CBC3 BE firmware on FC7

Electronics system meeting

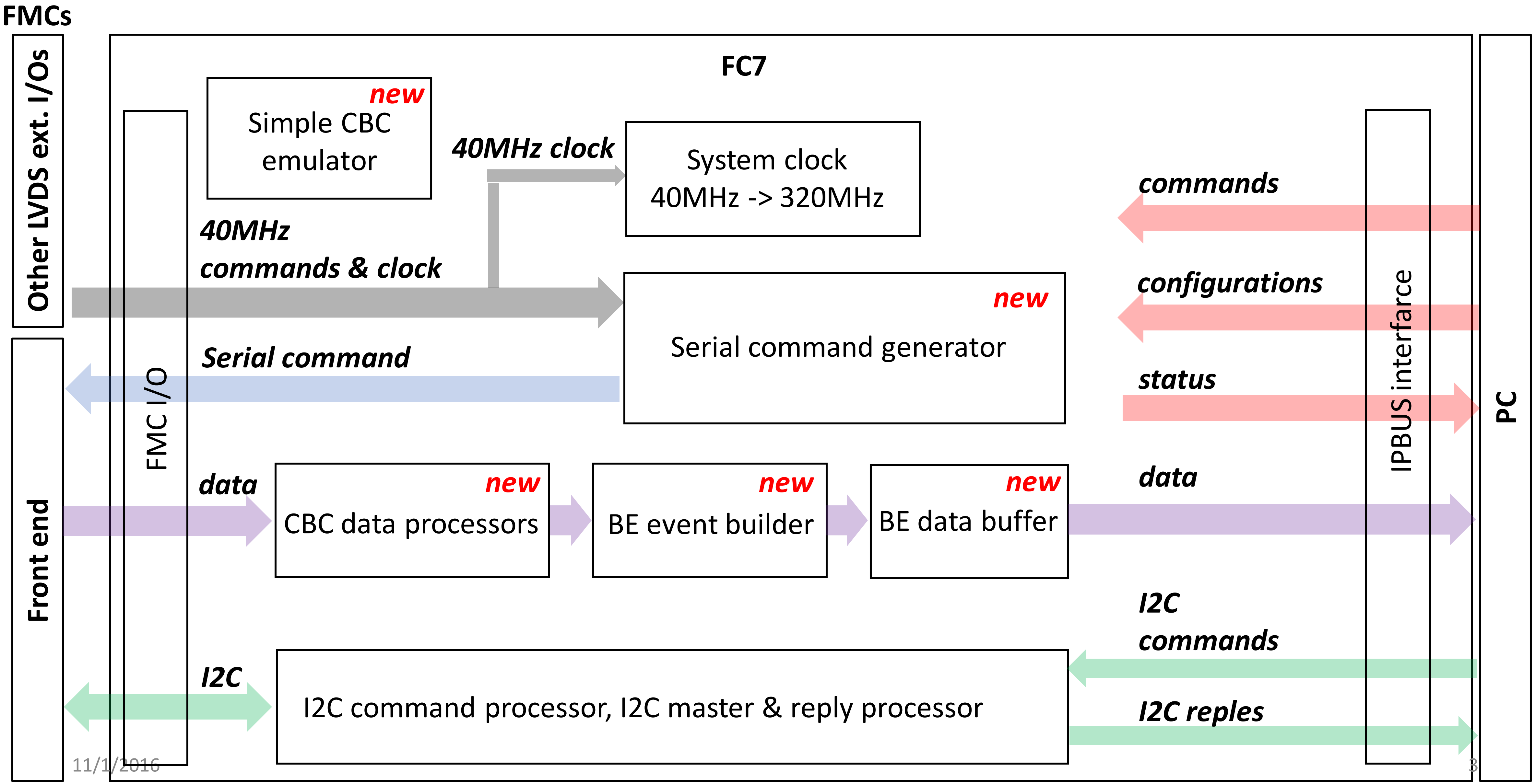
Kirika Uchida

CBC3 control & data readout firmware on FC7

- First version for single CBC3 expected to be available in Nov. and radiation tests in early 2017.
- Minimum system should be available in Nov.
    - I2C register read/write, fast control, data taking, simple CBC3 emulator (just to test the system)
- Optional functions are planned to be added later
    - Parameter tuning, full CBC3 emulator ?, CIC emulator ?.
- Some modules from CBC2 are used and some are newly developed
    - I2C module for CBC2 is used.
    - Serial command generator, serial data receiver, and variable length readout system.
    - The readout is divided into four different modules,
        - data segment(CBC or CIC processor) x N
        - event builder
        - data buffer(ring buffer)
        - ring buffer to ipbus fifo interface

    The data from data segment processors can be any length for each event so it is possible to process CBC data or any kind of data source can be connected if the given interface to the event builder is implemented.

Current backend board firmware overview



FMCs

Other LVDS ext. I/Os

Front end

FC7

FMC I/O

new

Simple CBC emulator

40MHz clock

System clock

40MHz -> 320MHz

40MHz

commands & clock

new

Serial command generator

Serial command

data

new

CBC data processors

new

BE event builder

new

BE data buffer

I2C

I2C command processor, I2C master & reply processor

commands

configurations

status

data

I2C

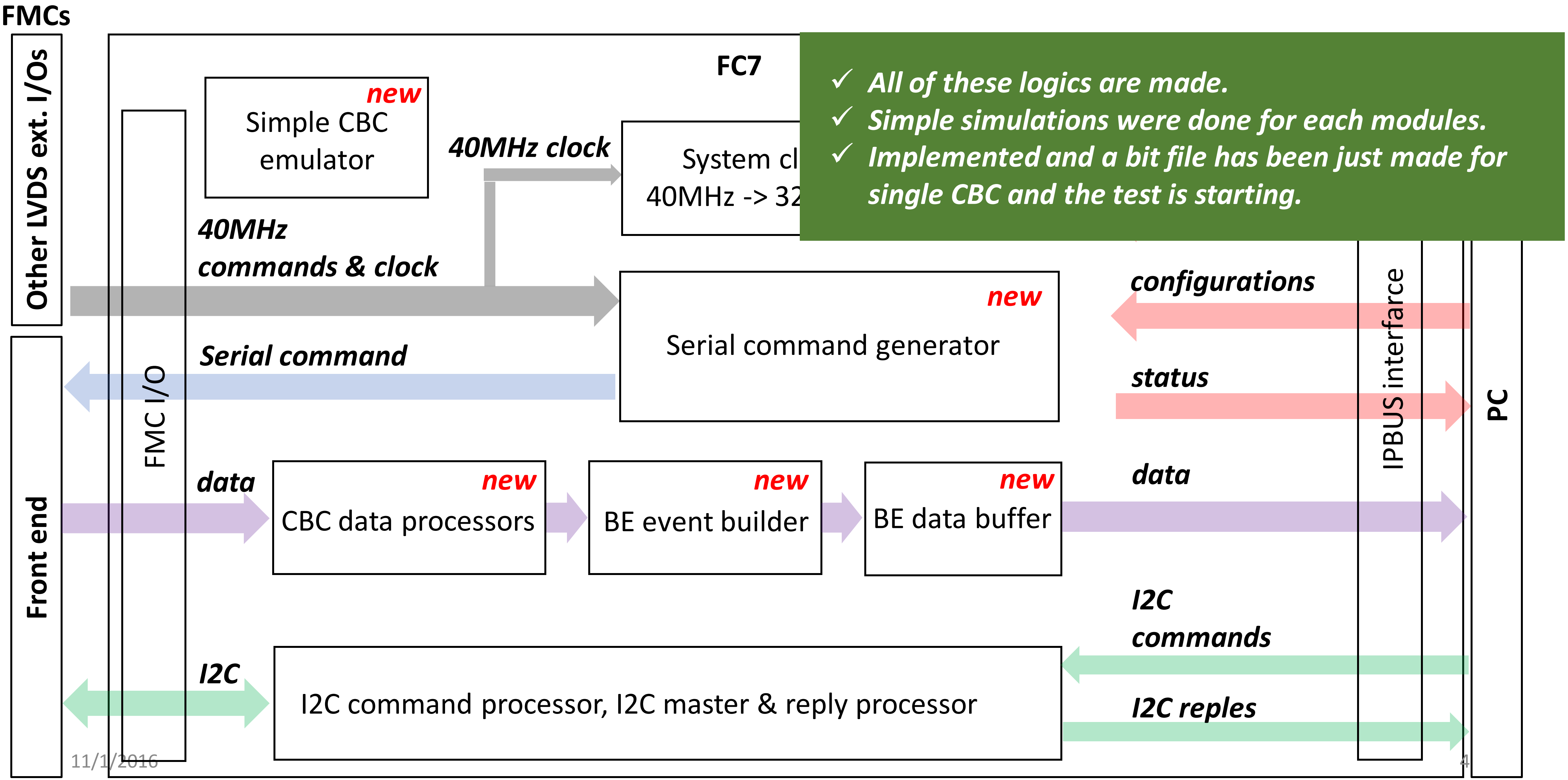
commands

I2C reples

IPBUS interfarce

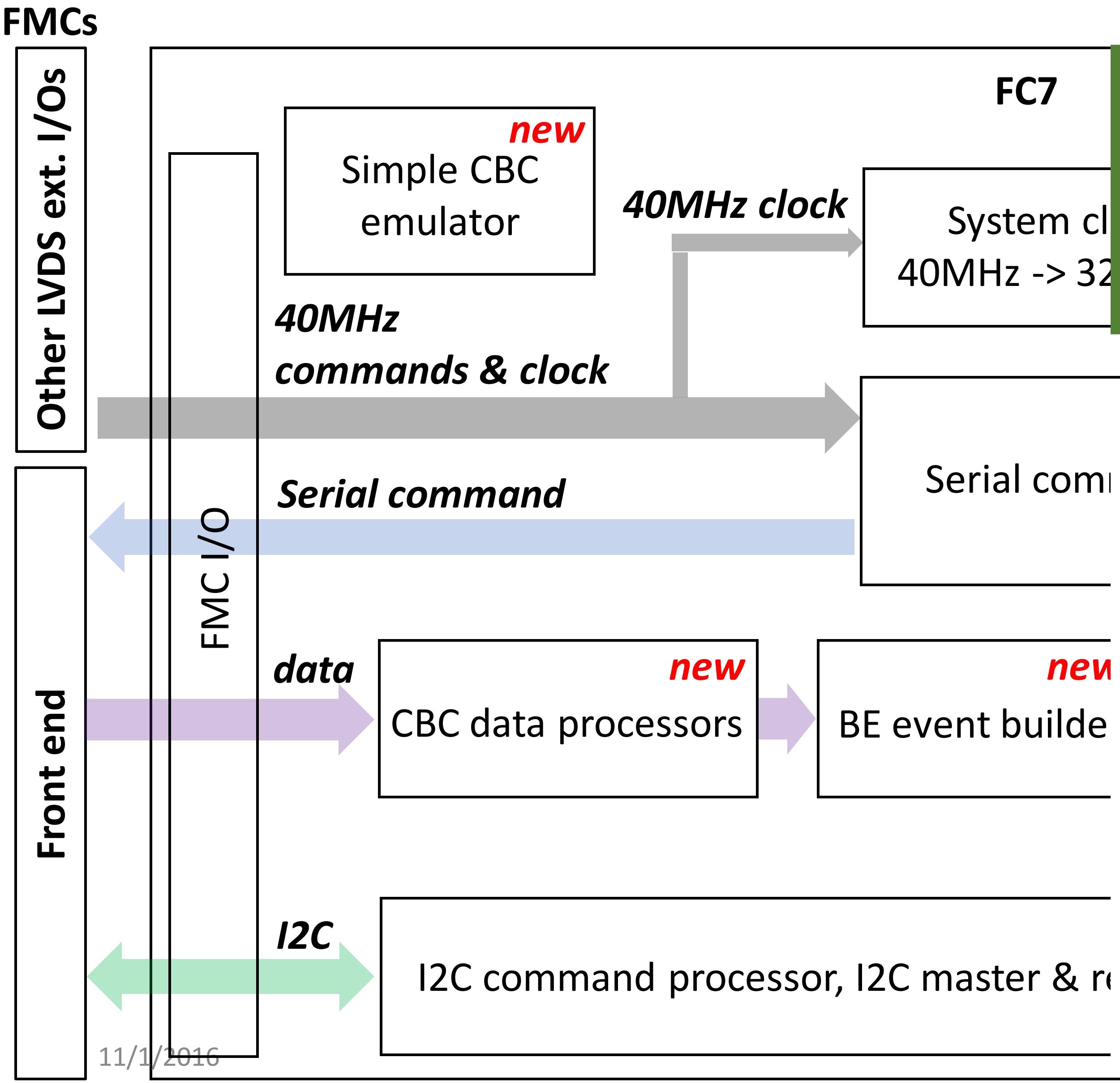
PC

Current backend board firmware overview



- ✓ *All of these logics are made.*
- ✓ *Simple simulations were done for each modules.*
- ✓ *Implemented and a bit file has been just made for single CBC and the test is starting.*

Current backend board firmware overview



FMCs

Other LVDS ext. I/Os

Front end

FC7

FMC I/O

new
Simple CBC emulator

40MHz clock

System cl
40MHz -> 32

40MHz commands & clock

Serial comı

Serial command

data

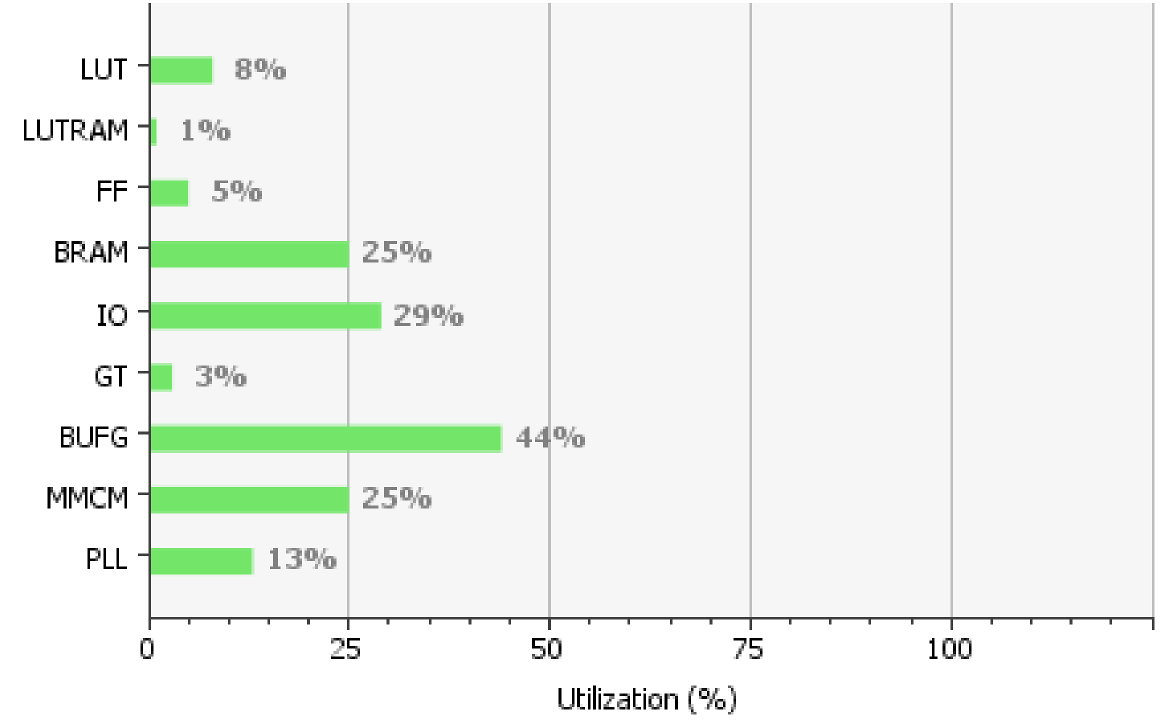
new
CBC data processors

new
BE event builde

I2C

I2C command processor, I2C master & re

- ✓ *All of these logics are made.*
- ✓ *Simple simulations were done for each modules.*
- ✓ *Implemented and a bit file has been just made for single CBC and the test is starting.*



FMC & clock system

- Physical connection to the frontend
  - FMC cards used for 2 CBC2 modules. Only one CBC3 can be connected.
- Clock system
  - IPBUS clock – 31.25MHz
  - MMCM
    - Input - 40MHz (on board or external through FMC)
    - Output – 40MHz, 320MHz

  The 40MHz is encoded in the serial command line.
  The stub data are stored in the 40MHz pipeline.
  I2C module run with the 40MHz.
  320MHz is used everywhere else.

Serial command generator

- Two external and internal 40MHz fast signal inputs can be independently enabled.
    - Ipbus
    - External LVDS in 40MHz parallel 4 signal lines with the clock.
      (This is enabled with selecting the external 40MHz clock is selected for the system clock source.)
    - periodic cycle of fast signals are generated internally.
- Output serialized commands to frontends.

I2C command processor, I2C master, & I2C reply processor

- 1MHz I2C clock is generated from 40MHz
- I2C register read/write commands are provided through IPBUS from PC and queued in FIFO.
- I2C master sends commands on FIFO to CBC and get replies.
- replies are put in FIFO interfaced to IPBUS for the PC to read.
- The broadcast at IPBUS side is implemented for CBC2 and it is still there, but IPBUS side per command is more than 100 times faster, so it is not very important.
- The FIFOs are very large this time. 4 CBC registers / bus commands can be sent at once.

CBC data processor

- SLVS signal deserializer
  - Data alignment with IDELAY & IDDR. Best possible timing tuning.
  - CBC has to be configured so that SLVS(5) line only send sync bits.
- SLVS data frame receiver & stub buffer
  - Stub information is buffered.
  - The raw data frame is identified and matched with the stub information stored in the stub buffer to create CBC event data.
  - *The frame data can be processed and hit data can be made.*
  - Make CBC event data packet with a header.
(header contains BE ID, FE ID, CBC ID, data)
  - Pass the CBC event data to the BE event builder.

BE event builder

- Receive CBC event data packet from multiple CBC data processors
- Make BE event packet adding BE counter data and a header (header contains FW ID, BE ID, # of CBC, CBC data type, data size)
- Pass the BE event packet to data buffer

BE data buffer

- Single ring buffer interface with an IPBUS read address.
- Write the size of available data on IPBUS registers,
  for all data and event block.
- Give free space information for back pressure.

Current status and the plan

- The minimum DAQ system has been implemented.
  (Resource usage for single CBC is tiny < 5% for LUT, 25% for block RAM)
- Modules except for CBC data processor has been passed simulations.
- The firmware can be loaded and test will start soon.
  - Performance check, data transfer rate, system stability, ease of use, etc.
- Data decoder is ready and CBC diagnosis software is under development.
- Georg is to implement the MW interface.
- Gitlab https://gitlab.cern.ch/kirika/Cbc3System/