POKER

POsitron resonant annihilation into dar**K** matt**ER**

Time synchronization of front-end digitizers in the POKER/NA64 experiment

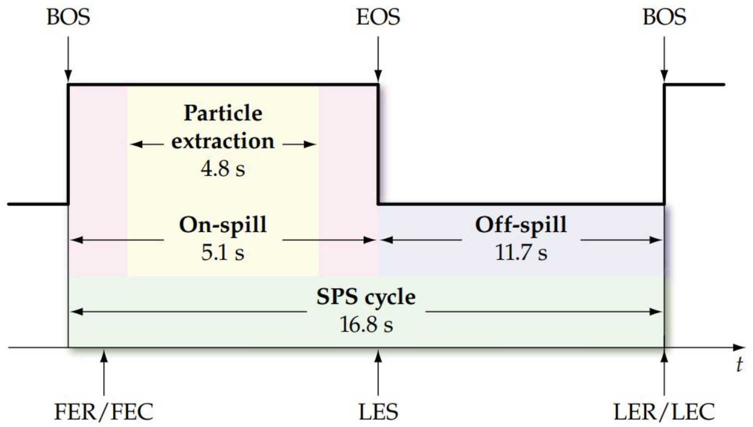
ICSC - Spoke 2 - WP2 meeting

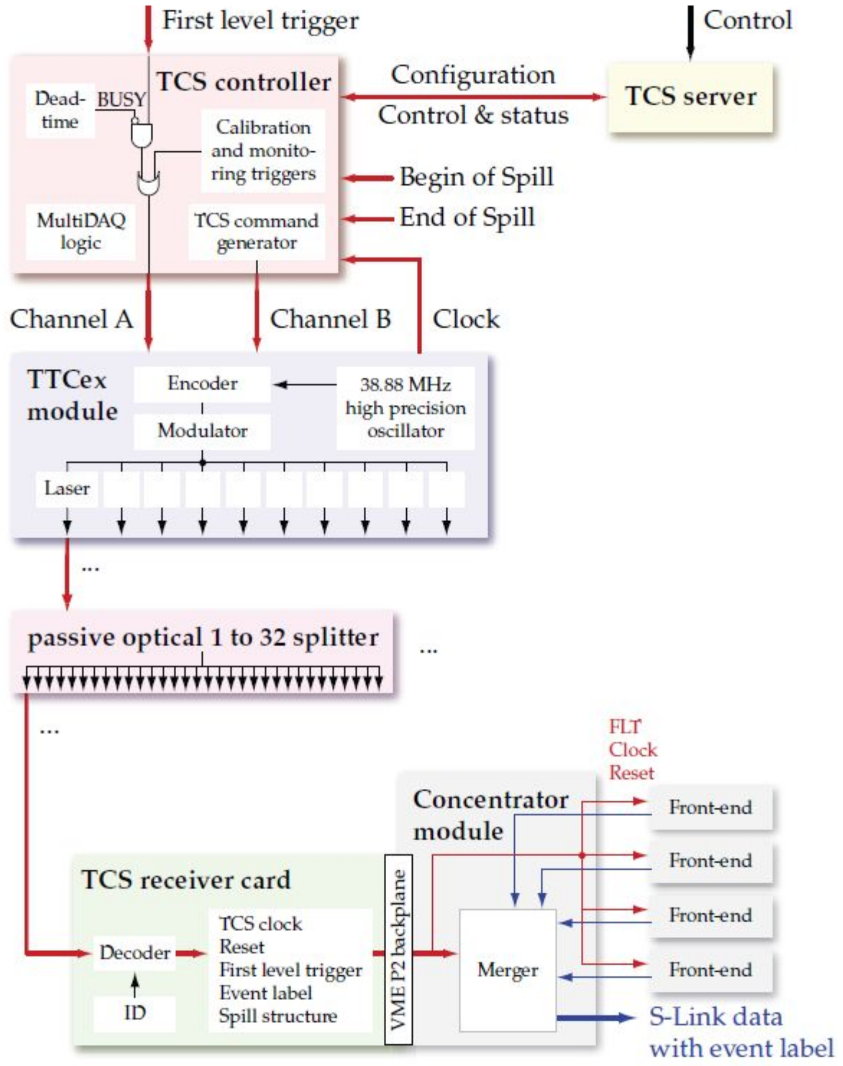
22/10/2024

Andrei Antonov - INFN Genova

DAQ synchronization in NA64

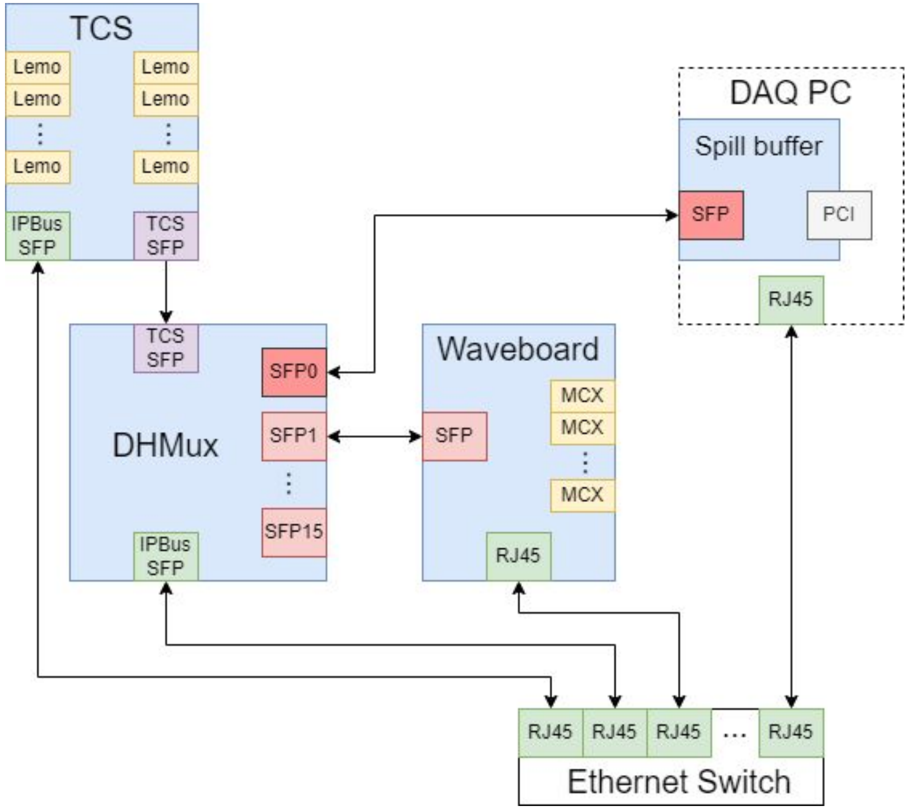
- The concept taken from the COMPASS experiment
- TCS – Trigger and Control System
- TCS provides DAQ with:
    - Reference clock (recovered by receiver)
    - Encoded trigger and control data
    - Spill structure



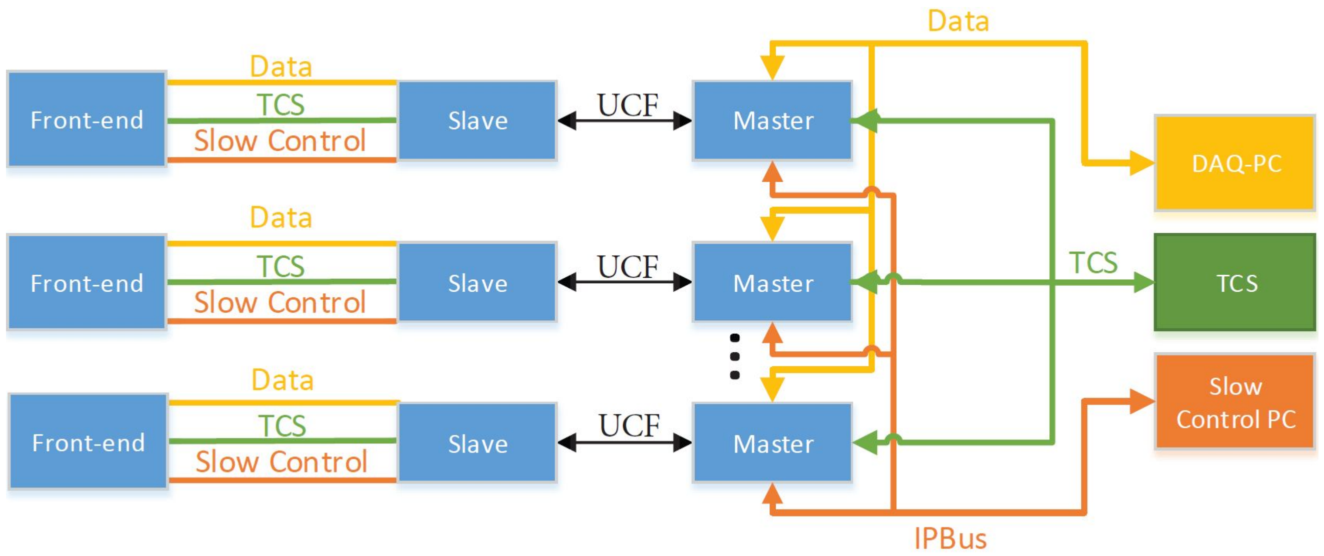


New NA64 DAQ structure

- Old ECAL front-end is not fast enough to fulfill the experiment requirements.
- Waveboard is the new front-end digitizer
- Has to be integrated into the existing DAQ system
- DHMux – DAQ-core, translates TCS-flow and aggregates data flows
- Uses UCF-protocol for data, trigger/synchronization and slow control



Unified Communication Framework protocol – all in one



- Originally developed for COMPASS experiment at TUM
- Works over optical fiber, uses 8b/10b encoding
- Used bit rate: 3.1104 Gb/s = 155.52 MHz * 20 bits (2-byte word 8b/10b encoded)
- Supports up to 64-streams in one physical link: one mandatory – for TCS, other optional – for data (including slow control)

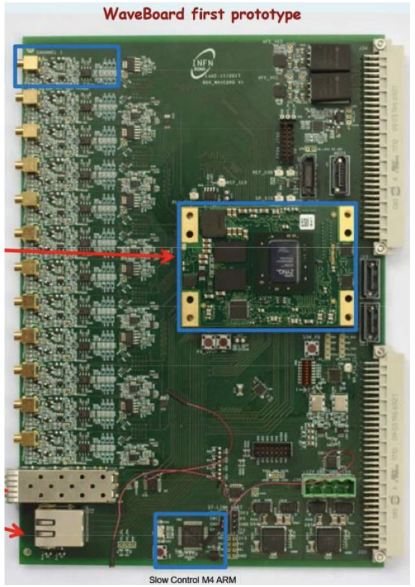
See also: https://indico.cern.ch/event/783347/contributions/3301582/attachments/1793625/2922863/DAQFEET2019_UCF.pdf

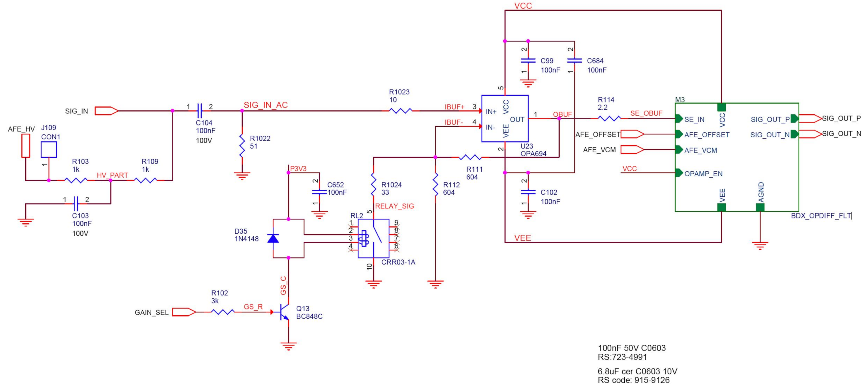
DHMux – the backbone of NA64 DAQ

- Aggregates data flows from up to 15 slaves (front-end boards) – builds event based on received data
- Uses UCF or S-Link interface to communicate with slaves
- Uses S-Link interface on a dedicated port 0 to communicate with Spill Buffer (or other DHMux)
- Gets triggers and commands from TCS, encapsulates them into UCF link and sends to UCF-compatible slaves.
- Uses the recovered TCS link clock 155.52 MHz as a reference clock for the UCF interfaces
- Uses IPBus as a slow control interface for itself and for the slaves
- Works as an Ethernet switch for IPBus-compatible slaves

Waveboard - main properties

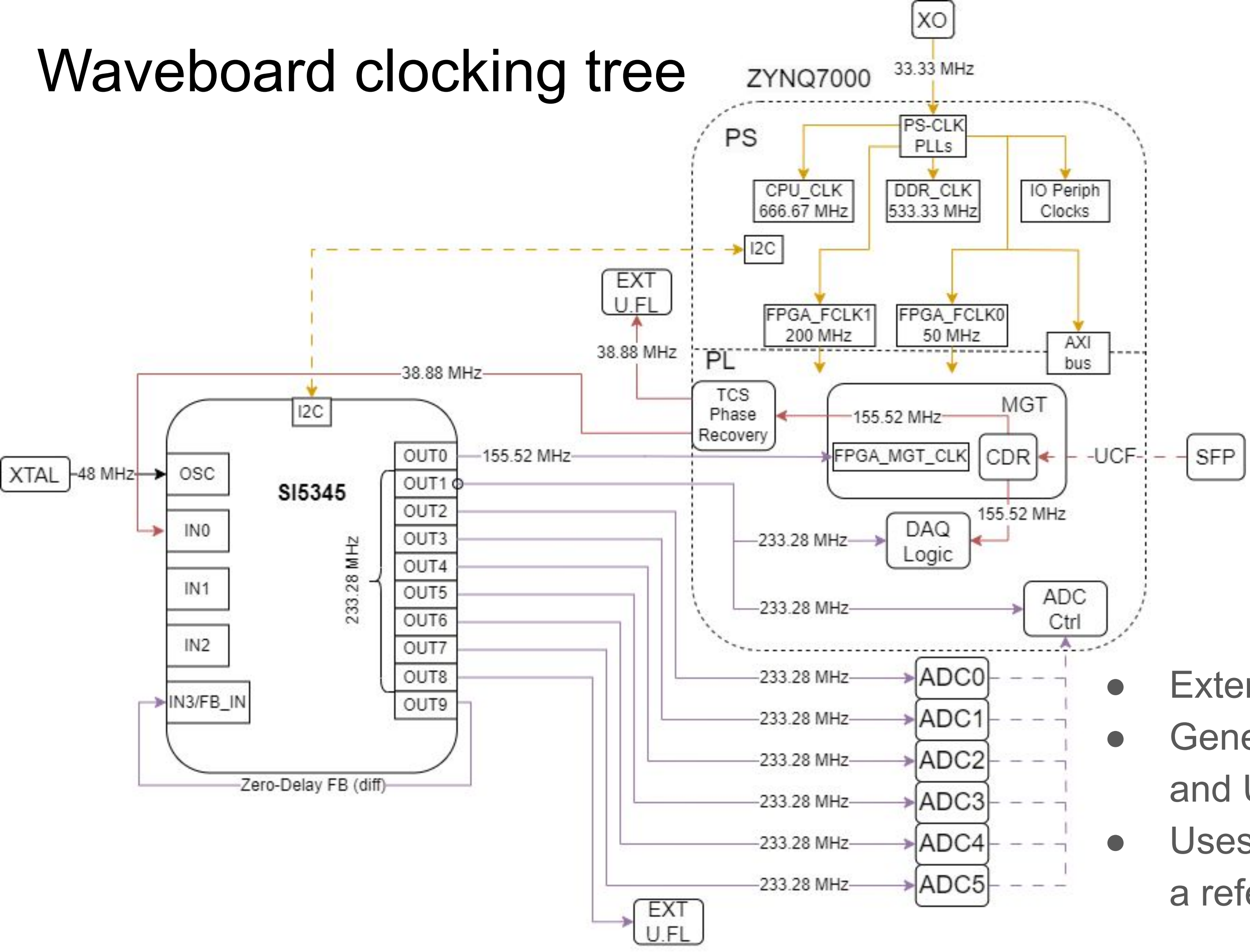
- 12-channels per board
- Front-end currently optimized for SiPM signals
  - OPA694-based amplifier with programmable gain (LG: 2 / HG: 20)
  - Single-ended to differential driver (based on LMH6552)
  - Anti-aliasing filter
- Each channel includes a programmable positive HV source
  - HV-source: 25...75 V, onboard/external HV reference. *HV generator can be switched on/off independently for each channel*
- ADC: 14 bits @ 65-250 MHz
  - Currently 250 MHz version is implemented
- Zynq FPGA SOMs from Trenz Electronics
  - Xilinx Kintex-7 + dual-core ARM
- Multiple choice of time synchronization interfaces
  - On-board PLL / jitter cleaner for clock distribution
  - Currently clock recovered from the SFP is used as a reference
- Form-factor: 6U VME module
  - VME only provides +5V, +12V power
  - Power can also be provided by an external supply





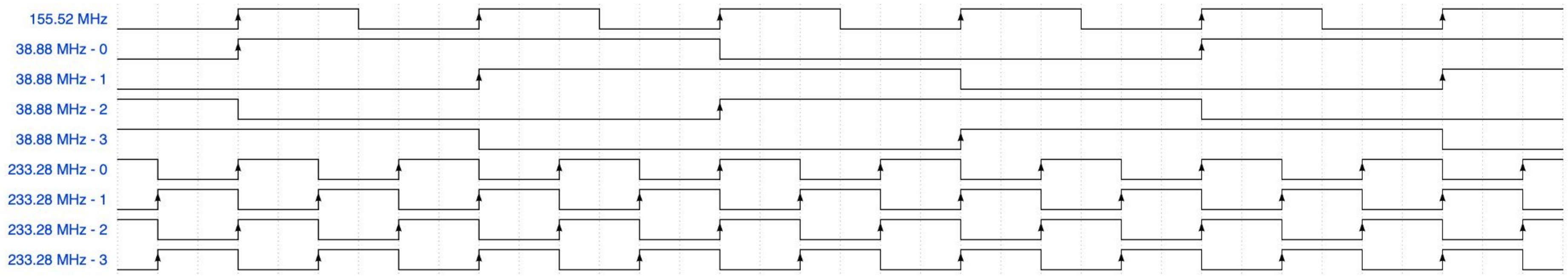
See also: https://agenda.infn.it/event/18179/contributions/89837/

Waveboard clocking tree



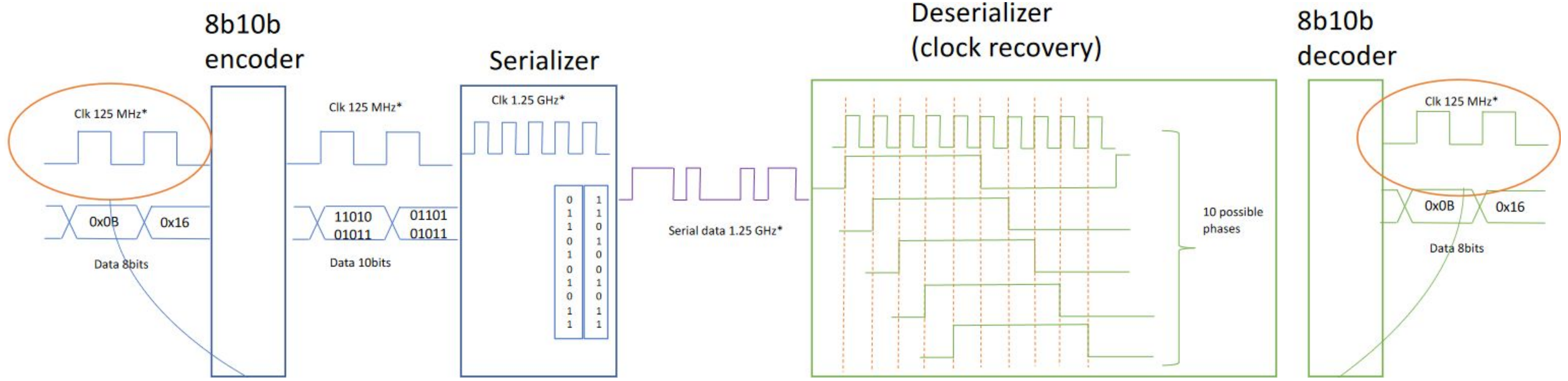
- External PLL SI5345 is used
- Generates clocks for FADCs and UCF GTX
- Uses recovered TCS clock as a reference input

Clock synchronization



- TCS-link clock 155.52 MHz is recovered from the UCF-link by GTX CDR PLL
- The recovered clock is used as an FPGA fabric clock for UCF data path
- It is also divided by 4 with phase reconstruction and goes to the SI5345 PLL reference input
- SI5345 generates FADC clock for the FADC themselves and for the ADC interface controller in the FPGA
- FADC clock frequency is 233.28 MHz = 38.88 MHz * 6.

Clock synchronization – link clock recovery

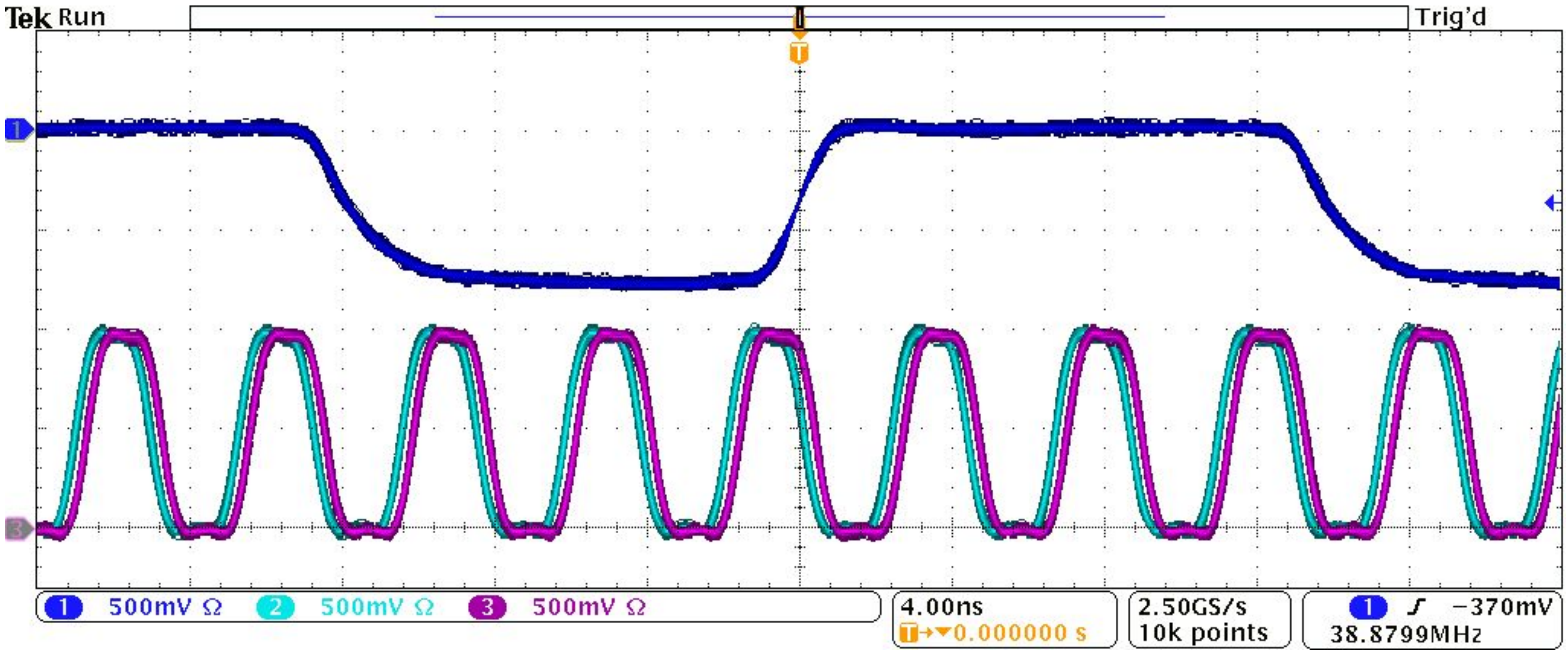


- By default in GTX clock recovery is done without care about phase, only frequency
- Not applicable in our case
- To achieve maximum clock phase recovery resolution we do manual word alignment is implemented using RXSLIDE directly in GTX PMA
- Data alignment is done by shifting slow clock and not the data word
- By shifting we reconstruct the slow clock with precision of one bit clock period
- The maximum possible resolution is: 1/3.1104 Gb/s = 320 ps

Clock synchronization – TCS phase recovery

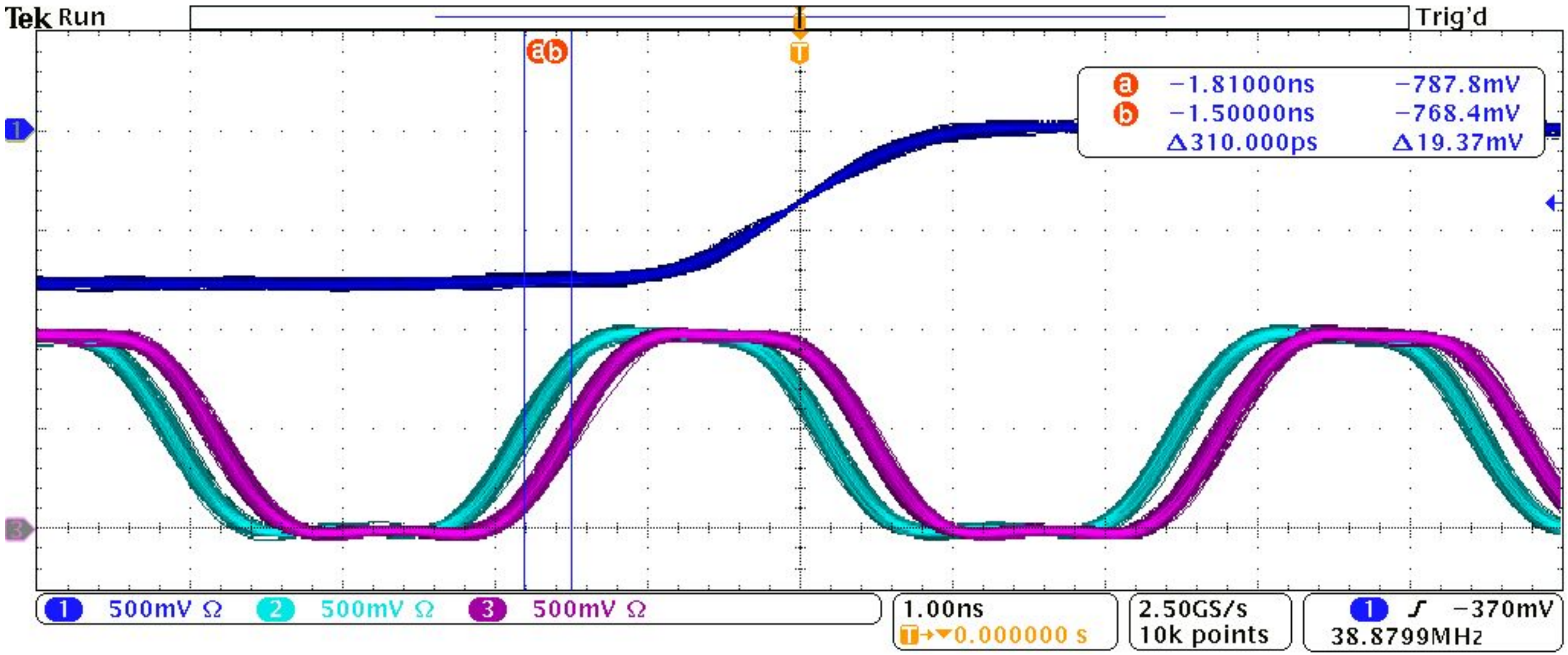
- The recovered 155.52 MHz UCF clock is divided by 4 using Xilinx BUFR primitive to generate 38.88 MHz TCS clock
- TCS periodically broadcasts SYNC command
- DHMux encapsulates SYNC into a UCF packet
- Transmission and reception of this packet has deterministic latency
- This allows slaves to use SYNC to recover TCS clock phase
- The recreated from the UCF packet SYNC signal is used as an asynchronous reset (CLR input) of the BUFR primitive
- Every time we receive SYNC packet we readjust TCS clock phase
- TCS also broadcasts RESET command to reset front-ends’ trigger counters and timers to achieve time synchronization within DAQ

Result analysis – Waveboard FADC clock



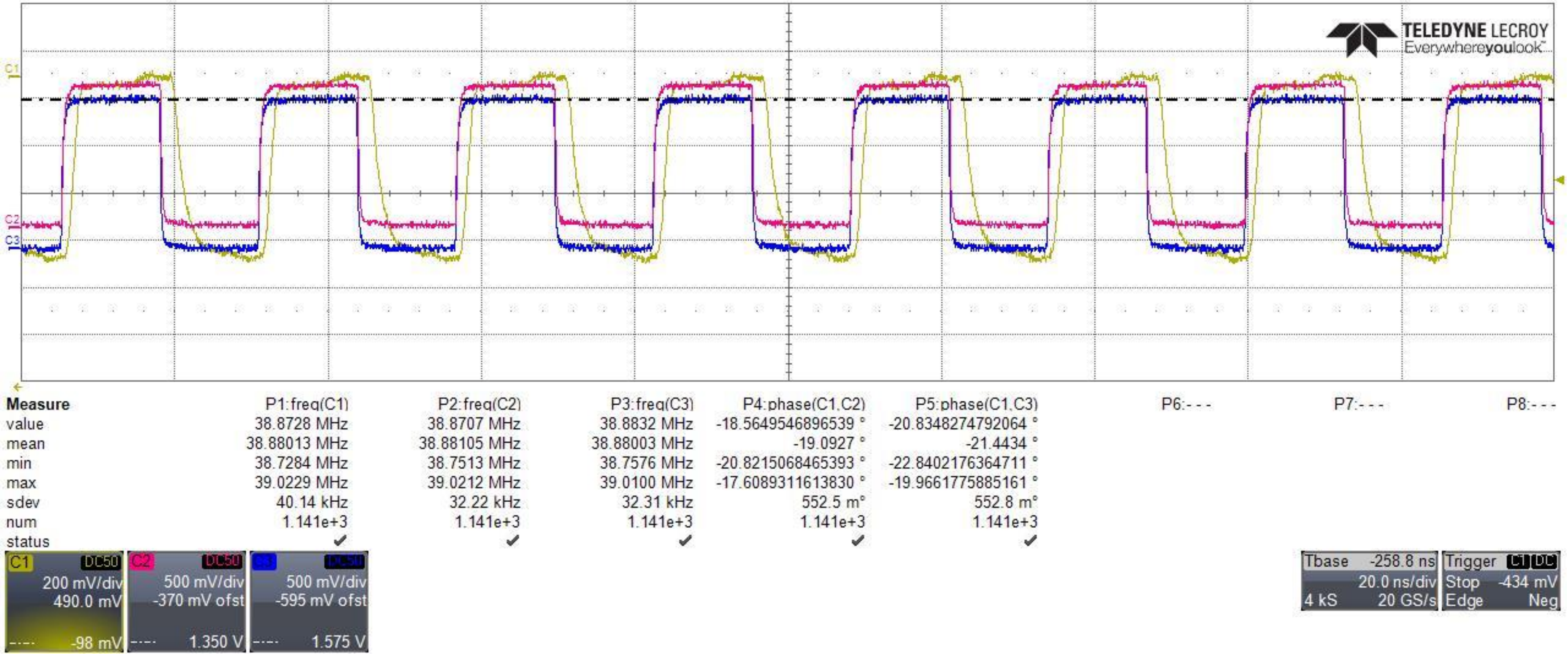
- Blue (1) channel – the TCS output clock, observed directly on the TCS module
- Light blue (2) and purple (3) channels – two Waveboards' FADC clocks

Result analysis – constant FADC clock phase difference



- Phase difference stays constant disregarding Waveboard/DHMux resets, cable reattachments – it was the main goal of this work

Result analysis – recovered TCS clock



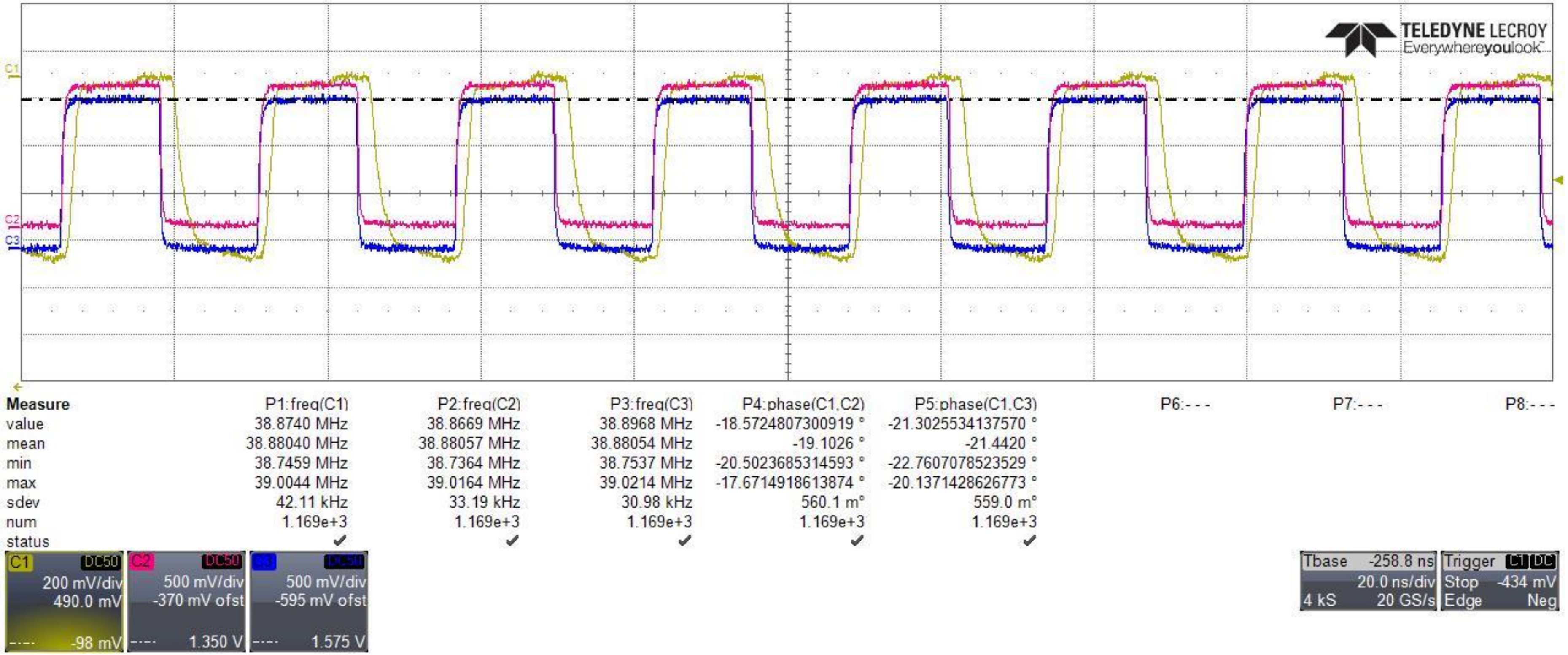
Measure	P1:freq(C1)	P2:freq(C2)	P3:freq(C3)	P4:phase(C1,C2)	P5:phase(C1,C3)	P6:- - -	P7:- - -	P8:- - -
value	38.8728 MHz	38.8707 MHz	38.8832 MHz	-18.5649546896539 °	-20.8348274792064 °			
mean	38.88013 MHz	38.88105 MHz	38.88003 MHz	-19.0927 °	-21.4434 °			
min	38.7284 MHz	38.7513 MHz	38.7576 MHz	-20.8215068465393 °	-22.8402176364711 °			
max	39.0229 MHz	39.0212 MHz	39.0100 MHz	-17.6089311613830 °	-19.9661775885161 °			
sdev	40.14 kHz	32.22 kHz	32.31 kHz	552.5 m°	552.8 m°			
num	1.141e+3	1.141e+3	1.141e+3	1.141e+3	1.141e+3			
status	✓	✓	✓	✓	✓			

```
C1        DC50   C2        DC50   C3        DC50
200 mV/div       500 mV/div       500 mV/div
490.0 mV         -370 mV ofst     -595 mV ofst
----    -98 mV   ----    1.350 V  ----    1.575 V

Tbase   -258.8 ns   Trigger   C1 DC
        20.0 ns/div Stop    -434 mV
4 kS    20 GS/s     Edge        Neg
```

- Yellow (1) channel – the TCS output clock, observed directly on the TCS module
- Blue (2) and purple (3) channels – two Waveboards' recovered TCS clock

Result analysis – recovered TCS clock after DHMux reset



```
Measure   P1:freq(C1)    P2:freq(C2)    P3:freq(C3)    P4:phase(C1,C2)        P5:phase(C1,C3)        P6:---  P7:---  P8:---
value     38.8740 MHz    38.8669 MHz    38.8968 MHz    -18.5724807300919 °    -21.3025534137570 °
mean      38.88040 MHz   38.88057 MHz   38.88054 MHz   -19.1026 °             -21.4420 °
min       38.7459 MHz    38.7364 MHz    38.7537 MHz    -20.5023685314593 °    -22.7607078523529 °
max       39.0044 MHz    39.0164 MHz    39.0214 MHz    -17.6714918613874 °    -20.1371428626773 °
sdev      42.11 kHz      33.19 kHz      30.98 kHz      560.1 m°               559.0 m°
num       1.169e+3       1.169e+3       1.169e+3       1.169e+3               1.169e+3
status    ✓              ✓              ✓              ✓                      ✓

C1  DC50         C2  DC50         C3  DC50                 Tbase  -258.8 ns   Trigger  C1 DC
  200 mV/div       500 mV/div       500 mV/div                    20.0 ns/div  Stop   -434 mV
  490.0 mV         -370 mV ofst     -595 mV ofst           4 kS   20 GS/s      Edge       Neg
  ----  -98 mV     ----  1.350 V    ----  1.575 V
```

- After resetting all the TCS clock recovery circuits, the DHMux-Waveboard couple restores it with the same frequency and phase

Conclusion

- The work results were tested in the lab (INFN GE) on a reduced but fully functional NA64 DAQ set-up
- Successfully used on a test beam (July 2024)
- For the next NA64 session (April 2025) we are going to use Waveboard as a main digitizer of the experiment