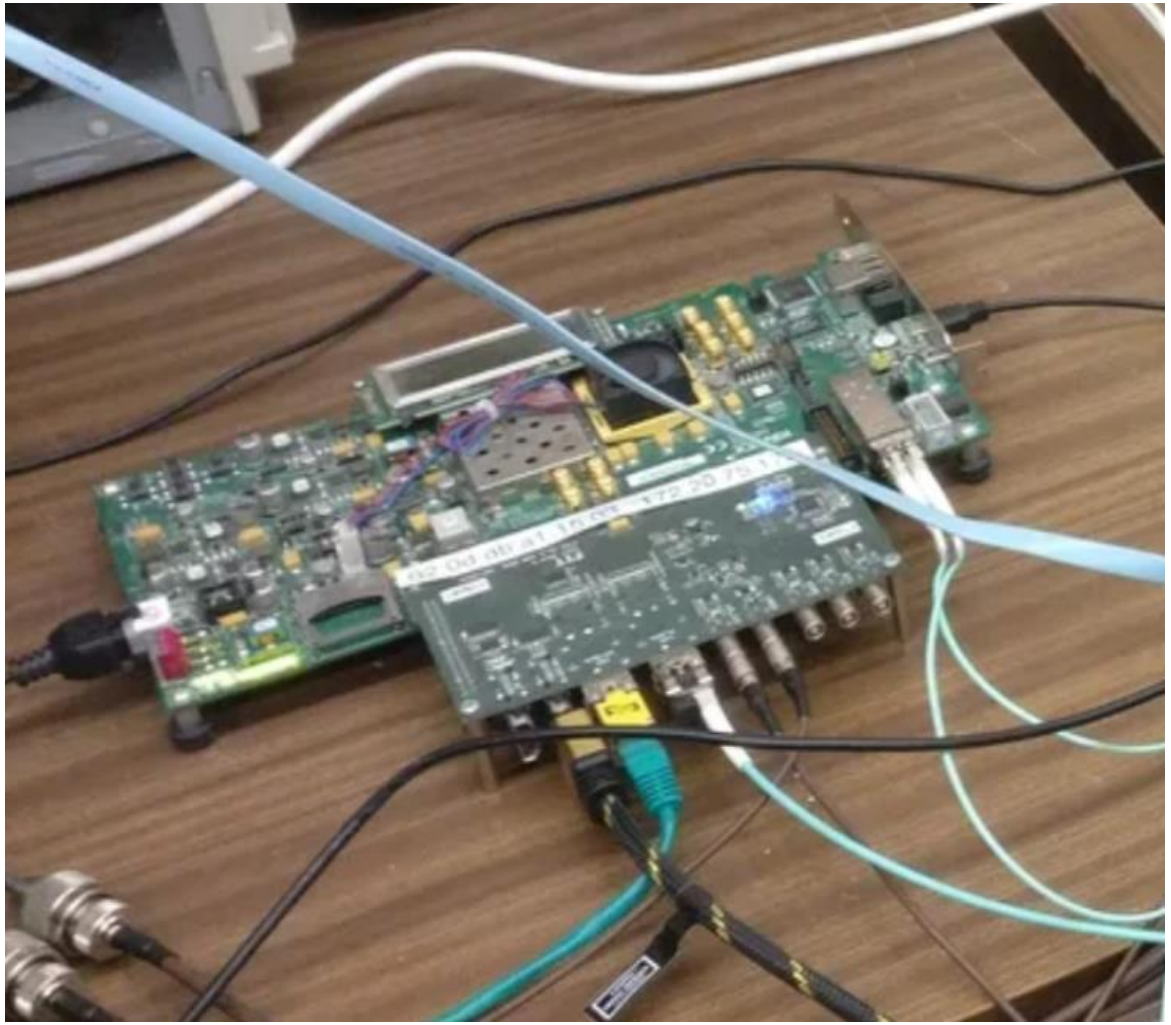
```
cmayer@FITSERVER2:/home/cmayer $ ls -lrt /media/sda1/fit-hardware-blobs/bitstreams
total 1
drwxrwxrwx 1 root root 144 Okt 20 14:05 CI-ace400ba665113c64c3f187510539df911f47f92
drwxrwxrwx 1 root root   0 Okt 20 14:58 CI-18c1652440dd6cb3da291a1403a38a6974770ba7
drwxrwxrwx 1 root root 176 Okt 20 15:32 CI-2ea5730d9b5107ee095e0eba1ae9c158df083d76
drwxrwxrwx 1 root root 384 Okt 20 16:57 CI-17150b71aaf6266436f78d622d5ea20e8f26ff2b
drwxrwxrwx 1 root root 448 Okt 20 17:51 CI-6a8f8535f7f82779b120c81994bad3ad448458f3
drwxrwxrwx 1 root root 448 Okt 20 18:17 CI-6fc53c8f38b2dbf08d4193a88d10b49d387e87da
drwxrwxrwx 1 root root 448 Okt 21 14:41 master-91ecf1ae6871274ce6ccbf65d7d5227c497464a6
lrwxrwxrwx 1 root root  47 Okt 21 14:41 latest -> master-91ecf1ae6871274ce6ccbf65d7d5227c497464a6
```

```
cmayer@FITSERVER2:/home/cmayer $ ls -lrt /media/sda1/fit-hardware-blobs/bitstreams/latest/
total 24484
-rwxrwxrwx 1 root root 11443726 Okt 21 14:41 FTM.bit
-rwxrwxrwx 1 root root  6692672 Okt 21 14:41 PM.bit
-rwxrwxrwx 1 root root  6692672 Okt 21 14:41 TCM.bit
-rwxrwxrwx 1 root root    68049 Okt 21 14:41 FTM_logs.tar.gz
-rwxrwxrwx 1 root root    94390 Okt 21 14:41 PM_logs.tar.gz
-rwxrwxrwx 1 root root    71443 Okt 21 14:41 TCM_logs.tar.gz
```



FIT TEST MODULE FW

- Includes readout firmware (two versions PM, TCM) for standalone tests
- Simulates CTP: triggers, BC and Orbit ID
- Generates laser start pulses
- GBT Readout
  - Receives data through GBT from PM, TCM
  - Sends received data to PC through IP-BUS
- Emulates TCM/PM HDMI connections for tests.
- Generates synchronous high quality clocks for TCM and LTU



I. FIT Readout Unit (description)

II. FIT Readout Software simulation

Readout simulation

- The goal is to perform software testing of firmware modules with Vivado simulation
- Firmware data and trigger generators are used, in hardware they are configured via IPbus
- Implemented with Python
- Included in git fpga repository

- Simulation workflow
  1. Software procedure generates readout configuration file
     - The file has readout control registers records for each 40MHz cycle
     - Includes generators parameters
     - Consists of chain runs with different parameters
  2. Vivado vhdl testbench load readout configuration and run behavioral simulation
     - Readout Status registers are stored in the file for each 40MHz cycle
     - GBT readout output is stored in the file
  3. Software macro to analyze testbench outputs
     - Configuration file analysis (checks types and configuration correctness)
     - Each run is analyzed with data from status and GBT files

  - 50 Mb files in total for 8 ms simulation (4 runs); 20 MB control + 30 MB status + 350KB GBT files
  - Allows to control firmware behavior on each cycle and checks control registers

Check list

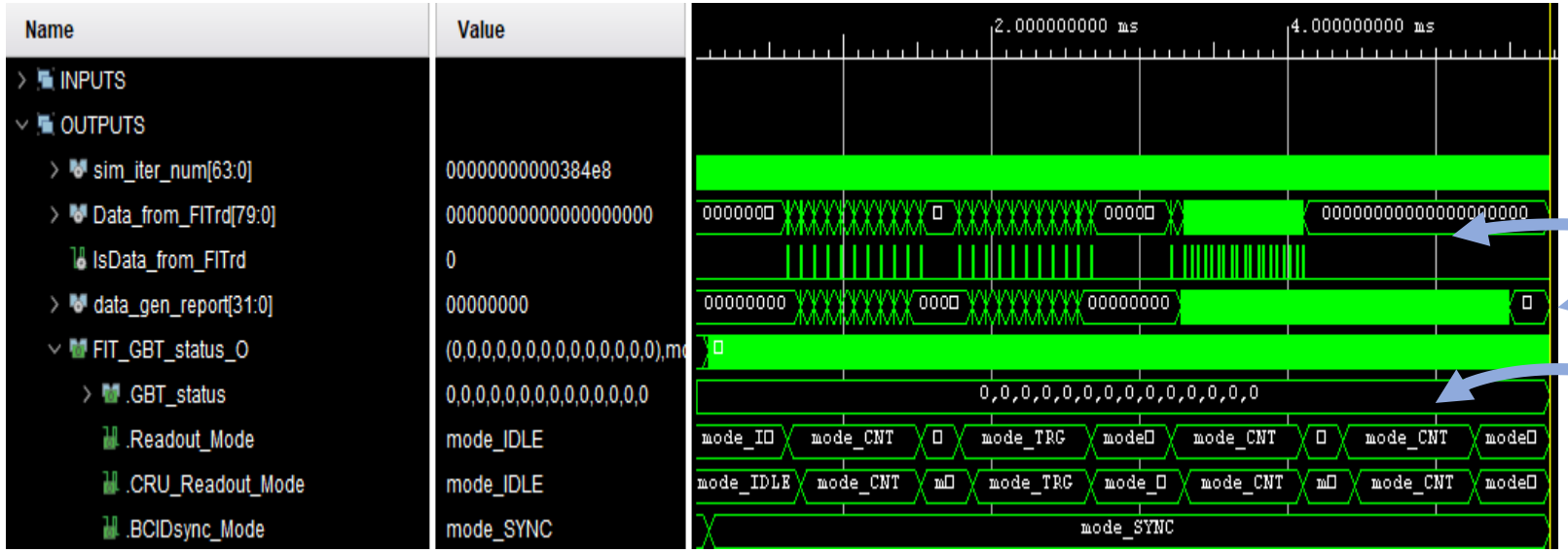
- RDH packet format
- SOR/EOR triggers presence in data
- Page counter
- Stop bit
- Accordance events orbit to RDH orbit
- Data integrity
  - All generated triggers and events saved in status file
  - Macro to check presence of events in GBT dataflow
  - Data selection in trigger mode is also implemented

- Dropped data in overload mode (to be implemented)

```
[INFO] ########################################
[INFO] ####### TESTING SIMULATION DATA #######
[INFO] ########################################
[INFO] Run N1
[INFO]
[INFO] Run rdh data successfully read ...
[INFO] Read 21 events
[INFO]
[INFO] Checking run ...
[INFO] First run trigger: 82 [True]
[INFO] Last run trigger: 102 [True]
[INFO] RDH page counters are correct ...
[INFO] RDH stop bits are correct ...
[INFO] Detectors orbits are correct ...
[INFO] Data integrity test [readout_cmd.trigger] ...
[INFO] Run orbits: [14 (72), 1e (143)]; total data packets: 71; selected data: 18
[INFO] All data in RDHs OK! ...
[INFO]
[INFO] !!! Run tested with 0 errors !!!
```

Debugging …

- During first tests with TCM CRU readout data wasn't sent occasionally
- Simulation with data generator with turned on before SOR evaluates the bug
- Bug was found in FSM of Data converter – only header was pushed to FIFO at the end of data taking



No GBT data

Data generator

Run with generator
turned on before SOR

```
[INFO]
[INFO] ########################################
[INFO] ####### TESTING SIMULATION DATA #######
[INFO] ########################################
[INFO] Run N3
[INFO]
[INFO] Run rdh data successfully read ...
[INFO] Read 0 events
```

No GBT data

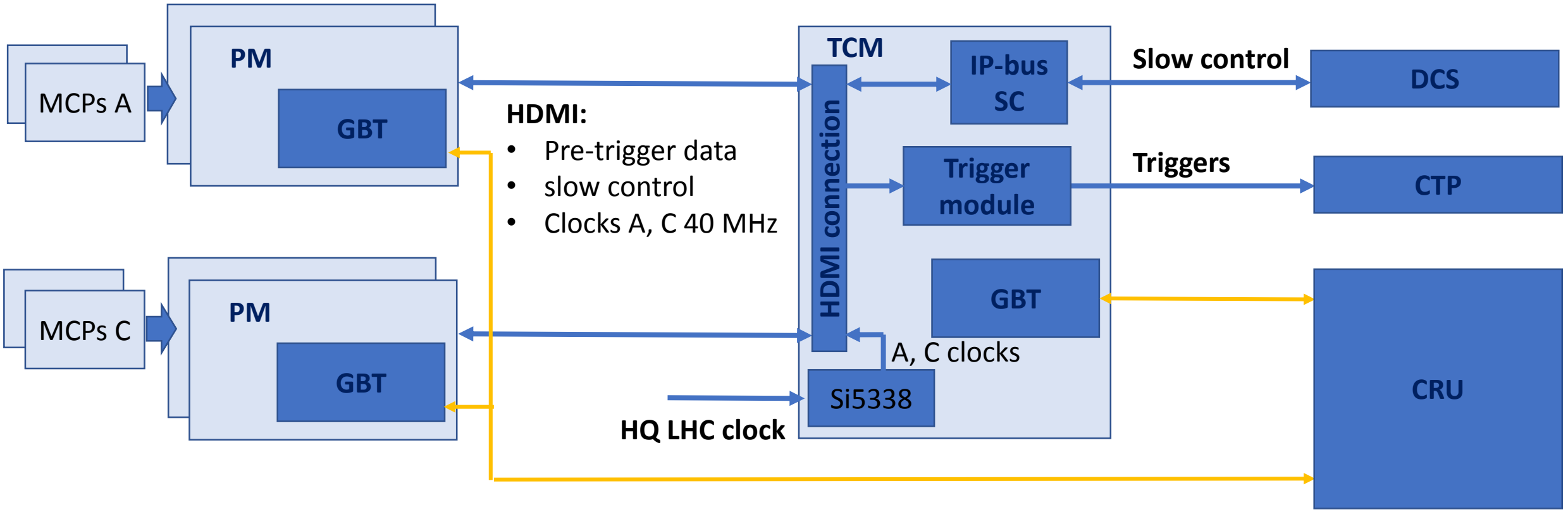
Conclusion

- FIT Readout firmware module is implemented and tested with PM/TCM + LTU + CRU
- Software data simulation and verification is implemented
- FIT FPGA projects are in git repository

Thank you for your attention

BACKUP

FIT (FT0) data flow structure



Fiber Optics connections:

- To CRU: readout data
- From CRU: trigger, event id

Readout Components functional description

- **Rx DataClkSync**
  - Synchronizes Rx data from CRU Tx to on board clock derived from HQClock
  - Calculates phase shift between clocks domain by clock 320MHz
- **Rx Data Decoder**
  - Decodes event id with BC and Orbit triggers
  - Synchronizes BC and Orbit counter with values, received from CTP
  - Corrects event ID for trigger message delay
  - Changes CNT/TRG/IDLE readout mode according to start/stop triggers and status bits
- **Data Converter**
  - Takes data from PM/TCM, build hit packet
- **Event Selector**
  - Selects hits and places them into CRU packet
- **CRU packet builder**
  - Builds CRU packet (adds header and trailer)
- **Test Generator**
  - Generates trigger data like CRU and emulates data from PM/TCM modules for stand-alone tests and simulations

Test generators features

- CTP/LTU generator
  - Generates Orbit/BC and triggers
  - Generates SOR/EOR triggers and status bits (by registers command)
  - Generates trigger with 64 pattern mask
  - Fixed frequency and fixed BC offset
  - Generator outputs (81bit GBT words) could be sent to readout or to RBT TX (LTU simulating)

- Detector data generator
  - Two data types: PM, TCM
  - 8 pattern mask, 4bit for data length each turn
  - Fixed frequency and fixed BC offset

- Trigger and data generators are synchronized by writing a command to the control register