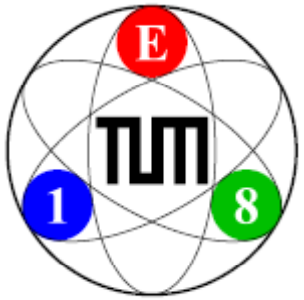


SODA: Synchronization Of Data Acquisition

I.Konorov

- Requirements
- Architecture
- System components
- Performance
- Conclusions and outlook

Requirements & Functions

Requirements

- Provision of absolute time at Front-end electronics
- Precision 20 ps for ToF and 50 ps for all other detectors
- Synchronization with operation of HESR
- Synchronization with Target

Implementation :

Time reference

- single clock to all front-ends via optical network – 100/155.52 MHz

Absolute Time

- RESET SIGNAL
  - sets local time counters to ZERO
  - issued at start of run

Start/End of data block

- Heart Bit – Start of Block/End Of Block signals

HESR beam structure : 2us beam & 400 ns no beam

Enable/Disable Data

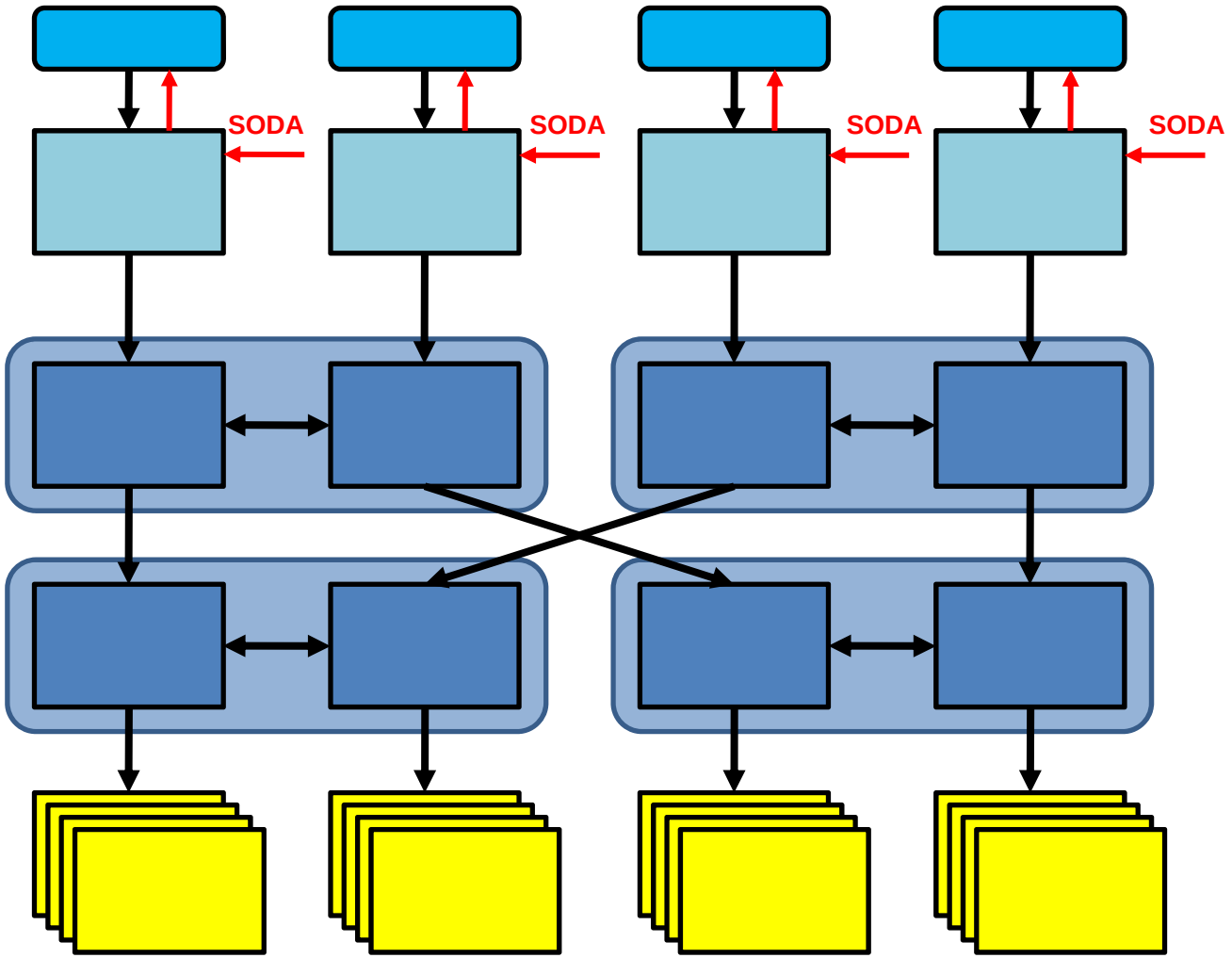
- data throttling

Extra functions :

Data flow control

calibration, tests …

DAQ Architecture



Detector Frontends

Data Multiplexer

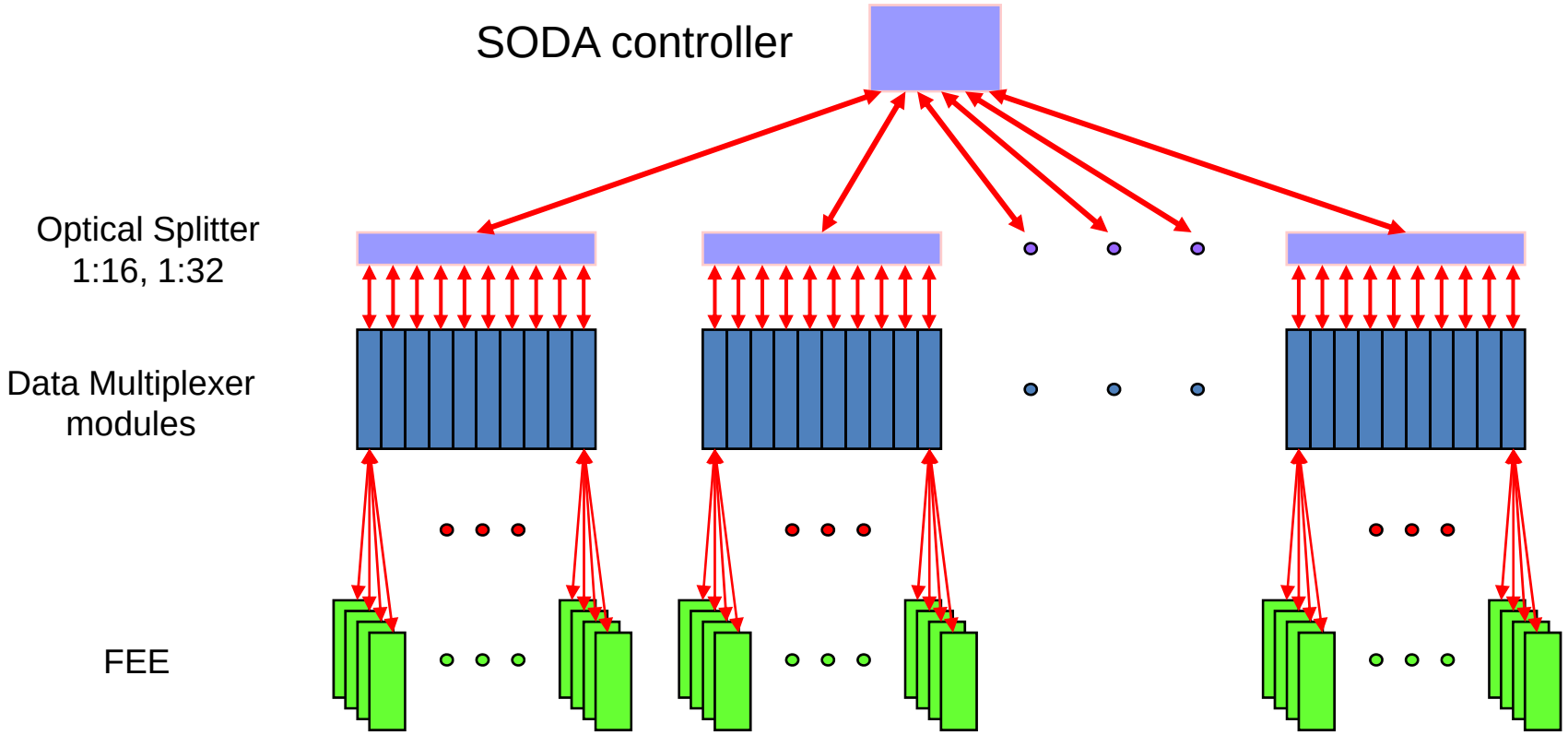
ATCA based Sub "Event" Builder

ATCA based "Event" Builder

Compute Nodes

SODA architecture

SODA
- High speed optical serial link
- Single source multiple destinations

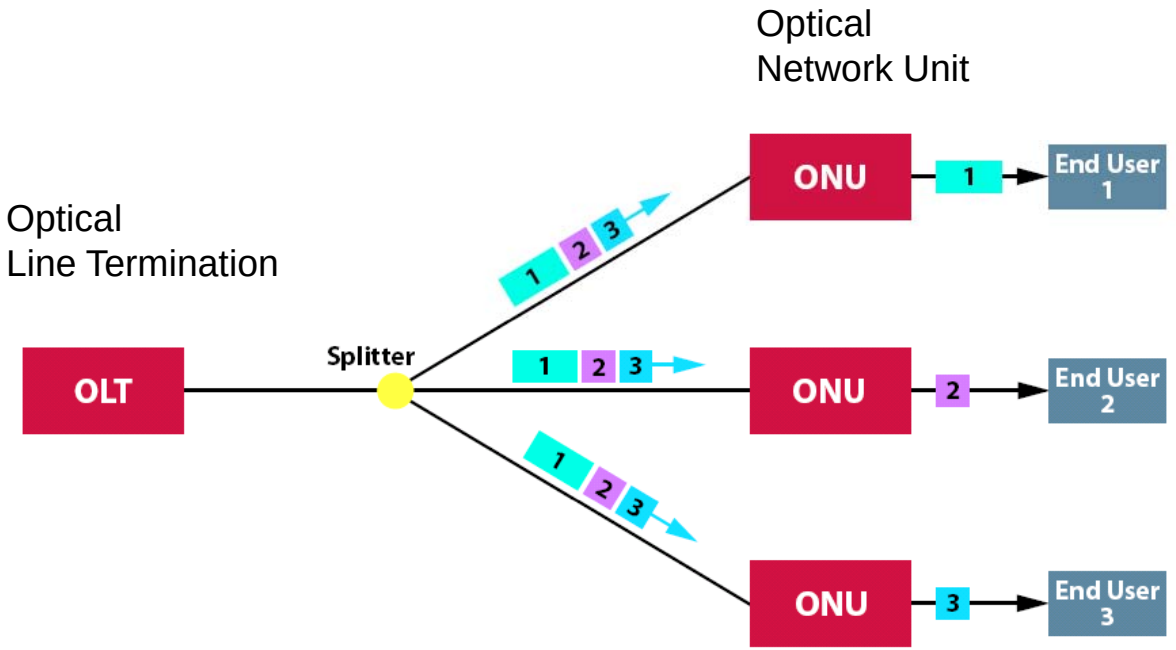


PON technology

PON – Passive Optical Network

- Objectives:
  - Video on Demand
  - Telephone over IP
  - High BW Network

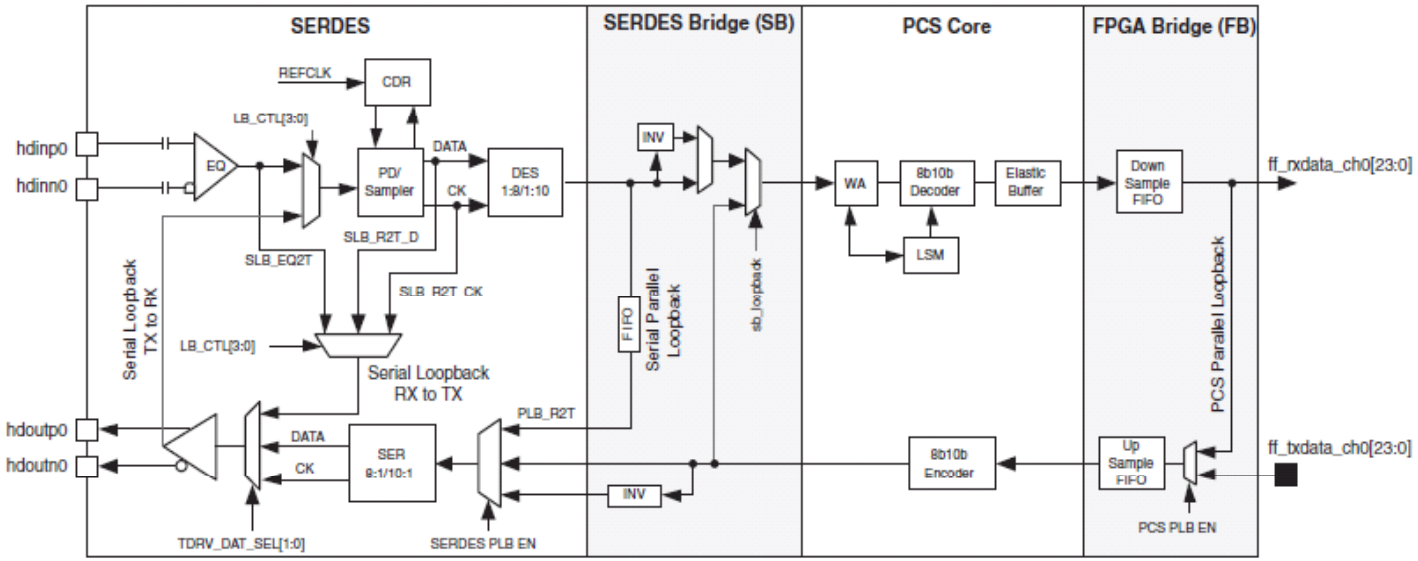
- PON – passive optical network
- FTTH – Fiber To The House



Choice of components

- SERDES
  - FPGA - very flexible ☺
    - Xilinx
      - Recovered Clock available for FPGA
      - Not fixed latency of recovered Clock
      - FPGA clock is not recommended as REFERENCE CLOCK for SERDES
      - $200 /chip
      - High power consumption
    - Altera (?)
    - Lattice ☺
      - Recovered Clock available for FPGA
      - Not fixed latency of recovered clock
      - FPGA clock can be used as REFERENCE CLOCK for SERDES
      - Low SERDES power consumption 100mW/SERDES
      - $40/chip
  - Phy layer chips ☹
    - Inflexible , fixed clock frequency
    - Long term availability
  - Dedicated SERDES chips ☹
    - Not fixed latency of Recovered Clock
    - No differential output for Recovered Clock

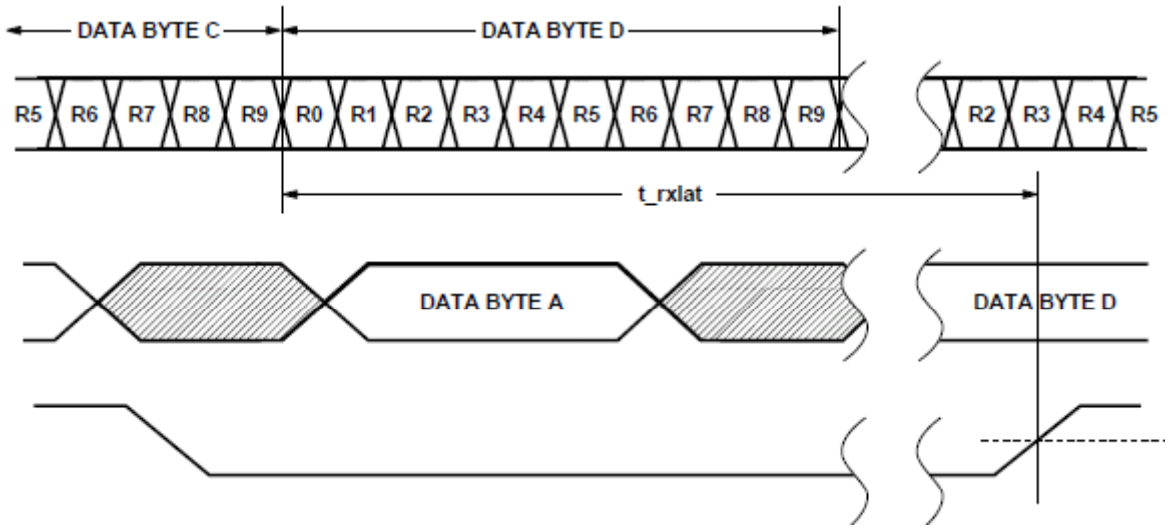
FPGA SERDES, Lattice ECP2M



Item	Description	Min.	Average	Max.	Bypass
Transmit Data Latency					
T1	FPGA Bridge Transmit[2]	1	3	5	1
T2	8b10b Encoder	2	2	2	1
T3	SERDES Bridge Transmit	2	2	2	1
T4	Serializer[3]			2.4	
Receive Data Latency					
R1	Deserializer[3]			1.2	
R2	SERDES Bridge Receive	2	2	2	1
R3	Word Alignment	4	4	4	0
R4	8b10b Decoder	1	1	1	1
R5	Clock Tolerance Compensation	7	15	23	1
R6	FPGA Bridge Receive[2]	1	3	5	1

Two clocks domain

Not fixed Latency of Recovered Clock



$0.2 \leq T\_rxlat \leq 1.2$

T_rxlat not fixed but it has discrete values of 0.3, 0.4, …, 1.2

Latency value is random after every power cycle, reset or resynchronization

Solution:

- Use two Deserializes + FPGA based TDC
- First deserialize - reference
- Second - reset second RX until Latency is minimum

Choice of components II

- Fibre transceivers
  - Single mode single wavelength - two fibers for full duplex mode
  - Single mode BIDIs (1310&1490 nm, 1310&1550 nm) - single fibre for full duplex mode☺

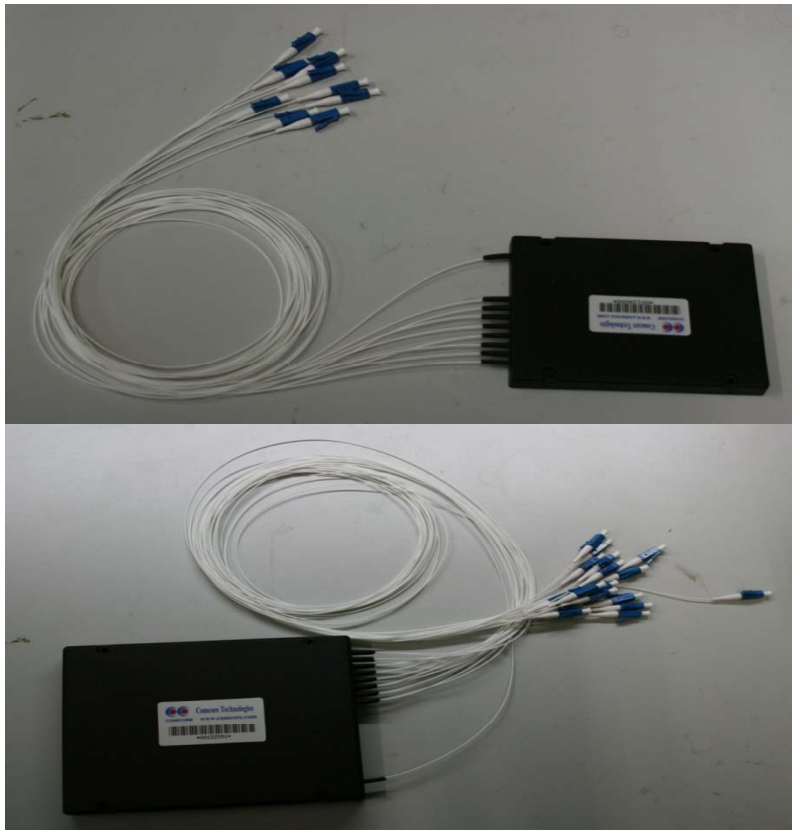
Fibre Transceivers

  - BiDi SFP point to point/ONU with fast laser ON/OFF function
    - Tx:1310 nm
    - Rx:1490 nm or 1550 nm
  - BiDi SFP point to point/OLT
    - Tx:1490 nm or 1550
    - Rx:1310 nm
  - Optical power budjet
    - 22dBm

→ Maximum losses < 19dBm

Choice of components III, passive splitters

- 1x8
  - Insertion losses in both direction:
    9.3 ÷ 9.8 dBm
- 1x16
  - Insertion losses in both direction
    12.2 ÷ 12.9 dBm
- 1x32
  - Insertion losses in both directions
    16 dBm



Problem:

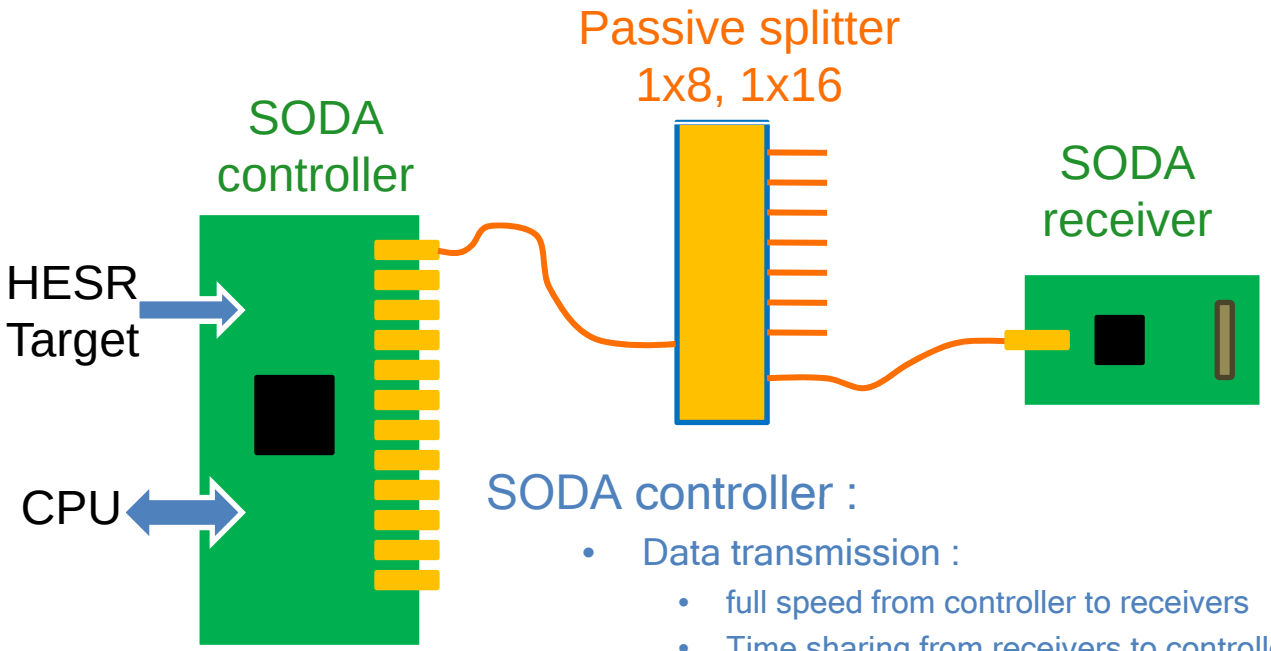
equal loss in both direction require powerful lasers on both sides

Example: 13 dBm transceivers $170

22 dBm transceivers $790

Discussion with NETOPTIC to build optical splitter with asymmetrical insertion losses

SODA architecture



SODA controller :

- Data transmission :
  - full speed from controller to receivers
  - Time sharing from receivers to controller (1/8 or 1/16)
- Lattice SCM FPGA
  - Fast clock synchronization 400 ns(?)

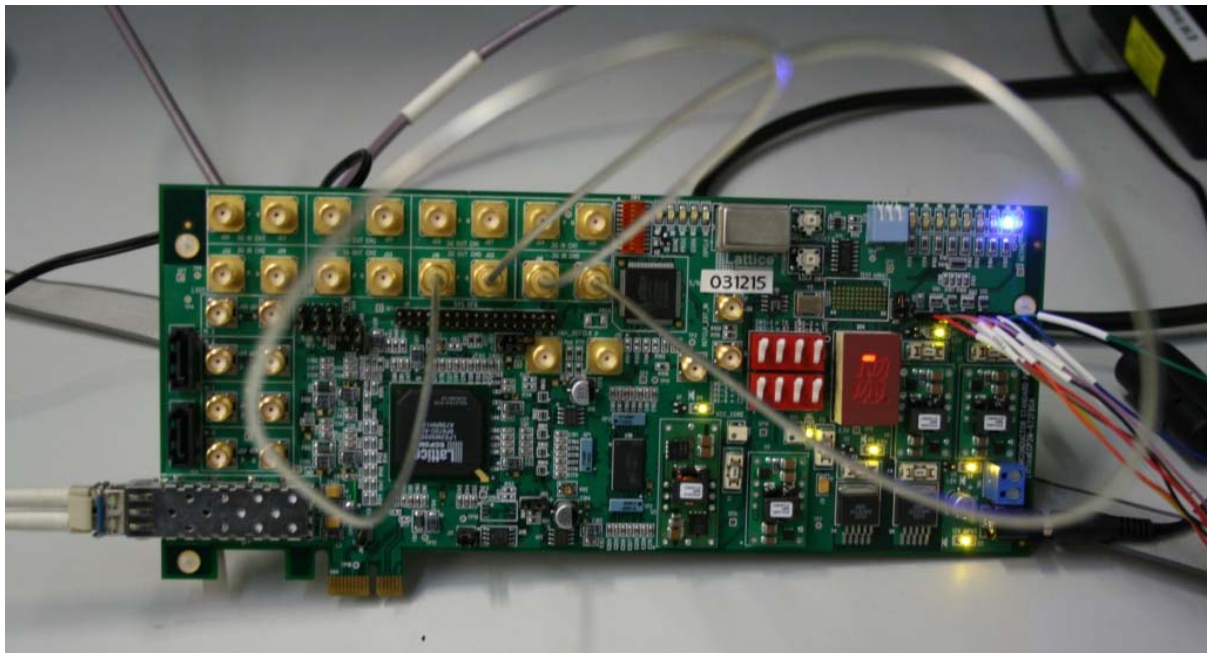
SODA receiver signals:

- OUT CLOCK - LVDS
- OUT RESET - LVTTL, synchronous to CLOCK
- IN/OUT SERDES control - LVDS, data transmission and data flow
- OUT UserDefSignals - LVTTL, synchronous to CLOCK

Test setup

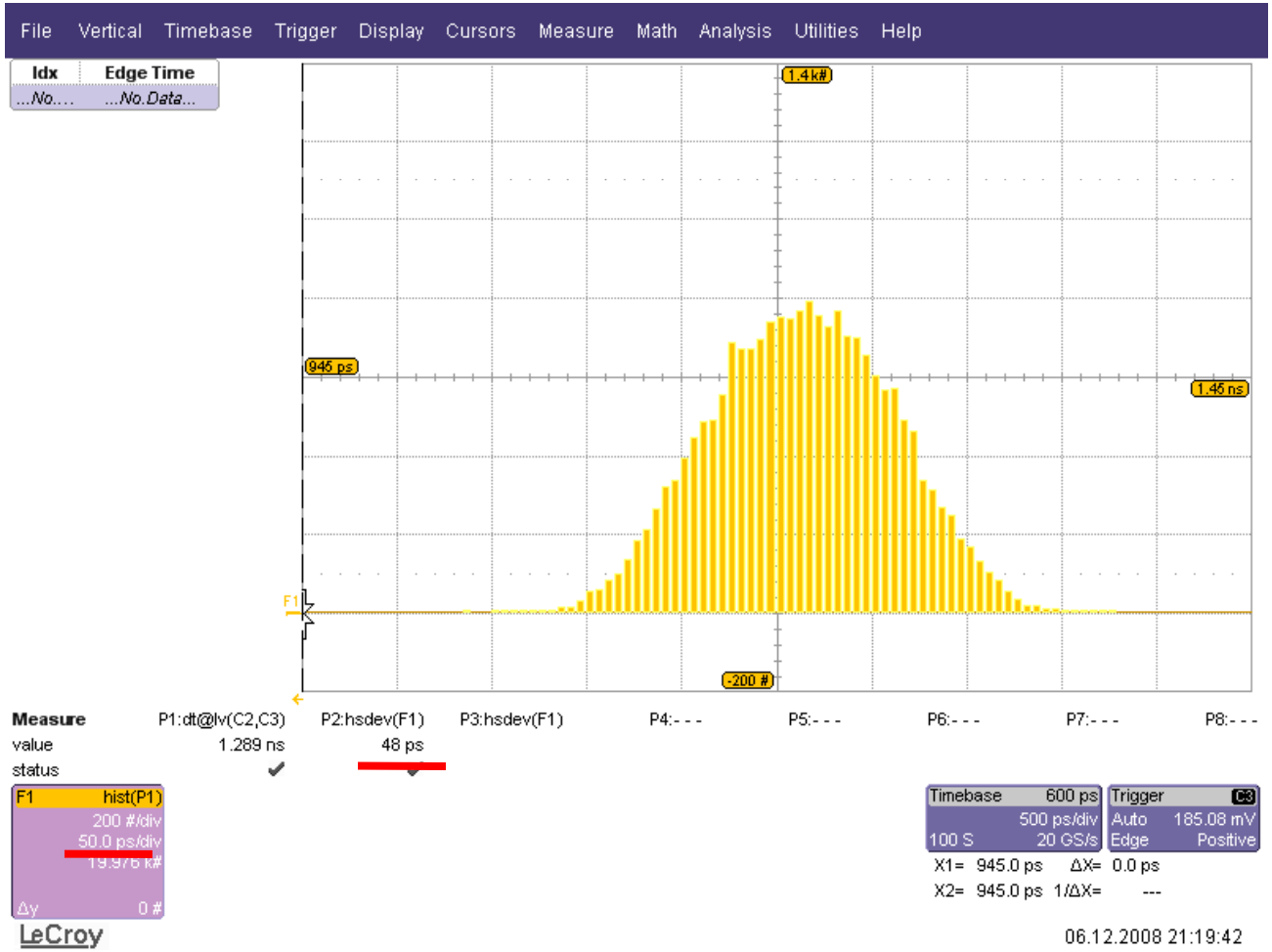
ECP2M evaluation card:

- 100 MHz Reference Clock LVTTL
- 1Gb/s pseudorandom byte pattern
- SFP transceiver : SO-SFP-1000BASE-BX40-53 & SO-SFP-1000BASE-BX40-35
- Passive splitter : 1x8, 1x16



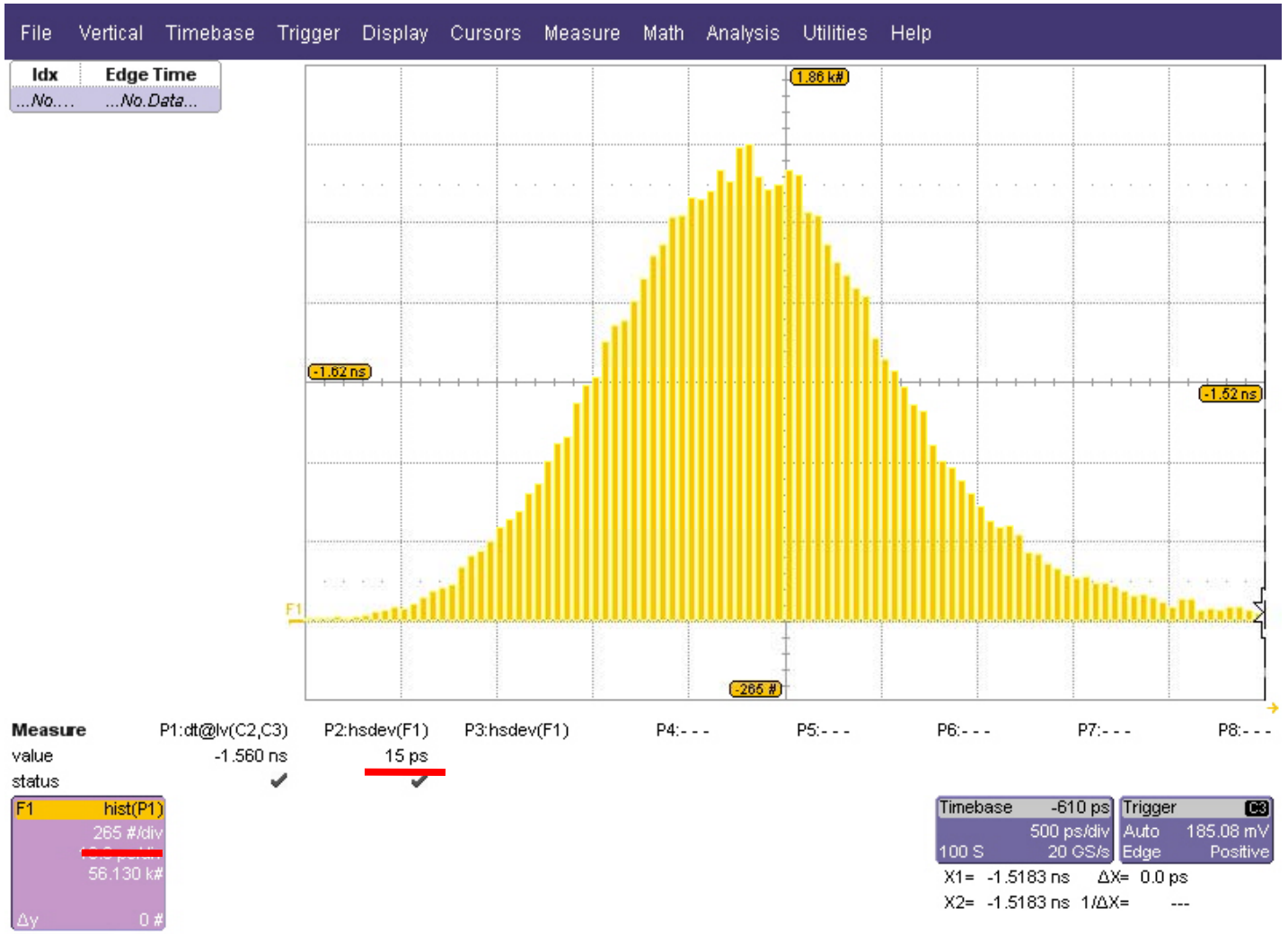
Jitter performance

100 MHz Oscillator clock vs Recovered Rx Clock



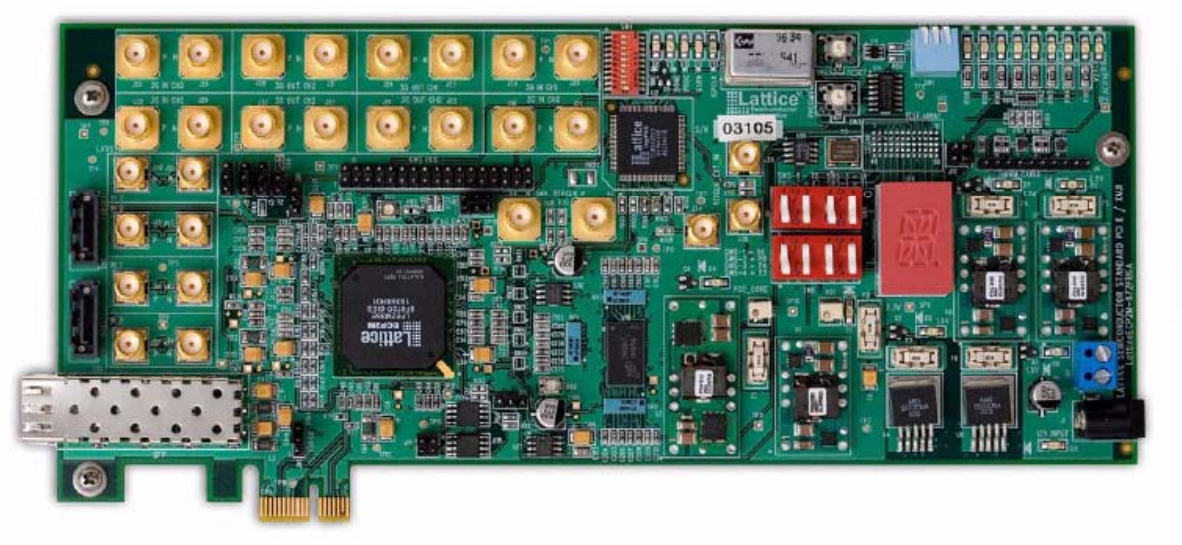
Jitter performance

Reference Tx Clock(after PLL) vs Recovered Rx Clock



SODA controller and receiver

Lattice SC PCIe evaluation card - SODA controller prototype



SODA receiver

- AMC card
- Lattice ECP2M35 256FBGA
- To be delivered this week

Conclusions and outlook

- System components been validated
  - FPGA SERDES validated – clock jitter < 20 ps
  - Fibre transceivers
  - Passive optical splitters
- Next step
  - Implement data transmission protocols with minimum functionality
    - COMPASS protocol for Time Distribution System
- Develop new optical splitter with asymmetrical insertion losses
- Define first application !!

DAQ Architecture