# Status of the HGND readout development and integration in the BM@N DAQ.

***D. Finogeev***, F. Guber, A. Izvetnyy, S. Musin, D. Serebryakov
INR RAS, Moscow

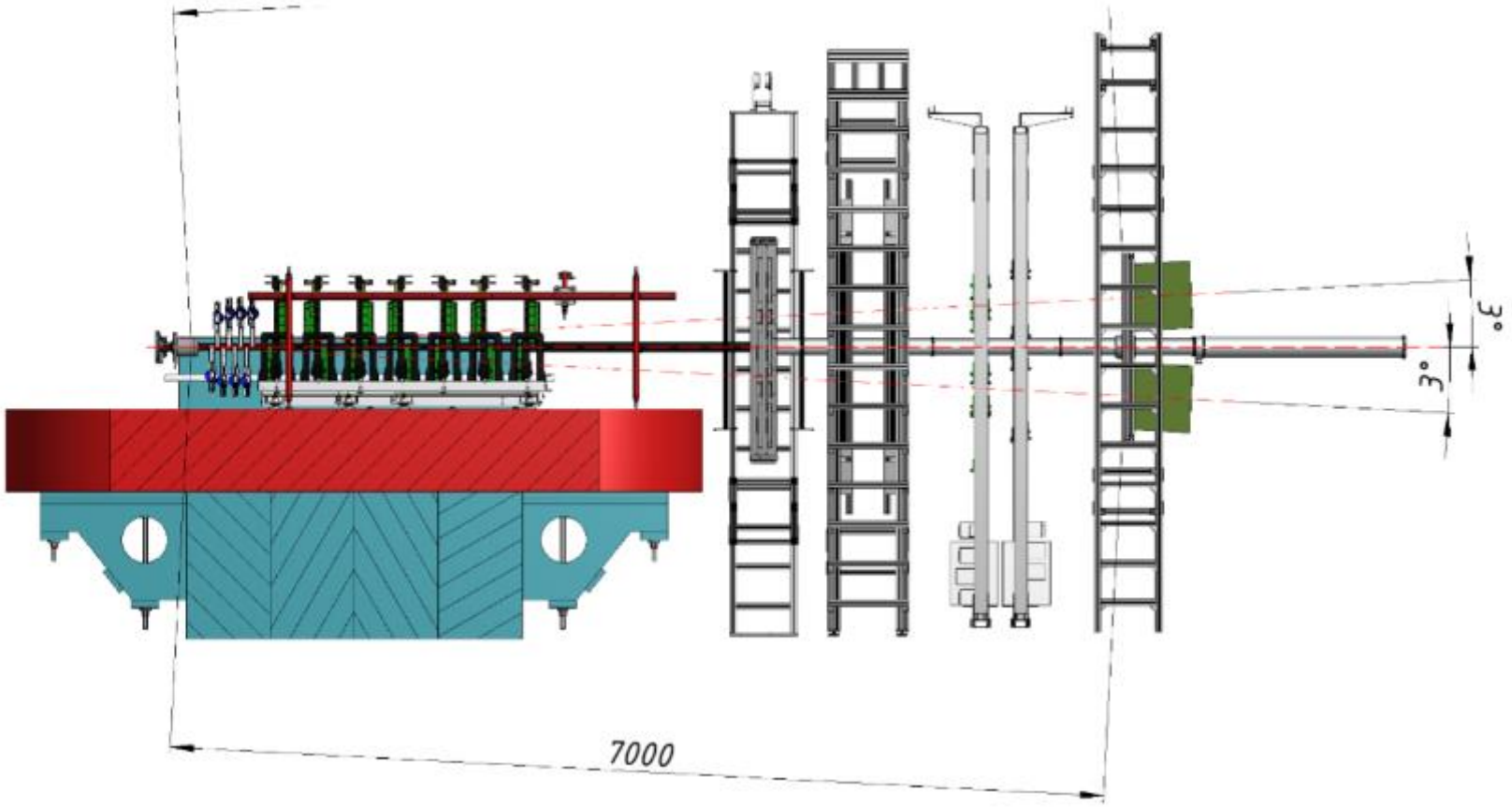
*Acknowledgements:*
*This work was carried out at the Institute for Nuclear Research, Russian Academy of Sciences, and supported by the Russian Scientific Foundation grant №22-12-00132.*

# Outline

- HGND readout topology
- 100ps FPGA based TDC
- Status of the FPGA firmware development
- Status of the readout board development
- Status of the DCS & readout software architecture

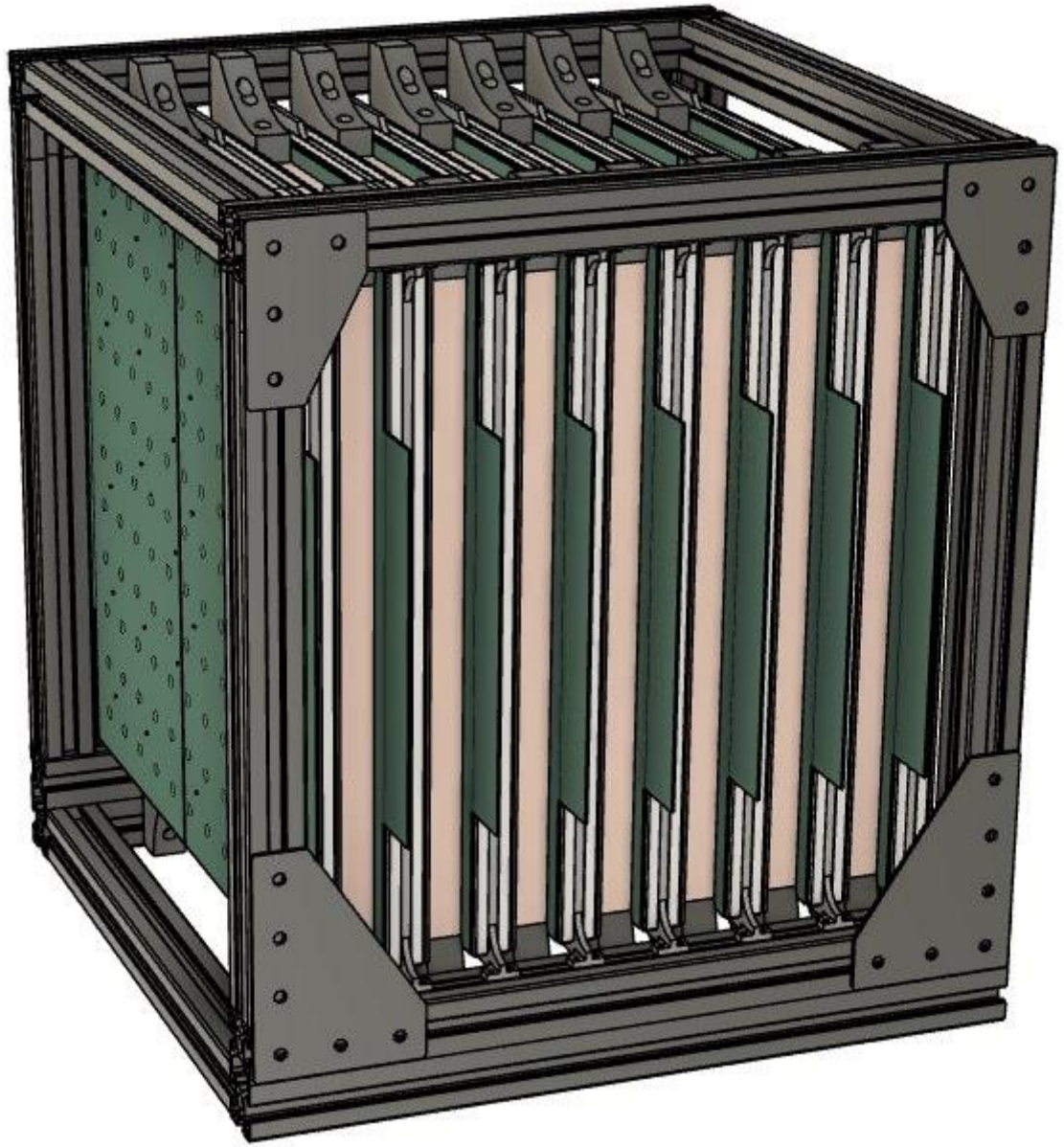
# Detector arrangement

- Detector for high-energy neutron flow measurement
- ToF method with T0 as the “start” signal source
- 7m measurement distance
- Detector is split into 2 “blocks” for improved acceptance



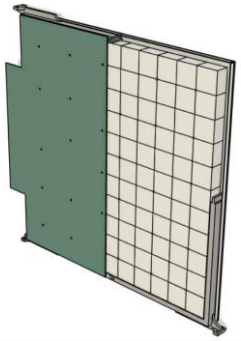
# Detector “block”

- Each block consists of:
  - A VETO-layer
  - 8 Cu absorbers
  - 8 sensitive layers
    - 11x11 grid of scintillatiors each
- Assembly is light-tight and air-cooled
- Framing is built with light-weight Al profiles

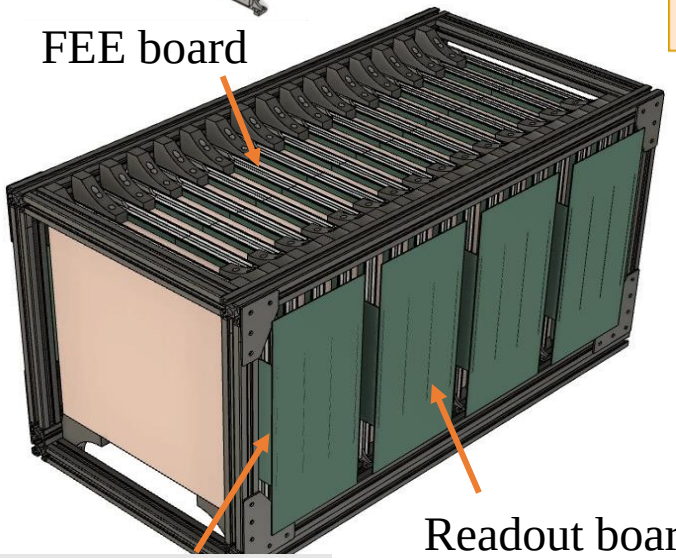


# FEE & readout architecture

- 16 layers with scintillation matrix 11X11
- 16 LED boards
- 32 FEE boards
- 8 Readout boards
- 3 FPGA per board
- 84 channels per FPGA
- 2000 channels in total

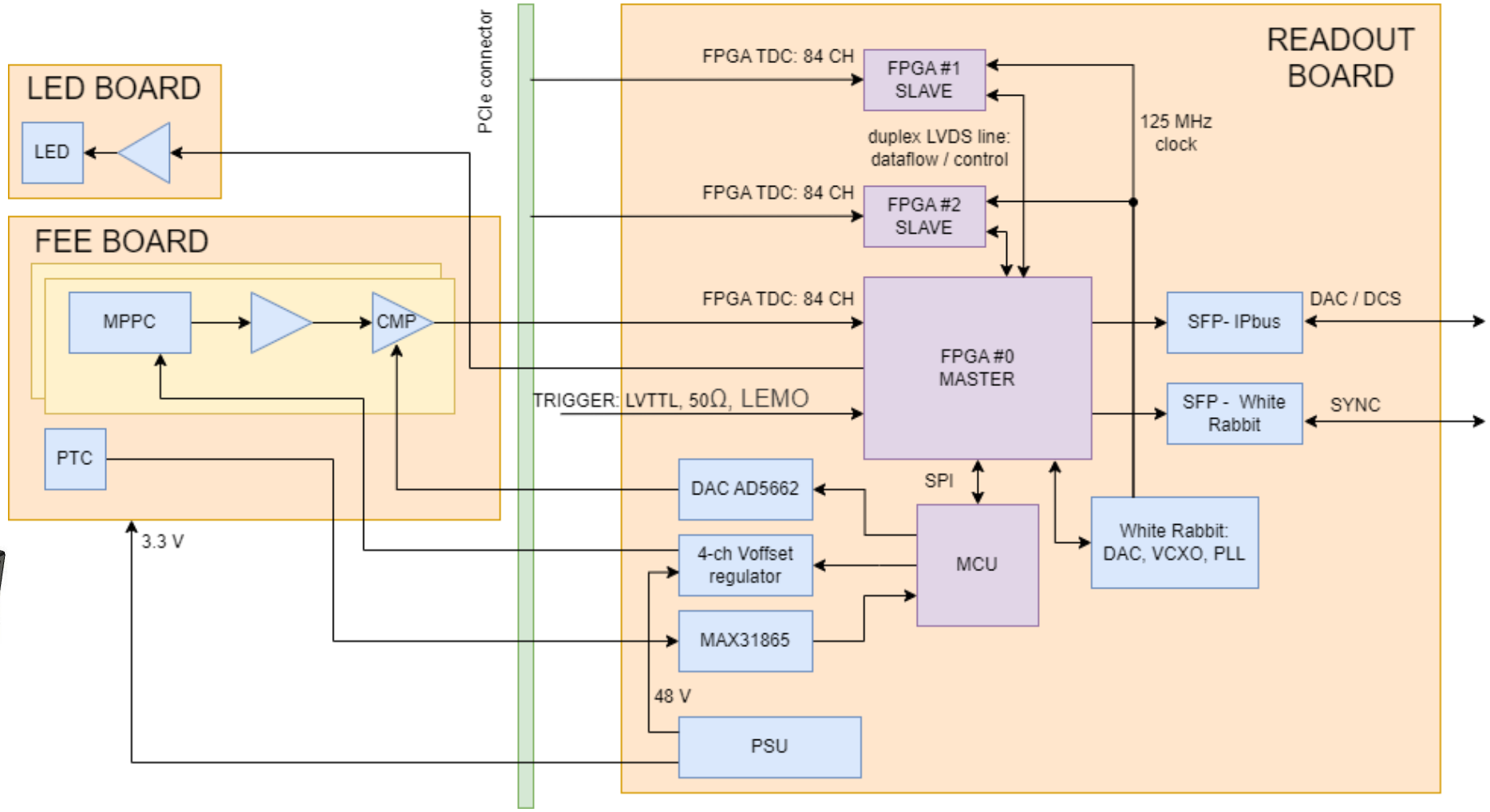


FEE board



Readout board

PCIe connector



# Readout & trigger

- ***100 ps*** TDC is implemented in Kintex 7 FPGA, 84 channels per FPGA chip (2000 total)
- ***White Rabbit*** (WR) is used for event's time synchronization (8 links total):
  - TDCs use clock sourced from WR synchronous to whole BM@N
  - WR timestamps are assigned to measured events

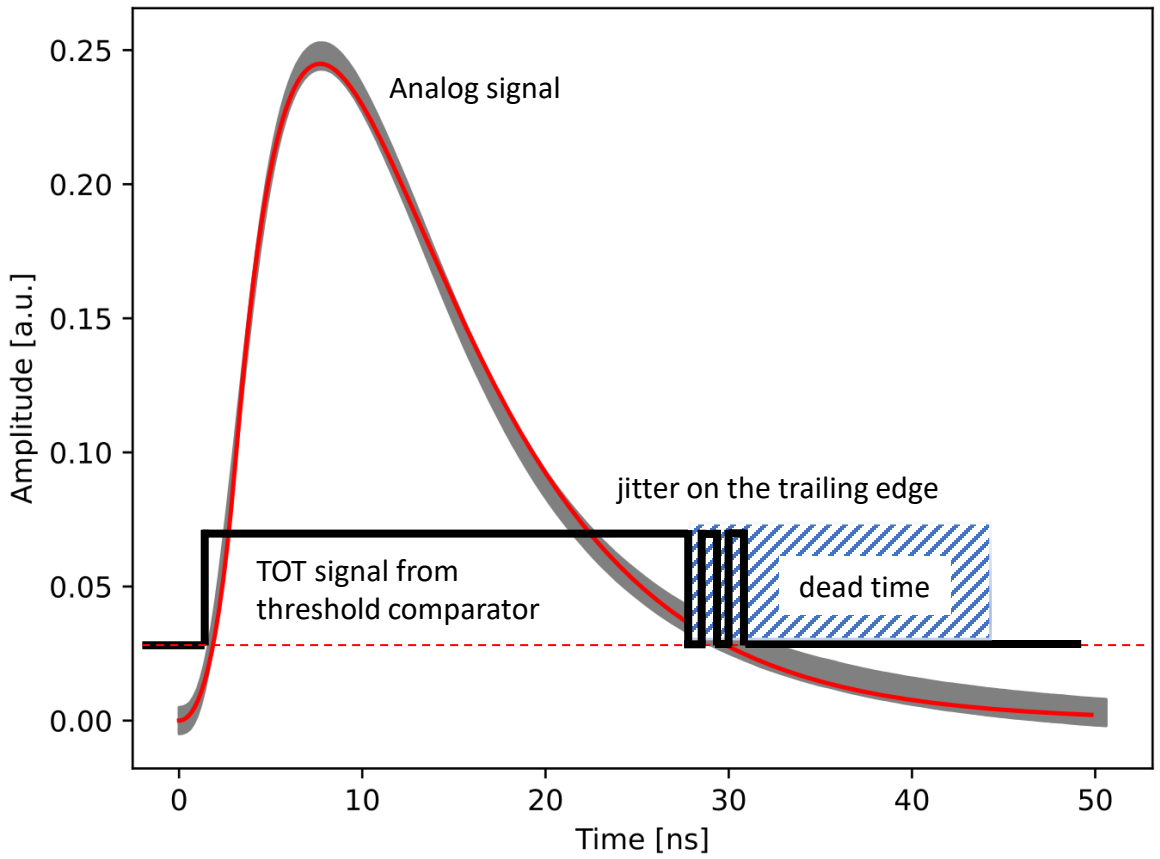
- Ethernet UDP protocol (***IPbus*** [1]) is used for data forwarding and board control
- Local network connect readout boards (8 ethernet links) with FLP

- The maximum HGND channel load is 3 kHz. The event size is 7x16 bits. The upper limit per link is not exceed ***100 Mbit/s***. ***The continuous readout*** is implemented without busy signal.

- The trigger is processed on FLP site:
  - Trigger signal is connected to TDC channel and digitized with WR timestamp in FPGA
  - Message trigger accompanied by a timestamp is transmitted to FLP for event selection

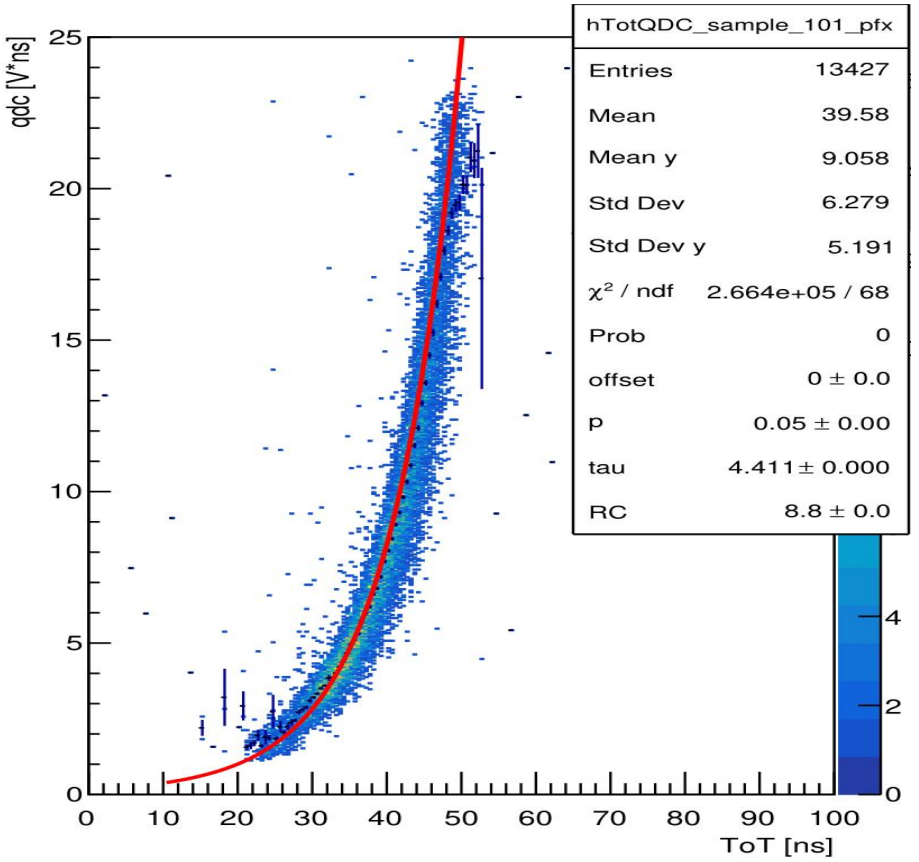
[1] C. Ghabrous Larrea, K. Harder, D. Newbold, D. Sankey, A. Rose, A. Thea and T. Williams, ***IPbus: a flexible Ethernet-based control system for xTCA hardware***, *JINST 10 (2015) no.02, C02019.*

# TDC Time Over Threshold (TOT)




- The threshold is tunable around 20 mV
- Signals length range is 20 – 60 ns
- Signals less than 6.4 ns are rejected for noise reduction
- Dead time is tunable in range 30 – 200 ns for comparator jitter filtering
- Minimum TOT time and dead time available in FPGA TDC are 3.2 ns

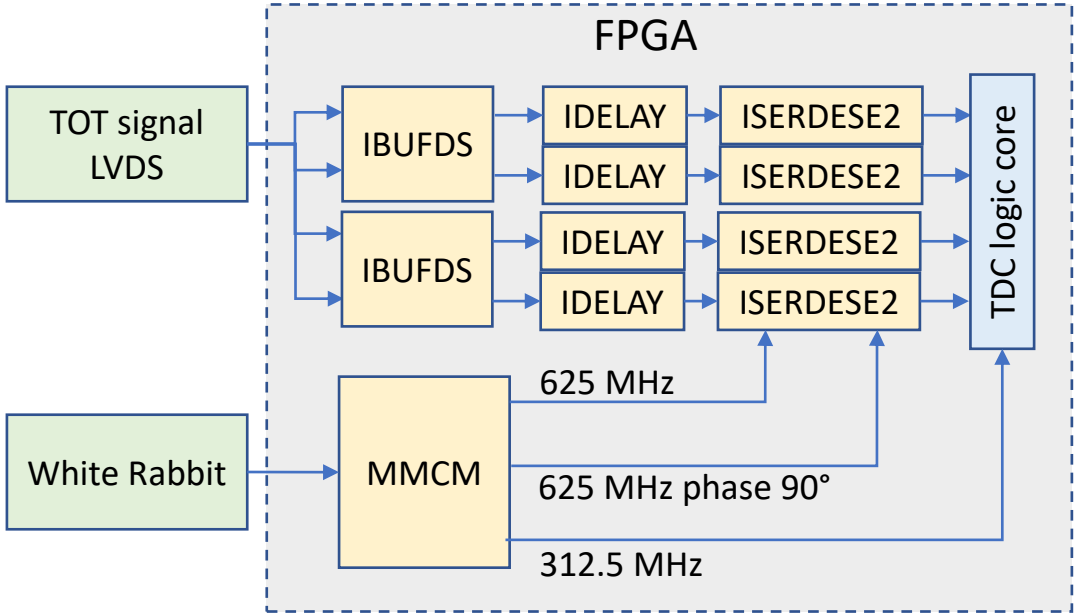
Amplitudes vs TOT time with analytical forecast

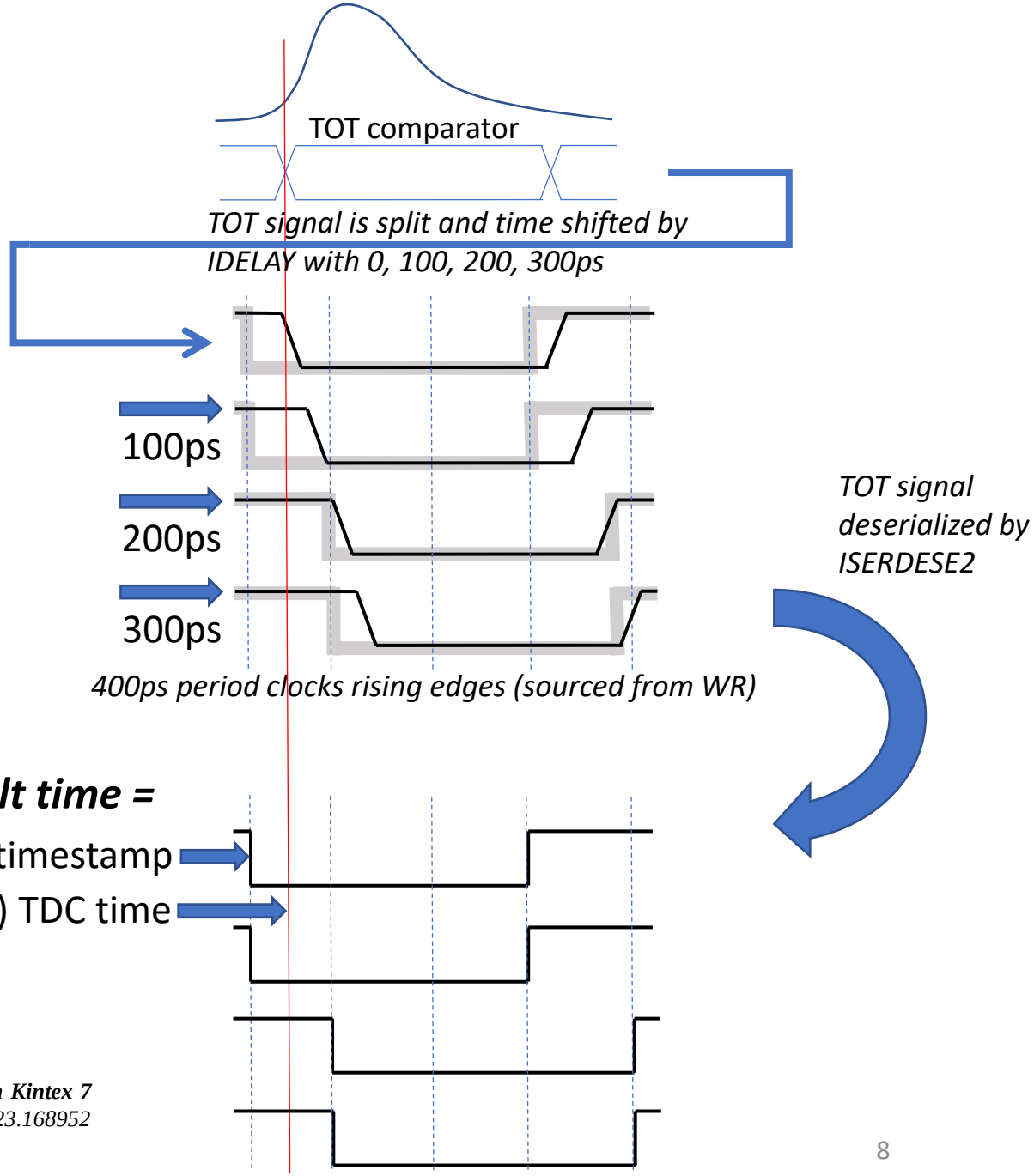



- TOT amplitude resolution is in range 14 - 22%
- Is used for time slewing correction


[2] N. Karpushkin, D. Finogeev, F. Guber, D. Lyapin, A. Makhnev et al., ***Analytical description of the time-over-threshold method based on time properties of plastic scintillators equipped with silicon photomultipliers,*** *DOI: 10.1016/j.nima.2024.169739*

# The 100ps FPGA TDC principle of operation



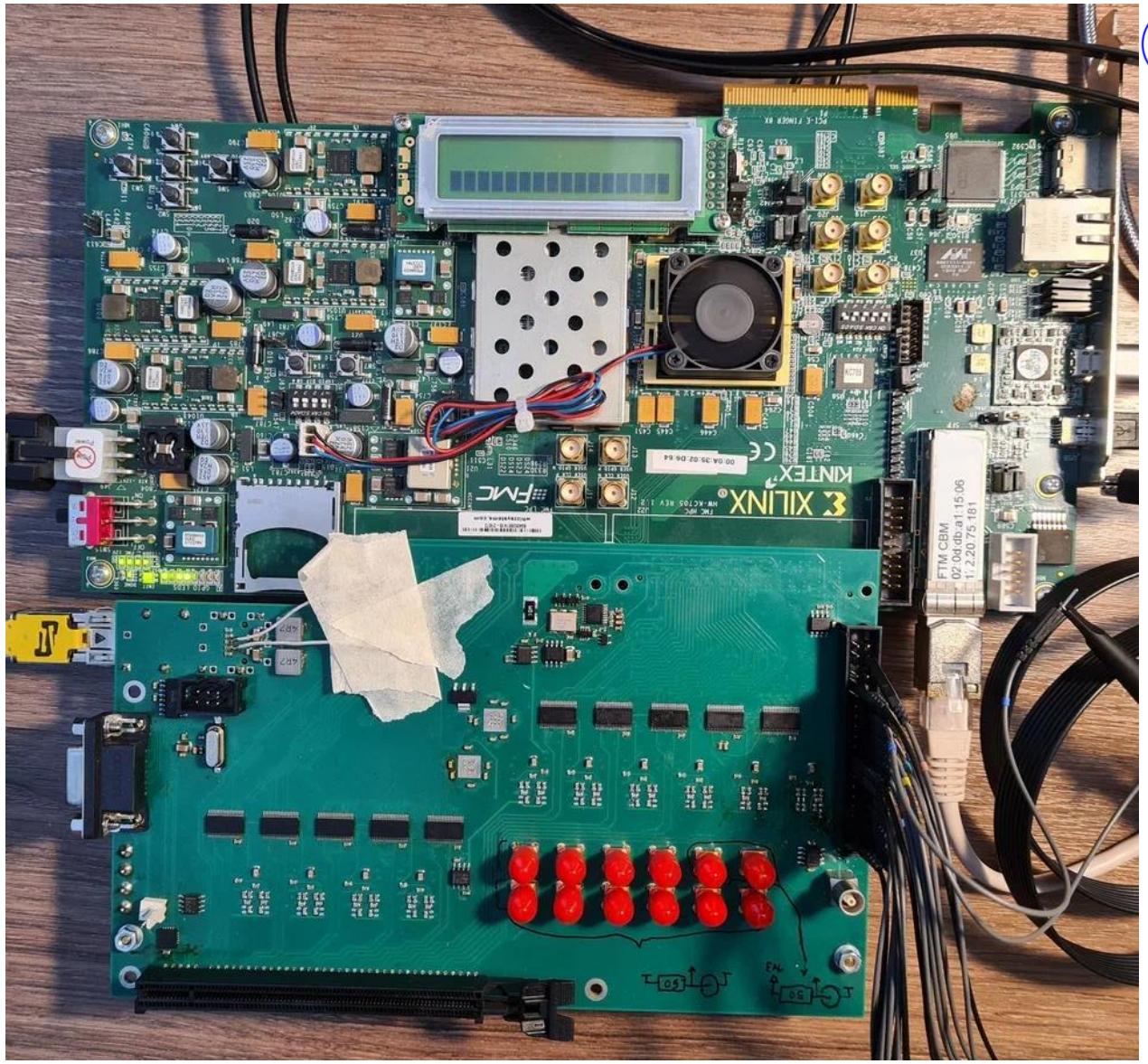



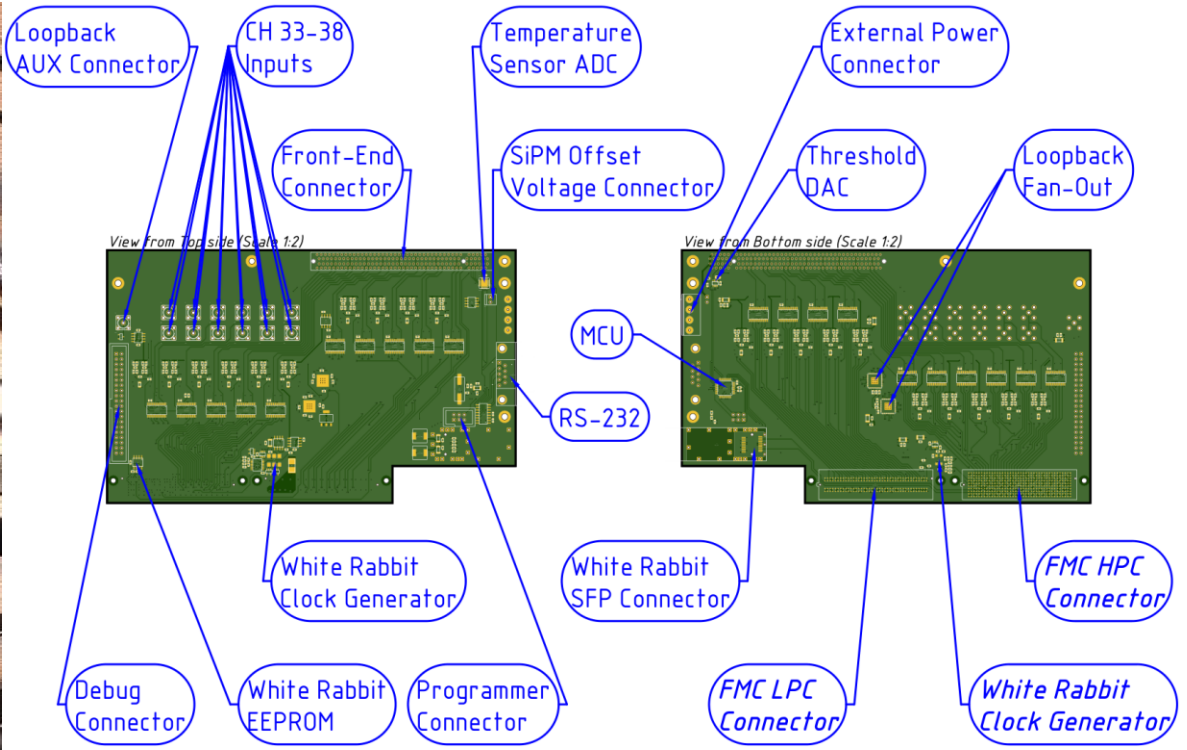

The TDC is **based on the Kintex-7** input serial-to-parallel converter with oversampling capability and programmable delay. The design is **based on Xilinx recommendations**, and uses **only documented features** of the FPGA within its specifications.


[3] D. Finogeev, F. Guber, A. Izvestnyy, N. Karpushkin, A. Makhnev et al., ***Development of 100 ps TDC based on Kintex 7 FPGA for the High Granular Neutron Time-of-Flight detector for the BM@N experiment,*** *DOI: 10.1016/j.nima.2023.168952*

# HGND readout v2 prototype (39 channels)

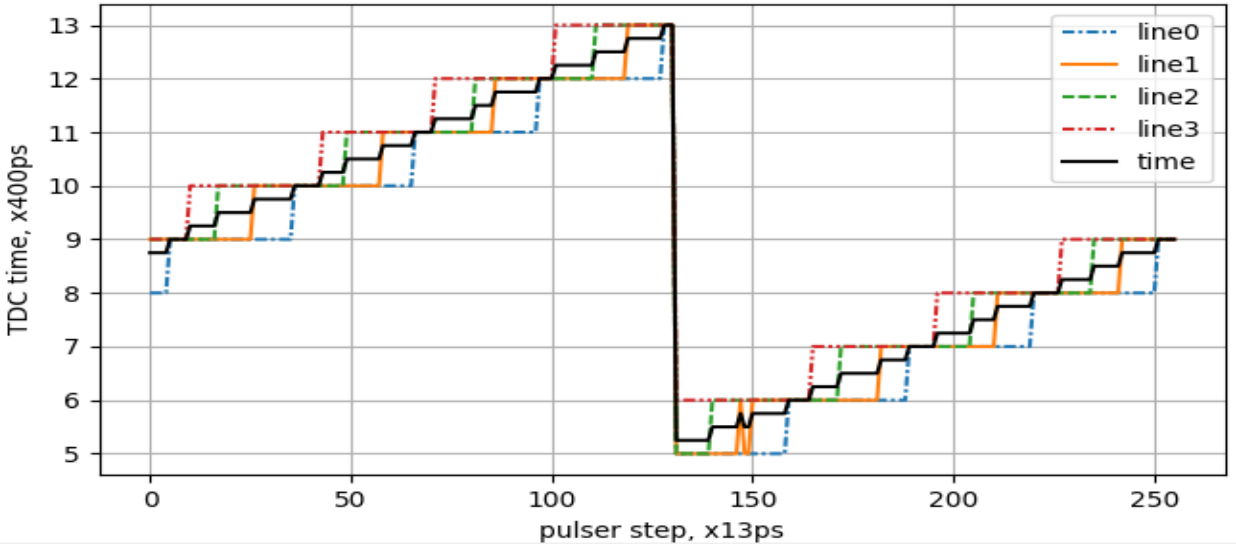
based on the Kintex 7 evaluation board (KC705)



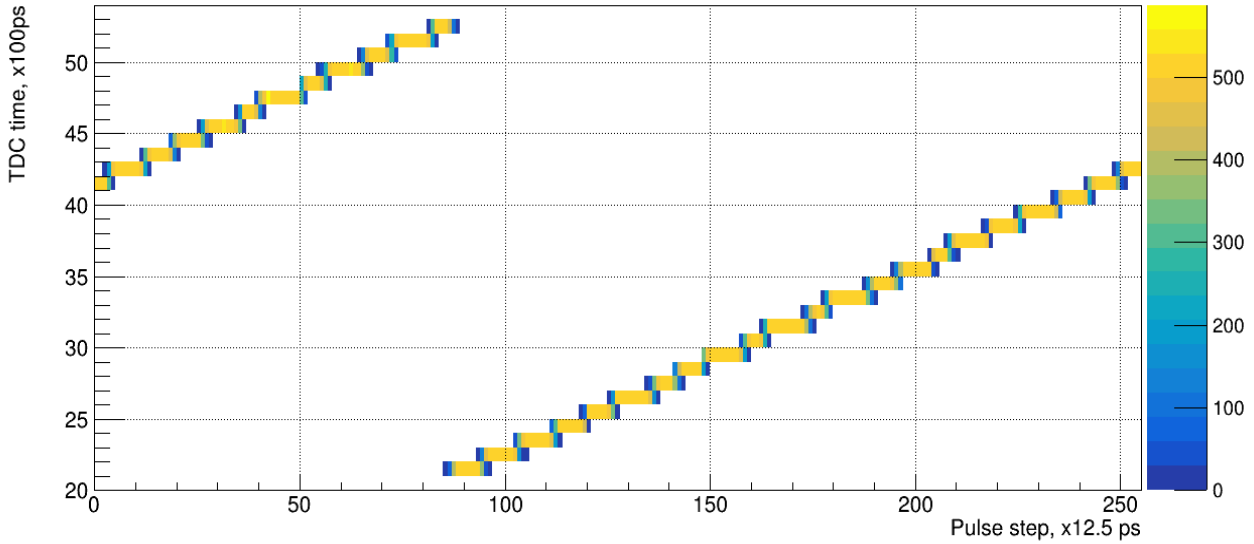



- Addon for Kintex 7 evaluation board
- 33 PCIe + 6 SMA TDC channels
- Ethernet readout
- White Rabbit synchronization
- FPGA loopback for TDC calibration
- Readout board functionality:
  - *PCIe connector for* scintillation matrix*, Temperature sensor, SiPM offset voltage control DAC threshold*

# The FPGA TDC time scan results



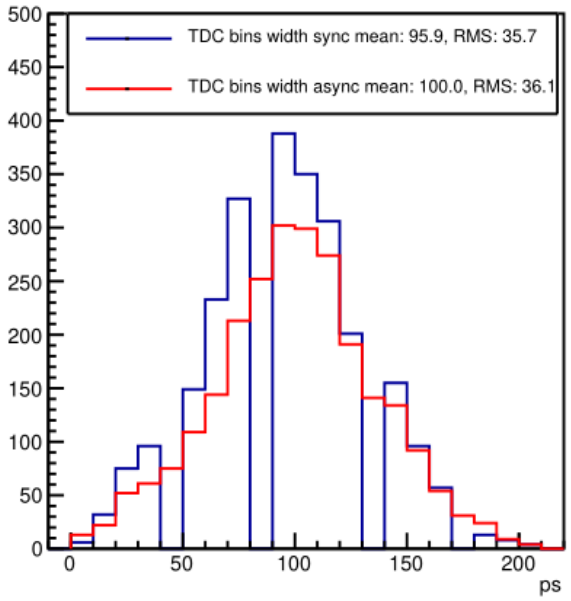

Four TDC lines and the resulting time dependence on the pulse time shift are shown. Pulses are generated by FPGA MMCM synchronously to the TDC clock with a phase step of 12.9ps. The single scan pass was taken with a digital FPGA logic analyzer.



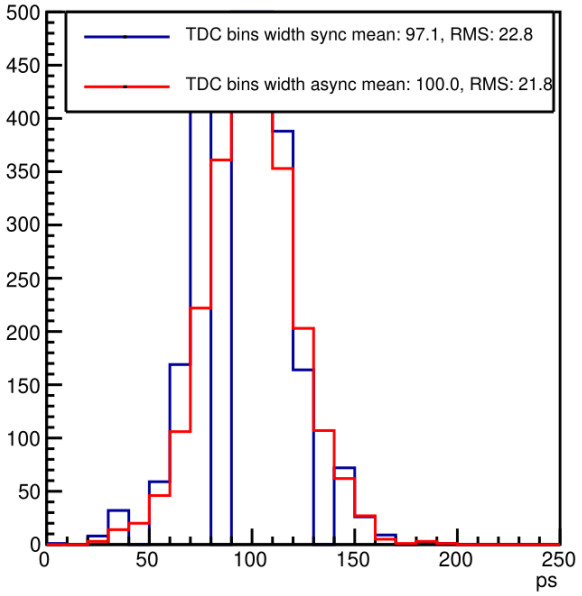

The TDC time dependence on pulse time shift. Pulses are generated by FPGA MMCM synchronous to TDC clock, and the time step is 12.9 ps. Data was taken with PC readout, 1000 events per single time shift step.

# New FPGA TDC alignment procedure

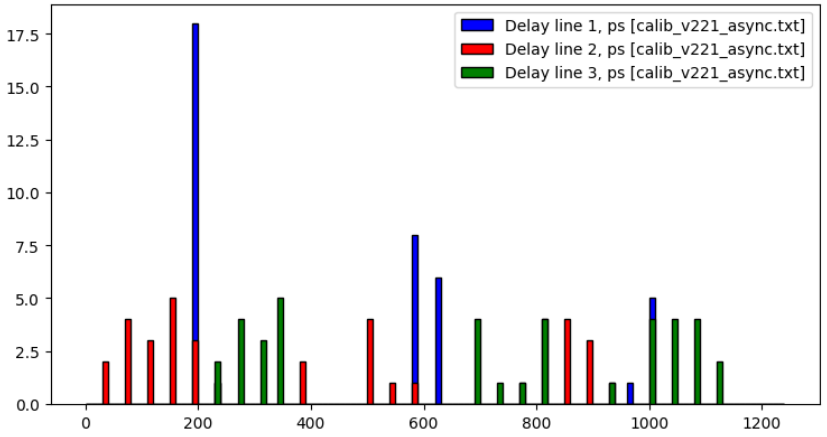
TDC alignment v1 (FPGA):
lines alignment algorithm



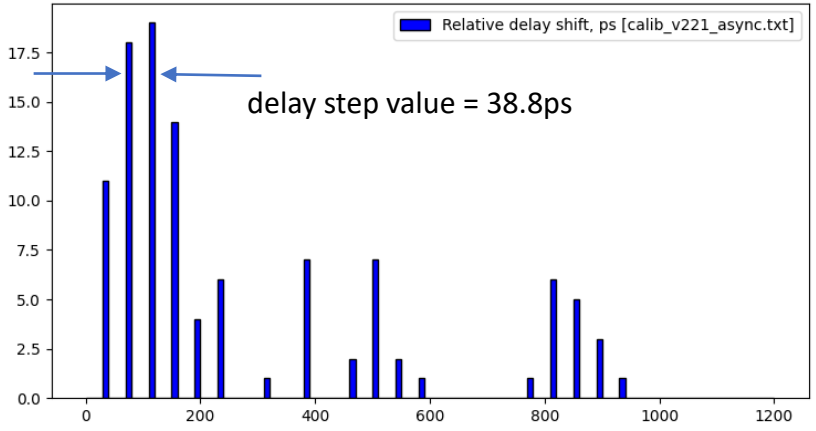

TDC alignment v2 (Software emu):
delays enumeration + density code test




TDC lines delays distribution, ps




TDC lines relative delays distribution, ps



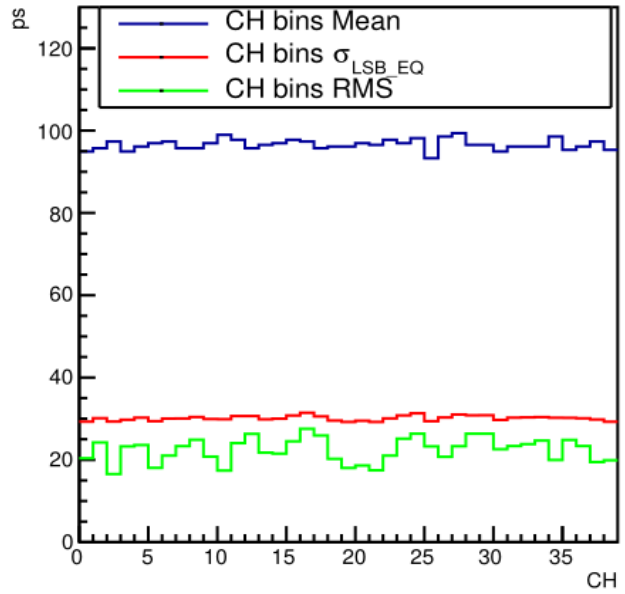

- New algorithm of TDC lines delay calibration was developed.
- Based on delays enumeration with code density test for verification.
- Requires minimum FPGA resources

➢ Working with raw TDC data in software for algorithms simulation.
❑ Will be implemented in FPGA TDC for auto-calibration procedure.
❑ The optimal delay step value should be found

# The FPGA TDC test results

**PROTO_V2**

TDC bins with for all 39 channels: mean value, RMS, equivalent LSB precision. Calculated with synchronous scan




**The equivalent LSB precision is ~30ps.**

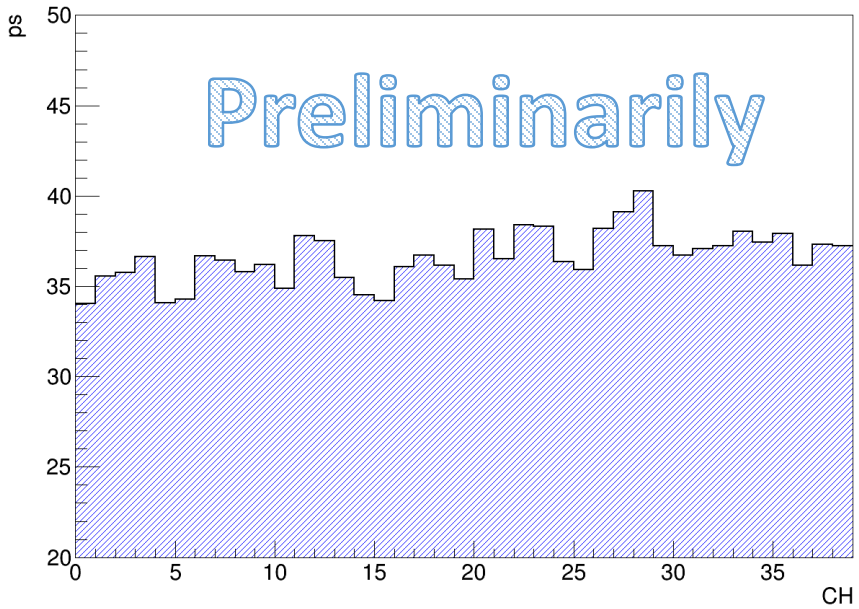
$$LSB_{EQ} = \sqrt{\frac{1}{\sum_{i=0}^{i=N-1} BinWidth[i]} \cdot \sum_{i=0}^{i=N-1} BinWidth^3[i]}$$

$$\sigma_{LSB\_EQ} = {LSB_{EQ}}/{\sqrt{12}}; \begin{cases} BinWidth[\forall i] = 100ps \\ \sigma_{LSB\_EQ} = 29ps \end{cases}$$

[4] N. Lusardi et al., "Quantization noise in non-homogeneous calibration table of a tcd implemented in fpga," DOI:10.1109/NSSMIC.2014.7431149

**PROTO_V2**

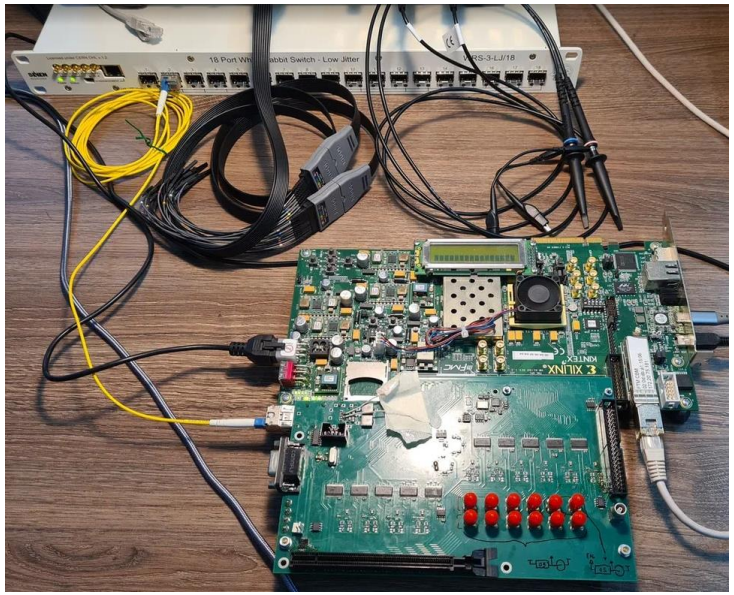
TDC bins precision measured with pulse length 101.2ns




**The measured TDC precision is 35 – 40 ps.**

**The time resolution of the scintillation cell is 130 ps**

# HGND White Rabbit synchronization



HGND WR test setup at INR
(Moscow) with provided WR switch

```
WR PTP Core Sync Monitor wrpc-v4.2-19-g54d3079
Esc = exit

TAI Time:                   Thu, Oct 3, 2024, 10:55:12

Link status:
wru1: Link up   (RX: 106, TX: 45) IPv4: 192.168.1.5 (static assignment)
Mode: WR Slave   Locked Calibrated

PTP status: slave

Synchronization status:
Servo state:                TRACK_PHASE
Phase tracking:             ON
Aux clock 0 status:         enabled

Timing parameters:
Round-trip time (mu):              910354 ps
Master-slave delay:                442861 ps
Master PHY delays:          TX:    220857 ps, RX:    245547 ps
Slave PHY delays:           TX:         0 ps, RX:         0 ps
Total link asymmetry:               24632 ps
Cable rtt delay:                   443950 ps
Clock offset:                           4 ps
Phase setpoint:                     10358 ps
Skew:                                   1 ps
Update counter:                        13
wrc#
```

HGND WR node monitor

```
WR Switch Sync Monitor v5.0.1-12-g553017c [q = quit]

WR time (TAI):      2024-10-03 10:56:27
Switch time (UTC): 2024-10-03 10:55:49
Leap seconds:  37
------------ HAL -----------|------------- PPSI -----------------------------
 Port | Link | WRconf | Freq |Inst| MAC of peer port   | PTP state | Pro | VLANs
------|------|--------|------|----|--------------------|-----------|-----|------
wri1  | down | Slave  |      |  0 |                    |           | R-W |
wri2  | down | Master |      |  1 |                    |           | R-W |
wri3  | down | Master |      |  2 |                    |           | R-W |
wri4  | up   | Master |      |  3 | 22:33:aa:55:aa:55  | master    | R-W |
wri5  | down | Master |      |  4 |                    |           | R-W |
wri6  | down | Master |      |  5 |                    |           | R-W |
wri7  | down | Master |      |  6 |                    |           | R-W |
wri8  | down | Master |      |  7 |                    |           | R-W |
wri9  | down | Master |      |  8 |                    |           | R-W |
wri10 | down | Master |      |  9 |                    |           | R-W |
wri11 | down | Master |      | 10 |                    |           | R-W |
wri12 | down | Master |      | 11 |                    |           | R-W |
wri13 | down | Master |      | 12 |                    |           | R-W |
wri14 | down | Master |      | 13 |                    |           | R-W |
wri15 | down | Master |      | 14 |                    |           | R-W |
wri16 | down | Master |      | 15 |                    |           | R-W |
wri17 | down | Master |      | 16 |                    |           | R-W |
wri18 | down | Master |      | 17 |                    |           | R-W |
Pro - Protocol mapping: V-Ethernet over VLAN; U-UDP; R-Ethernet

-------------------------- Synchronization status --------------------------
Master mode or sync info not valid

------------------------------- Temperatures -------------------------------
FPGA: 42.44 PLL: 26.38 PSL: 31.06 PSR: 33.94
HO: 0.94
```

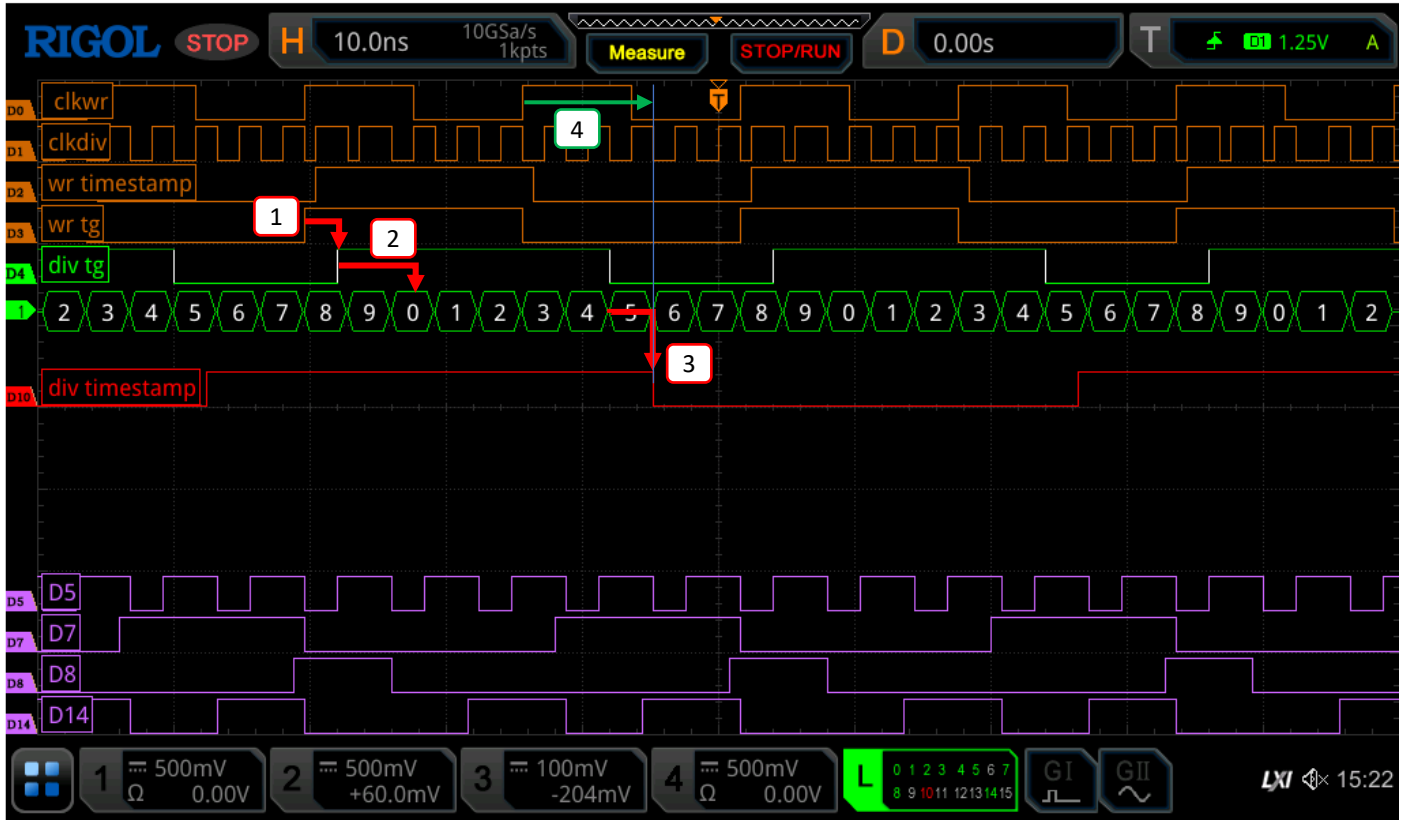
WR master switch monitor

## ***WR synchronization works:***

- ✓ WR switch connected
- ✓ The issue reason was wrong FIFO instance file version (for Virtex 6)
- ❑ WR design update to the latest v5.0 (optional)

***Thanks to Andrey Shchipunov for help***

# White Rabbit timestamp synchronization with FPGA TDC




WR clock 62.5 MHz -> TDC fabric clock 312.5 MHz

- Clocks are symphonious but phase shifted
- Clock frequencies ration is 5

1. WR clock toggle signal cross clocks domain
2. Counter 0 .. 9 starts by crossed toggle signal
3. WR timestamp cross clocks domain by counter
4. WR timestamp is latched by TDC fabric clock in stable range (falling edge) of WR clock

## TDC HIT timestamp format:

rejected (available in status REG)

64 bit event timestamp

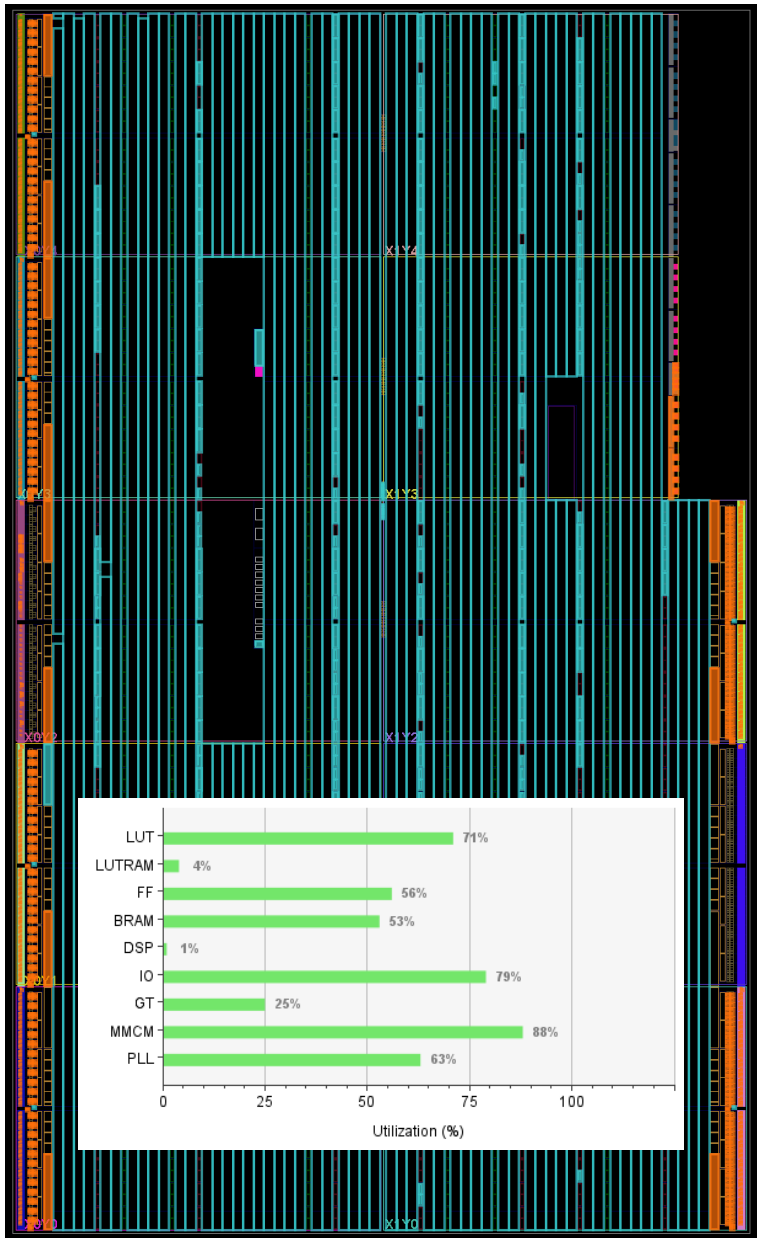
HIT rising / falling edges + pulse length 16 bit

timestamp bits: 63 | 31 | 30 | 4 | 3 | 0

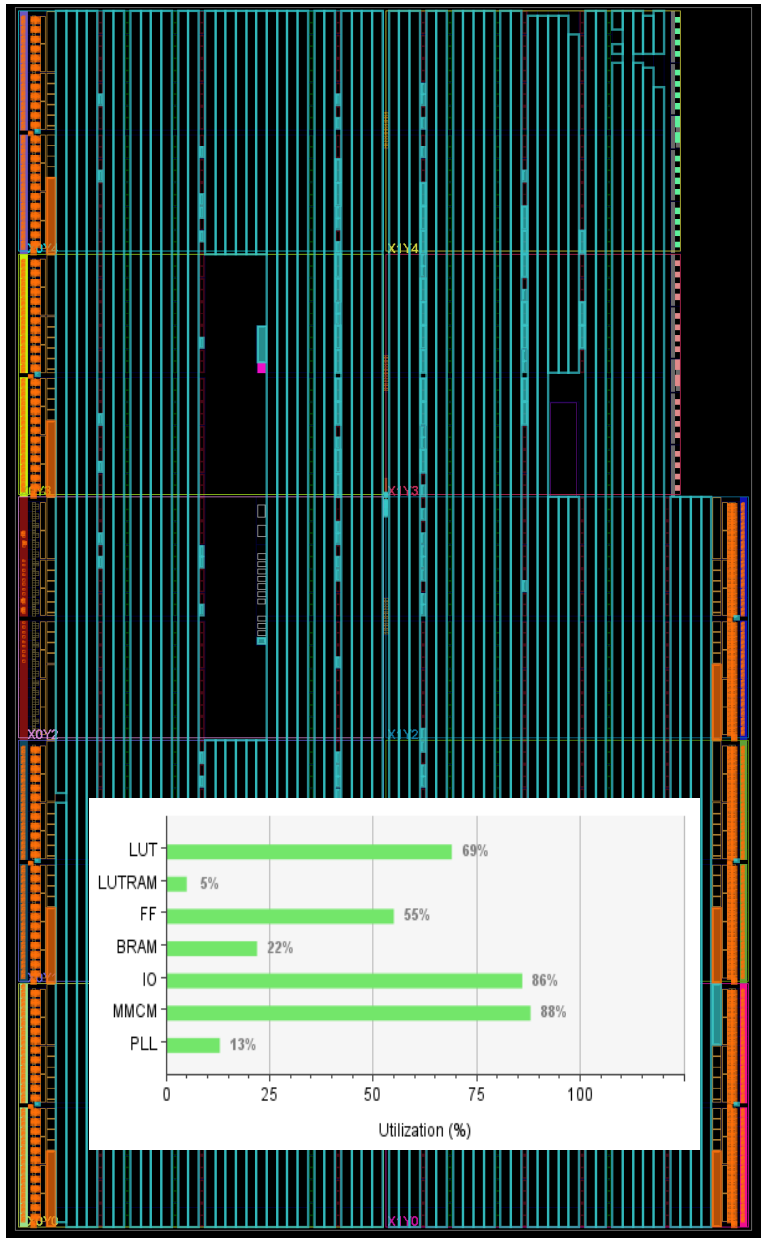
| WR tai [39 .. 33] | WR tai [32 .. 0] | WR cycles [27 .. 1] | TDC counter [3 ..0] | TDC time [4 .. 0] |
|---|---|---|---|---|
| 7 MSB WR tai (centuries) | WR tai (1 sec) | WR counter/2 (32ns) | TDC fabric counter 0 ..9 (3.2 ns) | TDC fine time (100 ps) |

# Full scale FPGA design

Master FPGA 72 ch




Slave FPGA 84 ch




- Master & Slave FPGA firmware are compiled
- Usable for PCB pinout verification

- TODO list:
- ❑ Control registers optimization
- ❑ TDC revision
- ❑ FPGA cross link implementation

# HGND readout board routing status:

May 2024:

- ✓ Front-End connectors pinout and schematic
- ✓ White Rabbit periphery schematic
- ➢ Board geometry and connector positions (May 2024)
- ❑ Power supply schematic (June-July 2024)
- ❑ TDC tests with prototype v2 (June – July 2024)
- ❑ FPGA pinout and schematic (June-July 2024)
- ❑ Board routing (August-September 2024)

Sep 2024:

- ✓ Front-End connectors pinout and schematic
- ✓ White Rabbit periphery schematic
- ✓ Board geometry and connector positions (May 2024)
- ✓ Power supply schematic (June-July 2024)
- ✓ TDC tests with prototype v2 (June – July 2024)
- ✓ FPGA pinout and schematic (June-July 2024)
- ➢ Board routing (August-November 2024)

# HGND readout & DCS readout test setup


**<u>Synthetic tests for proving the concept of developed software architecture</u>**

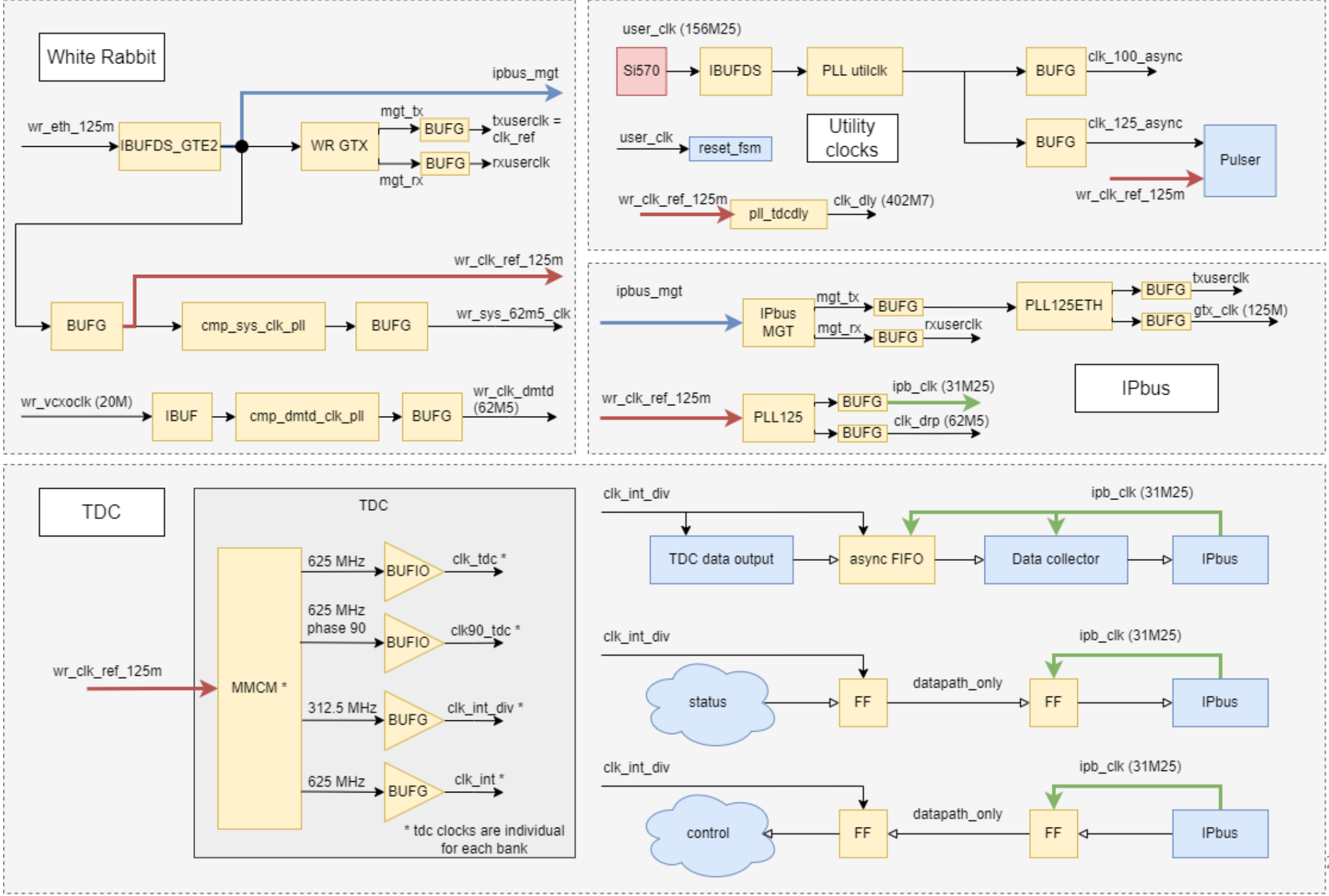
- ✓ Data & trigger FPGA emulator
- ✓ 400 Mbit/s data readout per board (100 Mbit is required)
- ✓ 800 Mbit/s readout rate with 2 boards was achieved (is the estimated rate for HGND with 8 links)
- ✓ Data sorting by trigger selection with two links readout: data rate 2.6 MHz (10 kHz/channel), trigger rate 10kHz.
- ✓ Data flow (soft emu) & DCS commands test on BM@N FLP
- ➢ Preparing software for TDC tests
- ❑ Working on GUI filling and debugging

# Conclusions

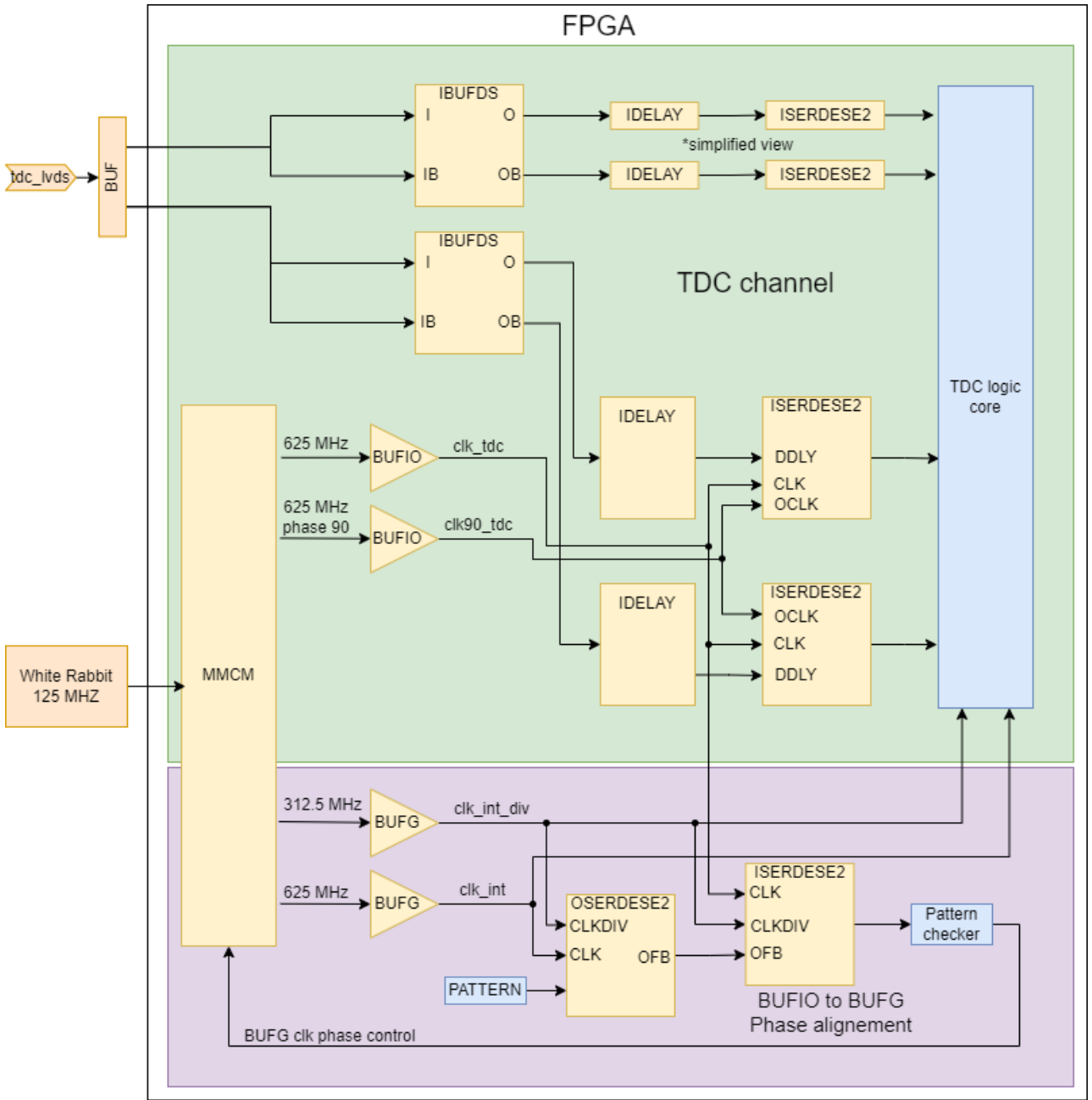
- Status of the HGND readout development:
  - ✓ Basic TDC tests was performed with 33 channels prototype
  - ✓ The White Rabbit synchronization works
  - ✓ Full scale FPGA design was prepared (dummy)
  - ✓ Readout & DCS software is ready (basic)
  - ➢ Continue tests with prototype v2 (cosmic and beam tests with matrix, TDC development)
  - ➢ Working on the FPGA firmware: TDC revision, cross FPGA link routine
  - ➢ Working on the design of the full scale readout board: board routing
  - ➢ Software debugging and updates

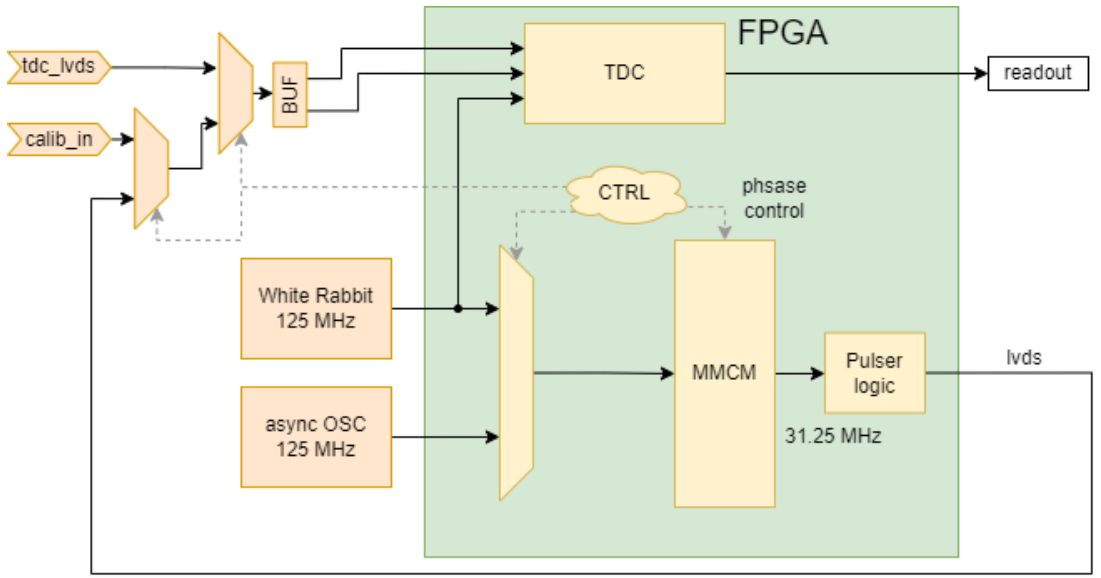
Thank you for your attention!

# BACKUP

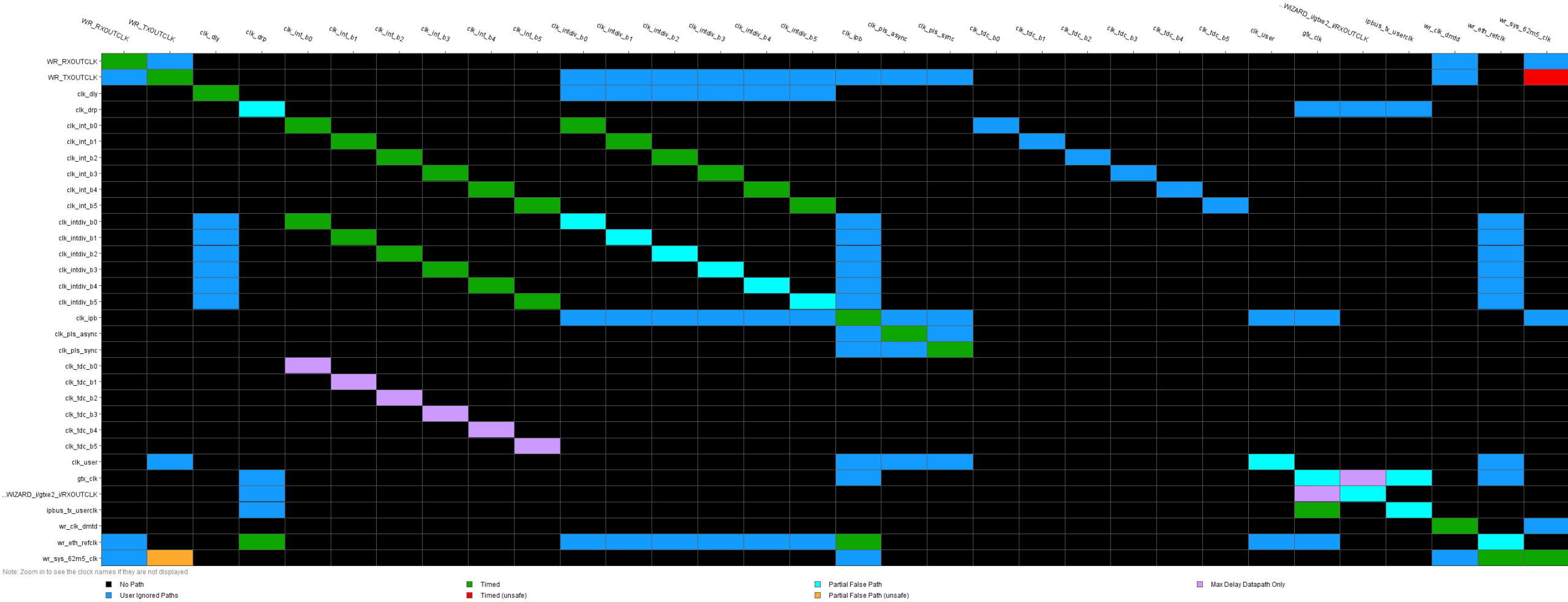

White Rabbit
ipbus_mgt
wr_eth_125m
IBUFDS_GTE2
WR GTX
mgt_tx
BUFG
txuserclk = clk_ref
BUFG
rxuserclk
mgt_rx
wr_clk_ref_125m
BUFG
cmp_sys_clk_pll
BUFG
wr_sys_62m5_clk
wr_vcxoclk (20M)
IBUF
cmp_dmtd_clk_pll
BUFG
wr_clk_dmtd (62M5)
user_clk (156M25)
Si570
IBUFDS
PLL utilclk
BUFG
clk_100_async
user_clk
reset_fsm
Utility clocks
BUFG
clk_125_async
Pulser
wr_clk_ref_125m
wr_clk_ref_125m
pll_tdcdly
clk_dly (402M7)
ipbus_mgt
IPbus MGT
mgt_tx
BUFG
mgt_rx
BUFG
rxuserclk
PLL125ETH
BUFG
txuserclk
BUFG
gtx_clk (125M)
IPbus
wr_clk_ref_125m
PLL125
BUFG
ipb_clk (31M25)
BUFG
clk_drp (62M5)
TDC
TDC
wr_clk_ref_125m
MMCM *
625 MHz
BUFIO
clk_tdc *
625 MHz phase 90
BUFIO
clk90_tdc *
312.5 MHz
BUFG
clk_int_div *
625 MHz
BUFG
clk_int *
* tdc clocks are individual for each bank
clk_int_div
ipb_clk (31M25)
TDC data output
async FIFO
Data collector
IPbus
clk_int_div
ipb_clk (31M25)
status
FF
datapath_only
FF
IPbus
clk_int_div
ipb_clk (31M25)
control
FF
datapath_only
FF
IPbus

# FPGA TDC and calibration pulser clocks layout

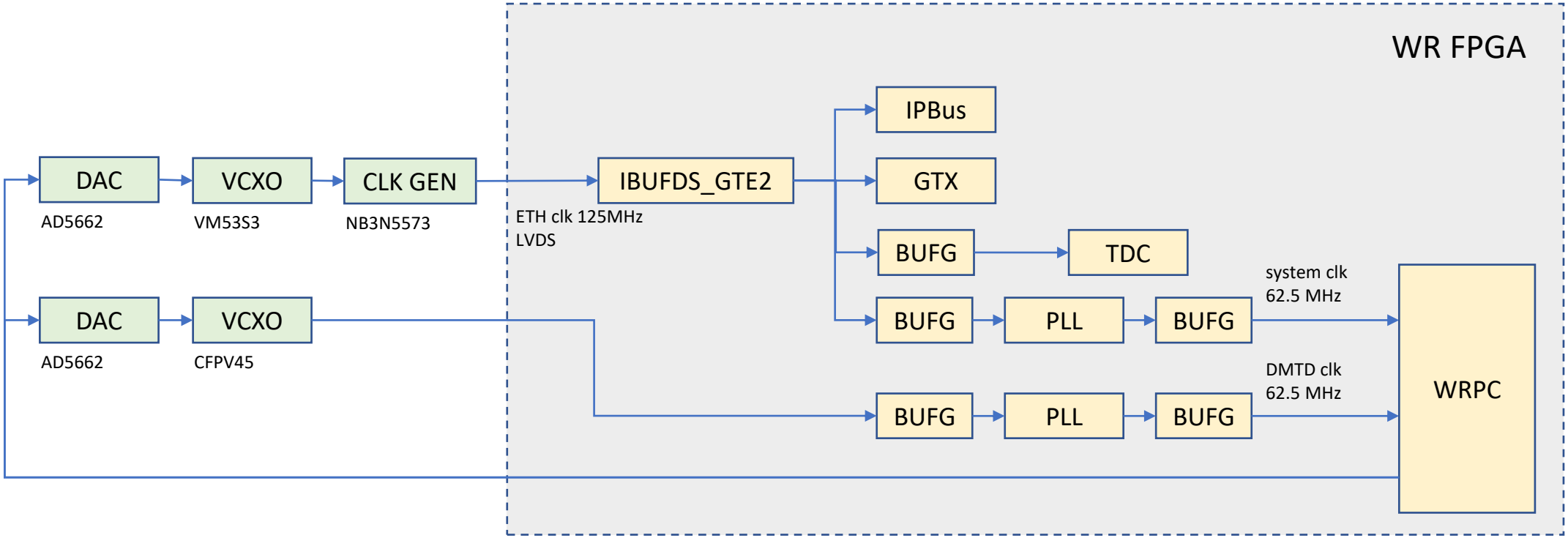


# Master FPGA clock interaction



# The White Rabbit implementation

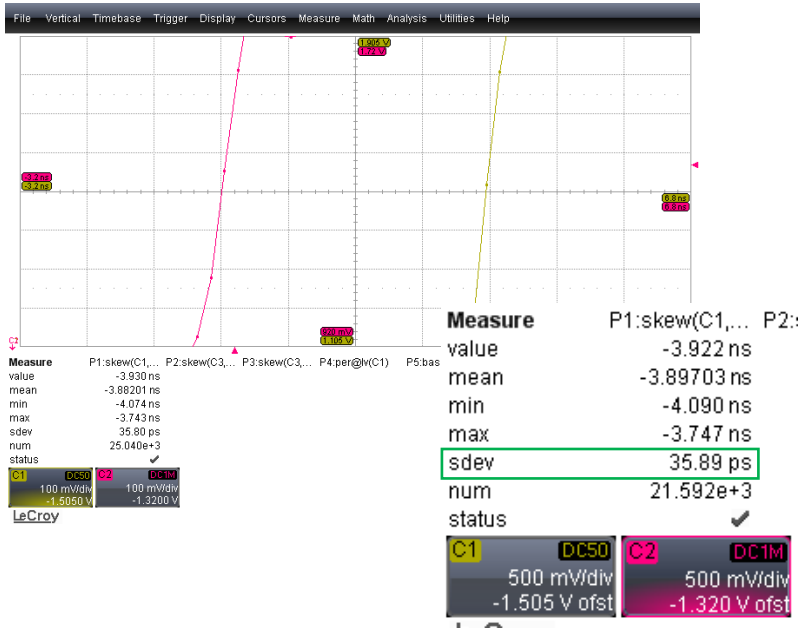


# HGND readout proto_v2 WR tests at JINR (16 – 17 of April)

HGND readout proto_v2 connected to WR master at JINR

WR TX clocks 62.5 MHz:
orange – WR master; pink – WR slave (hgnd)


Clock sync jitter ~35ps (measure on FPGA output buffers)

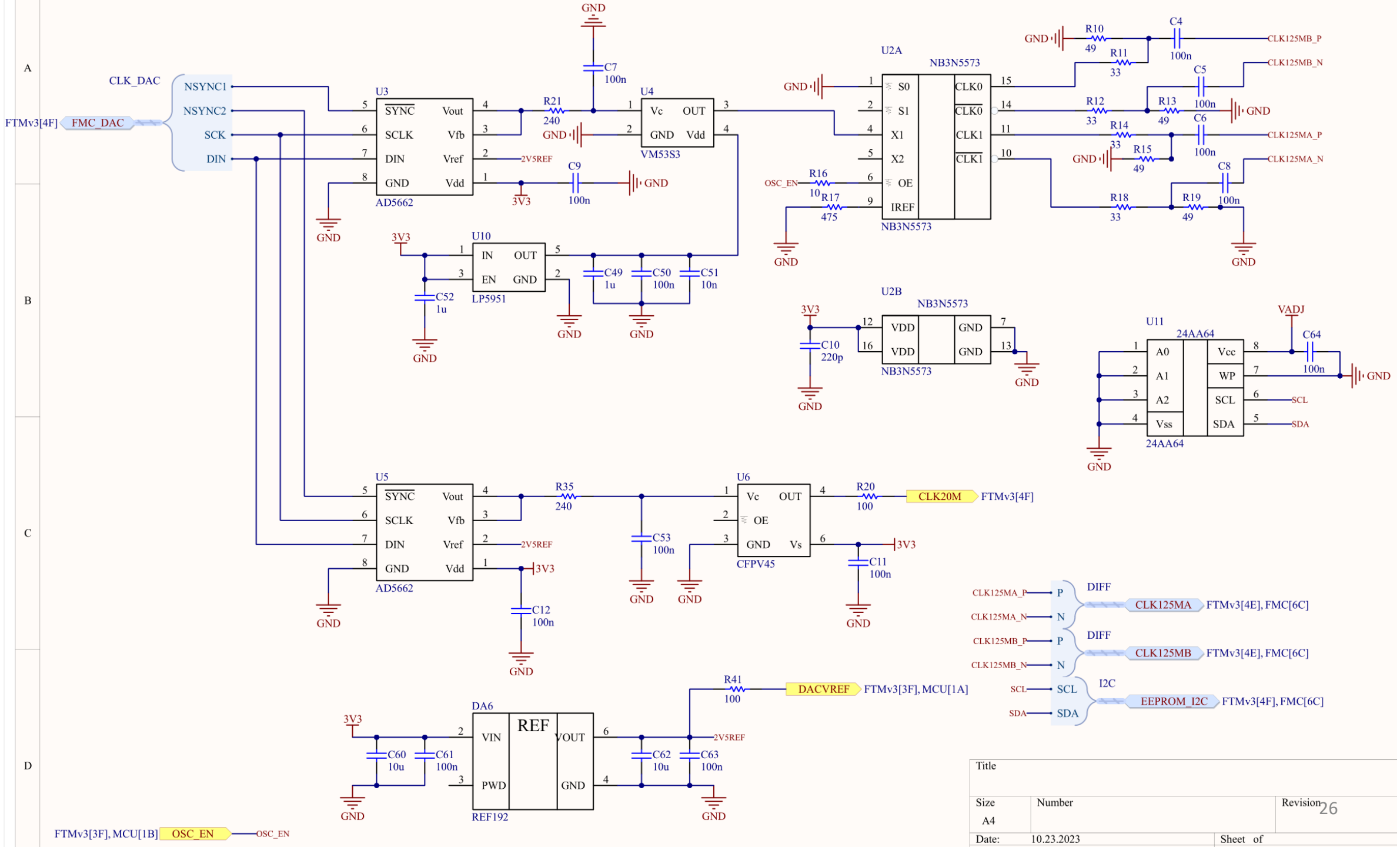


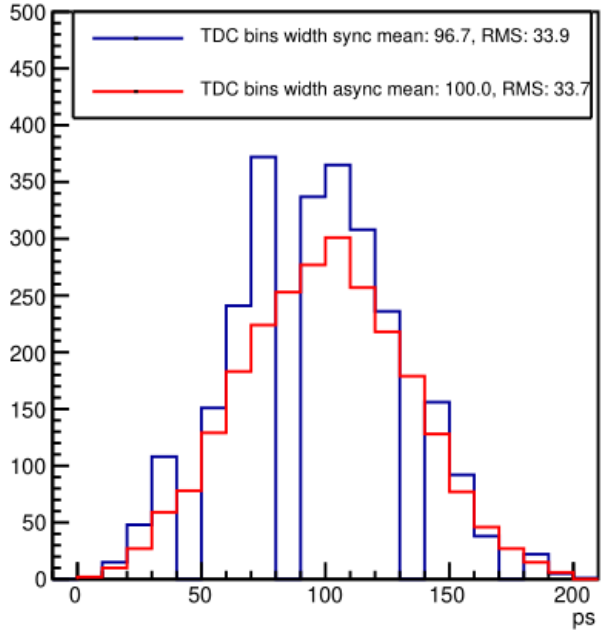

WR tests at JINR conclusions:

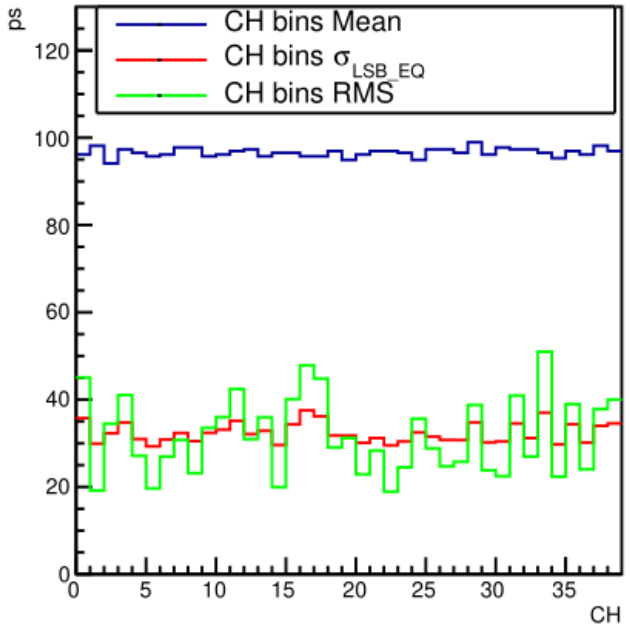
- WR synchronization works
- Tx timestamp error (reason found: generic fifo synthesis bug)
- EEPROM not works (fixed – I2C wires swapped)

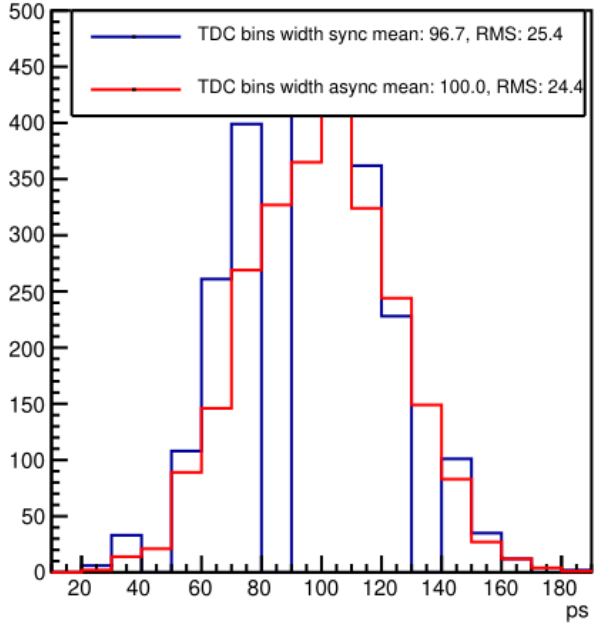
Next steps June - July:

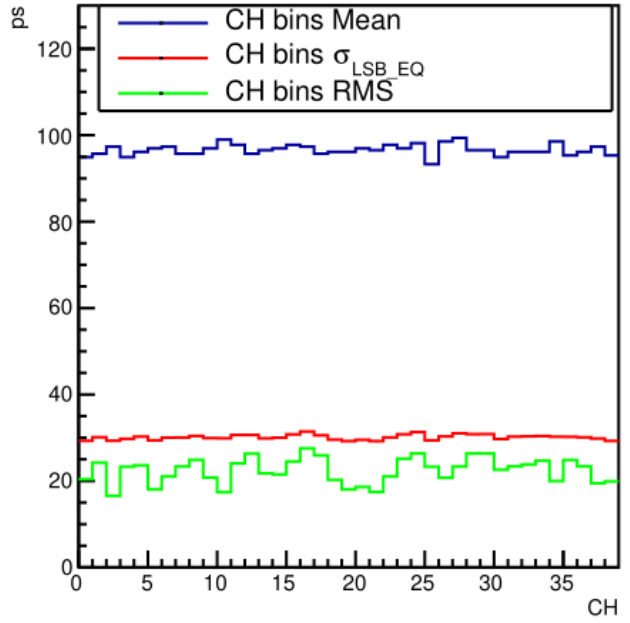
- WR error fix +  project modification (mmcm replacement with pll)
- Combining WR + Ipbus + TDC FPGA modules
- 33 channels TDC tests
- Next round of WR tests at JINR

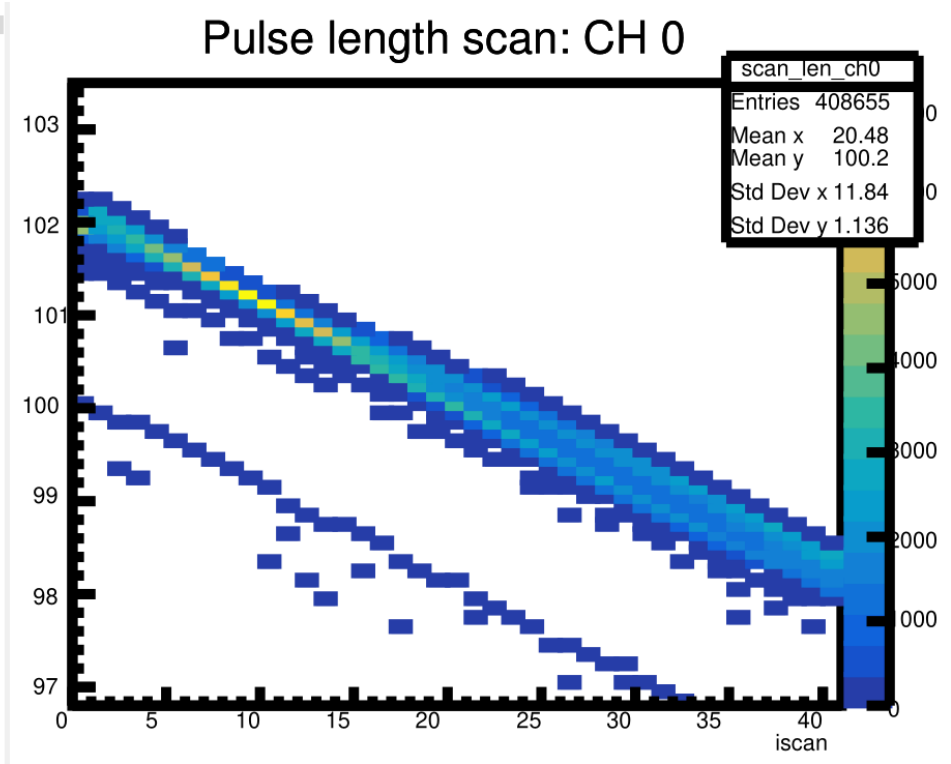

CLK_DAC
NSYNC1
NSYNC2
SCK
DIN
FTMv3[4F]
FMC_DAC
U3
AD5662
SYNC
SCLK
DIN
GND
Vout
Vfb
Vref
Vdd
R21
240
2V5REF
3V3
C7
100n
C9
100n
U4
VM53S3
Vc
GND
OUT
Vdd
U10
LP5951
IN
EN
OUT
GND
C52
1u
C49
1u
C50
100n
C51
10n
U2A
NB3N5573
S0
S1
X1
X2
OE
IREF
CLK0
CLK0
CLK1
CLK1
OSC_EN
R16
10
R17
475
R10
49
R11
33
R12
33
R13
49
R14
33
R15
49
R18
33
R19
49
C4
100n
C5
100n
C6
100n
C8
100n
CLK125MB_P
CLK125MB_N
CLK125MA_P
CLK125MA_N
U2B
NB3N5573
VDD
VDD
GND
GND
C10
220p
U11
24AA64
A0
A1
A2
Vss
Vcc
WP
SCL
SDA
C64
100n
VADJ
SCL
SDA
U5
AD5662
R35
240
U6
CFPV45
Vc
OE
GND
OUT
Vs
R20
100
CLK20M
FTMv3[4F]
C53
100n
C11
100n
C12
100n
DA6
REF192
REF
VIN
PWD
VOUT
GND
C60
10u
C61
100n
C62
10u
C63
100n
R41
100
DACVREF
FTMv3[3F], MCU[1A]
CLK125MA_P
CLK125MA_N
CLK125MB_P
CLK125MB_N
SCL
SDA
P
N
DIFF
I2C
CLK125MA
CLK125MB
EEPROM_I2C
FTMv3[4E], FMC[6C]
FTMv3[4F], FMC[6C]
FTMv3[3F], MCU[1B]
OSC_EN
Title
Size
A4
Number
Revision
Date:
10.23.2023
Sheet of


TDC bins width sync mean: 96.7, RMS: 33.9
TDC bins width async mean: 100.0, RMS: 33.7
ps

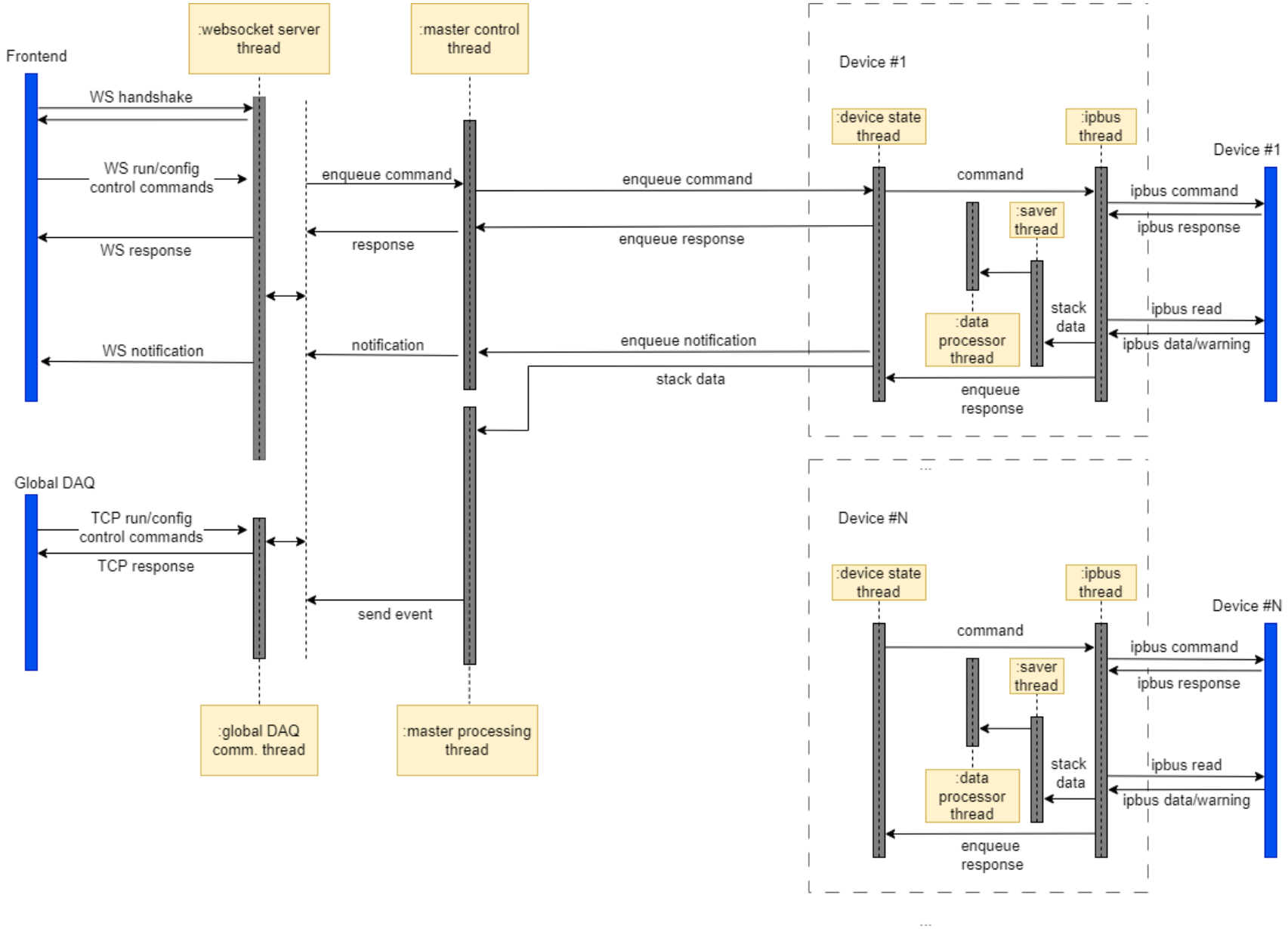

CH bins Mean
CH bins σLSB_EQ
CH bins RMS
ps
CH


TDC bins width sync mean: 96.7, RMS: 25.4
TDC bins width async mean: 100.0, RMS: 24.4
ps

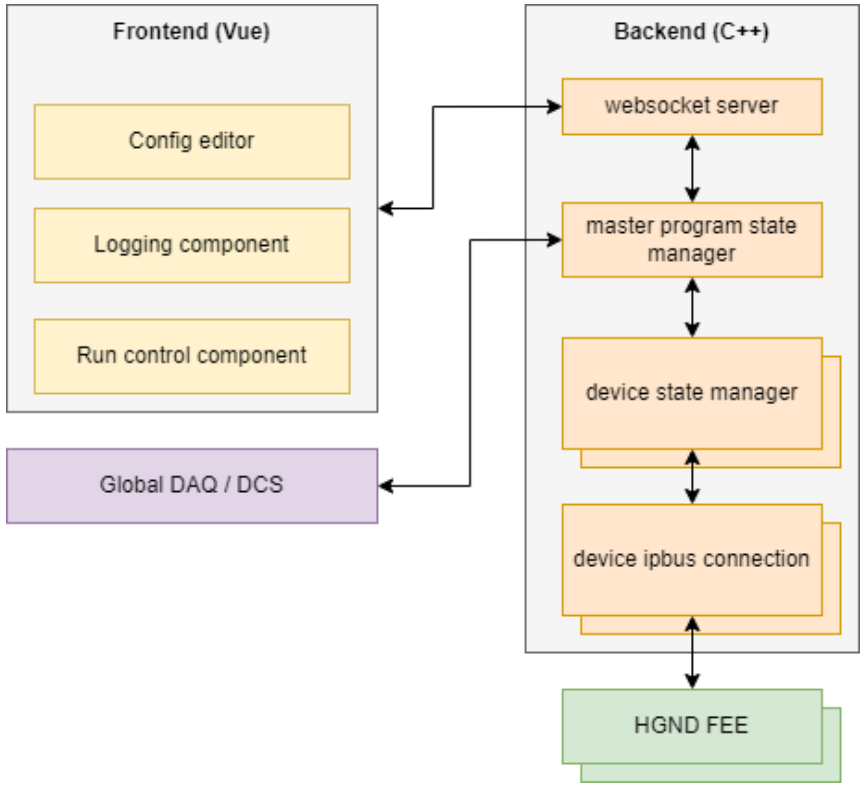

CH bins Mean
CH bins σLSB_EQ
CH bins RMS
ps
CH

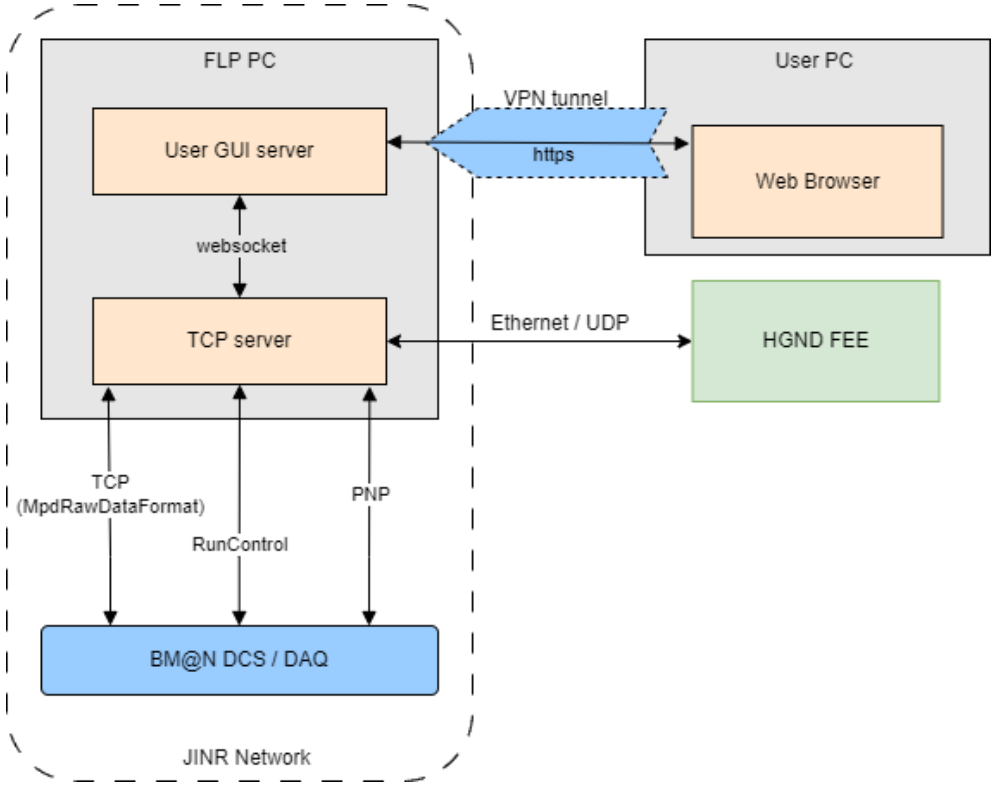

Pulse length scan: CH 0
scan_len_ch0
Entries 408655
Mean x 20.48
Mean y 100.2
Std Dev x 11.84
Std Dev y 1.136
iscan

# The threading scheme of a C++ backend



# The topology of the DCS & readout software




- The detector control system consists of 2 independent modules: frontend and backend
- The backend is a C++ server running on a DCS computer in the same network with detectors.
- The server part provides write/read operations to the detectors via an IPBus connection to the control and data acquisition board.

- Frontend is the user interface for this backend. It can be implemented in various ways; the key parameters are reactivity and an ability to establish a connection via websocket with the server part.
- The current version is the Vue web application (open-source JavaScript framework).
- The interface can be run either on the DCS computer or on the operator's computer, provided that the port is available.