FF-LYNX(*): protocol and interfaces for the control and readout of future Silicon detectors

G. Magazzù INFN – Sezione di Pisa / UCSB Dept. of Physics
WIT-2010 Workshop, Berkeley, February 5st 2010
(*) INFN-Pisa, University of Pisa (Dept. of Information Technology), UCSB (Dept. of Physics)

FF-LYNX: project genesis (1)

- **Standard and flexible solutions are very rare in current HEP (e.g.: LHC) experiments**
- **Each experiment and (very often each detector within the same experiment) developed custom protocols and hardware and software components performing very similar functions.**
- **Examples:**

  Trigger distribution

  Missing clock pulses in the CMS Tracker vs. 5-bit pattern on serial line in the ATLAS Tracker

  "Slow" control

  "Standard" I2C in the CMS strip detector vs. "40-MHz" I2C in the CMS pixel detector

  Data readout

  Analogue readout in CMS strip detector vs. digital readout (characters encoded on discrete analogue levels) in the CMS pixel detector

FF-LYNX: project genesis (2)

- **There are common requirements in control and readout systems of High Energy Physics (HEP) experiments with respect to latency, bandwidth, robustness against transmission errors and component failures, radiation hardness and power dissipation of hardware components.**

- **Common and "standard" solutions for the distribution of Trigger, Timing and Control (TTC) signals and the data readout can be identified and can bring to an optimized design of future control and readout systems in terms of R&D costs an operational capabilities.**

FF-LYNX: “baseline” targets (1)

- **Analysis of possible requirements in future HEP scenarios (e.g.: LHC upgrades) and survey of protocols currently used in HEP experiments (e.g.: in CMS and ATLAS control and readout systems) and in space applications (e.g.: Space-Wire).**

- **Definition of a “standard” and “flexible” protocol for the integrated distribution of TTC signals and data readout.**

- **Protocol validation through high level (System-C) simulations and evaluation of basic Figures Of Merits (e.g.: Loss Trigger Rate vs. Bit Error Rate on physical links).**

- **Development of HDL (VHDL) models of TX and RX interfaces implementing the FF-LYNX protocol and validation through HDL functional simulations**

FF-LYNX: “baseline” targets (2)

- **Development of an FPGA based emulator of links implementing the FF-LYNX protocol: synthesis and validation of the interfaces.**

- **Development of the TX and RX interfaces as custom low power and radiation tolerant macros, designed and produced in a commercial CMOS technology (≙130nm).**

- **Test and characterization (including irradiation tests) of test circuits including prototypes of the TX and RX interfaces and development of a library of IP-Cores available to designers of ICs for the future experiments.**

FF-LYNX: “spin-off” activities

- **Development of further version of the FF-LYNX protocol implementing new capabilities (requirements from the HEP community)**
- **Development of an Integrated Simulation Environment (ISE), a test bench based on high level (e.g.: System-C) models and “realistic” test vectors (i.e.: derived from GEANT4 simulations), to validate and compare protocols, implementations of trigger algorithms under different hypotheses on sensor geometry, detector architecture and working conditions (i.e.: luminosity, error rates).**
- **Development of a General Purpose Emulator (GPE), an FPGA based “proof of concept” system for the integrated distribution of TTC signals and data readout based on the FF-LYNX protocol, to be used also for test of prototypes of hardware components (e.g.: physical links, ASICs, detector modules).**
- **Development of additional IP cores (e.g.: Data Concentrator, Redundancy Manager), to be used as building blocks of TTC and readout systems.**

FF-LYNX: key features (1)

- **Compatibility with serial electrical links**
  - **"double-wire" → separate clock and data lines, no need of clock recovery devices in receivers , long (~1m) links sensitive to skew effects at high (~500Mbps) speed (V.1: 2009/2010)**
  - **"single-wire" → clock and data encoded onto one line, "8b/10b like" encoding in the FRM channel, long links not sensitive to skew effects at high speed (V.3: 2011/2012)**

- **Flexibility w.r.t. data rates**

  Different speed options available to fit different bandwidth requirements: 4xF, 8xF, 16xF (F = frequency of the reference clock) → 160 Mbps, 320 Mbps, 640 Mbps in LHC (F = 40MHz)

FF-LYNX: key features (2)

Integrated distribution of TTC signals and DAQ

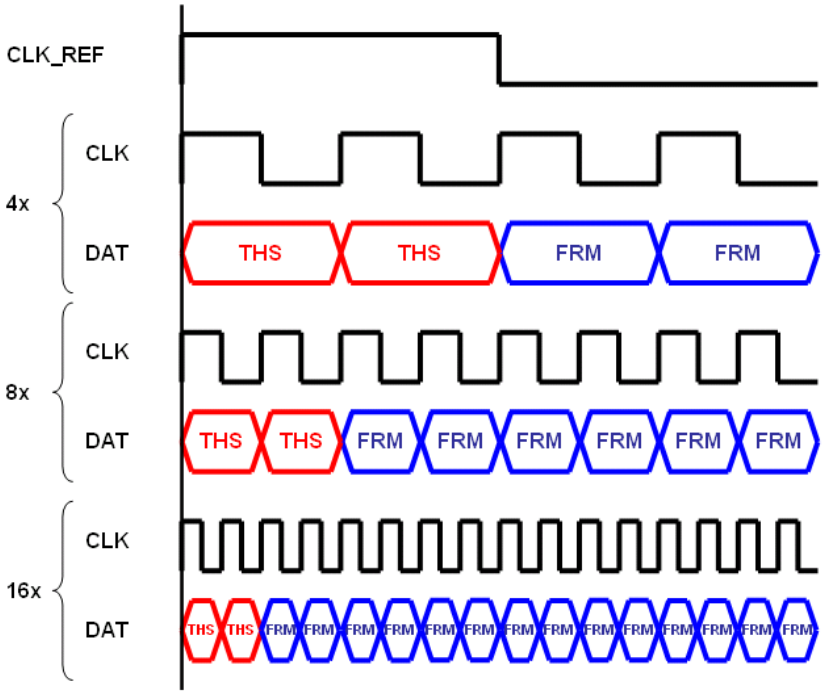
- Transmission of triggers, controls and data handled by the same protocol and hardware components (e.g.: interfaces and physical links)
- Two channels multiplexed in the time domain:

THS channel

- Triggers, frame headers and synchronization patterns
- 2 bits in each cycle of the reference clock
- 6-bit error robust encoding ⇒ patterns detected and timing correctly reconstructed also with single bit flips, double bit flips detected)

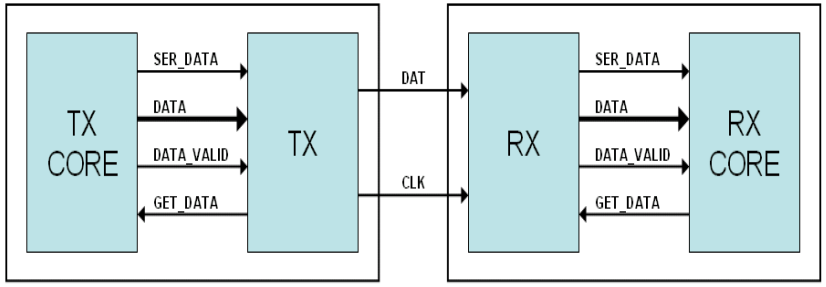
FRM channel

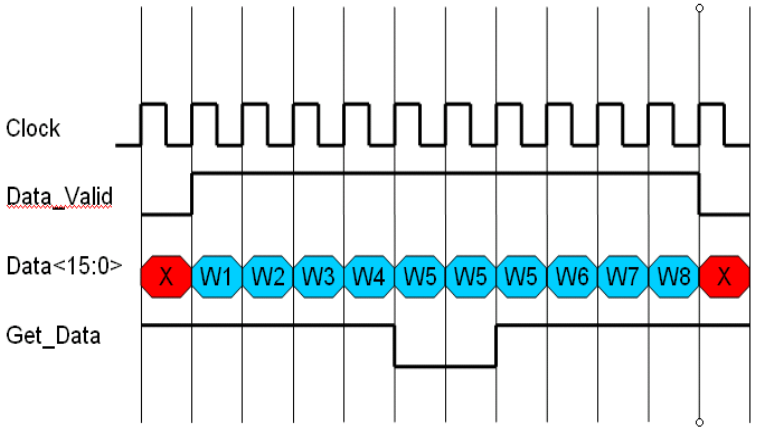
- Data frames
- 2 (4x), 6 (8x) or 14 (16x) bits in each clock cycle (ref. clock)
- Frames associated to headers transmitted in the THS channel



FF-LYNX: key features (3)

- **Easy coupling of TX and RX interfaces to host devices**
  - **serial and parallel (16-bit) ports available**
  - **simple handshaking based on data_valid and get_data lines (get_data for optional flow control)**
    - **new data in the source → data_valid = 1 and data available on the data port**
    - **more data available and get_data = 1 (i.e.: input buffers in the destination not full) → continue data transfer;**
    - **more data available and get_data = 0 → stop data transfer until get_data = 1**



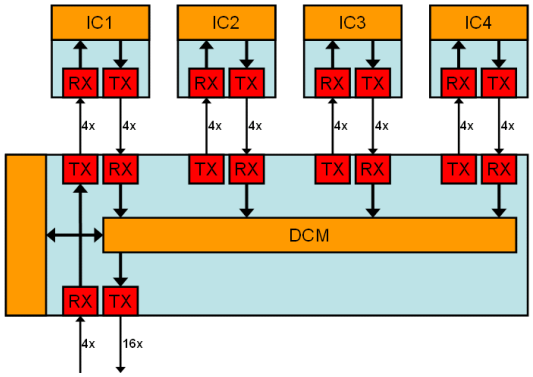


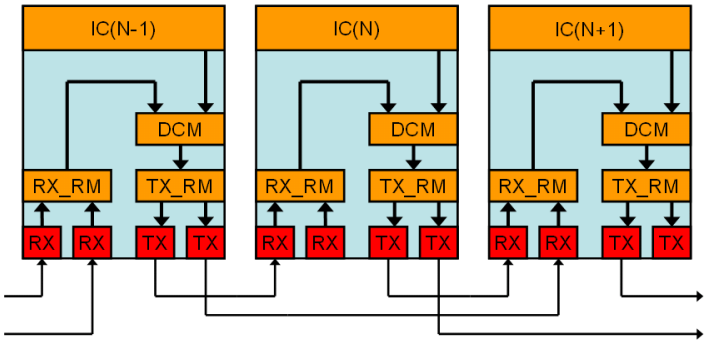
FF-LYNX: key features (4)

- **General structure of data frames (transparent w.r.t. data type)**
  - **Frame Descriptor**
  - **Label (Optional)**
  - **Payload**
  - **CRC (Optional)**
- **Different latency options in frame transmission**
  - **Variable Latency (VL) Frames (V.1: 2008/2009) → no fixed size, no privileges in frame transmission, suitable for configuration and monitoring data and "raw" data (transmitted from Front-End ASICs after the reception of a L1 Trigger)**
  - **Fixed Latency (FL) Frames (V.2: 2009/2010) → pre-defined size, transmitted with the highest priority in TX interfaces, suitable for "trigger" data (e.g.: hit timing and position) to be used in the generation of the L1 Trigger**
- **Robustness of critical information against transmission errors**
  - **6-bit robust encoding for triggers, frame headers and synchronization patterns (single bit flips detected and corrected, double bit flips detected)**
  - **Hamming encoding for frame descriptors**
  - **Optional Cyclo Redundancy Check (CRC) for the payload**

FF-LYNX: key features (5)

- **Flexibility with respect to system architecture**
  - compatible with “star” and “ring” topologies of the control and readout systems
  - additional modules required: Data Concentrator (DC), Redundancy Manager (RM)





“Star” topology

- DCM → input data streams (e.g.: 4x) from FE ASICs are merged into one output stream (e.g.: 16x)
- DCM → event building is optionally performed: input frames with the same label (e.g.: time stamp or event number in “raw” data) can be merged into one output frame

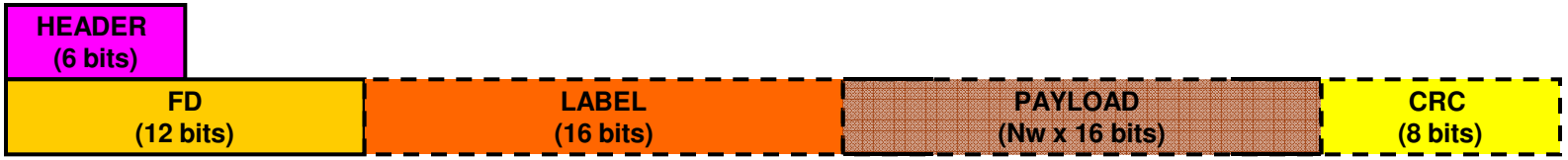
“Ring” topology

- DCM → data generated locally are merged with data received from the previous nodes and event building is optionally performed
- RM → redundancy against component failures is handled (i.e.: faulty nodes are bypassed)

FF-LYNX: frame Structure (1)

Variable Latency (VL) frames

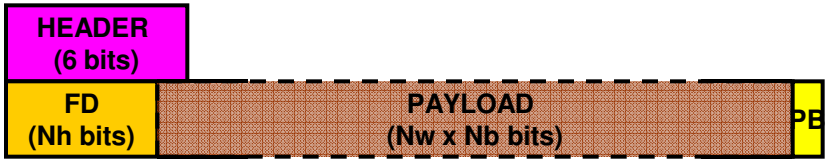
- **Frame Descriptor (7/12 Hamming Encoding)**
  - **Frame Length (4 bits) → number of words (16-bits) in the payload (including the optional label)**
  - **Data Type (1 bit) → data type (i.e.: configuration/monitoring data or "raw" data)**
  - **Label On (1 bit) → optional label included**
  - **Last Frame (1 bit): → last frame associated to a data packet**
- **Label (16-bits) → optional field containing information associated to the payload (e.g.: address and operation code of commands, time stamp or trigger number of "raw" data)**
- **Payload → 16-bit data words (0 → 15)**
- **CRC (8-bits) → optional field for Cycle Redundancy Check**



FF-LYNX: frame Structure (2)

Fixed Latency (FL) frames

- Frame Descriptor → Hamming Encoding (N_{wh} bits) of the number N_w of N_b-bit data words of the the payload
- Payload ($N_w \cdot N_b$ bits)→ N_w data words (N_b bits each)
- Parity bit → Payload Parity

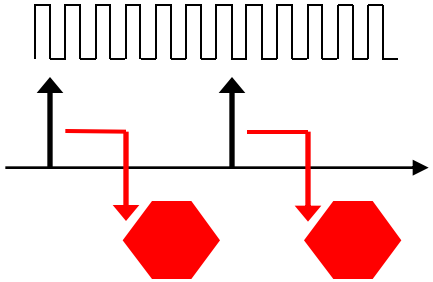


FL frames & "trigger" data

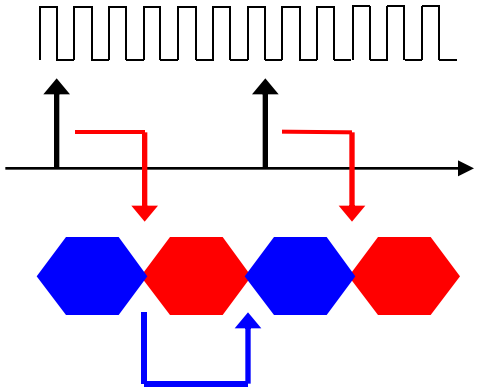
- "Trigger" data (e.g.: hit timing and address) from Front-End ASICs to the L1 trigger processor should have a fixed and constant latency ⇒ they can be the payload of FL frames
- Each frame can contain one or more "trigger" data

FL Frames & Trigger Data(1)

- **When a hit is detected the FE ASIC start the transmission of a FL frame**

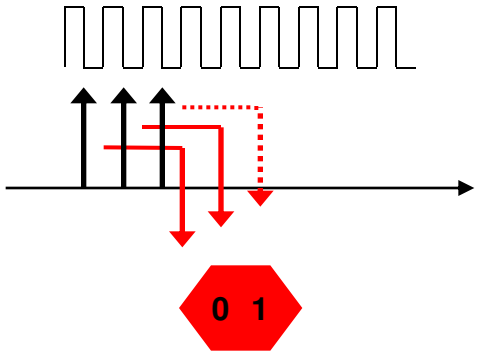


- **If the transmission of a VL frame is ongoing, it is stopped and it restarts only when the transmission of the FL frame is completed**



FL Frames & Trigger Data(2)

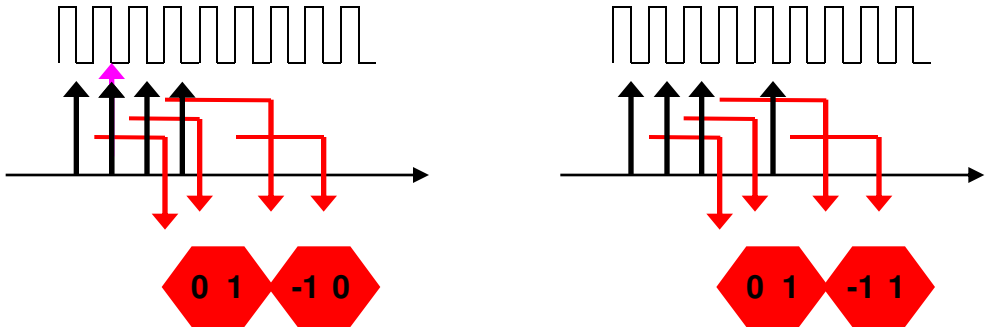
- **Hit data associated to other hits detected during the transmission of the FL frame can be inserted in the payload of the FL frame (with their relative timing w.r.t. the first hit) up to the maximum number of hits that can be associated to the FL frame**



- **The hit timing can be reconstructed in the destination node by adding latency of FL frame and relative hit timing**

FL Frames & Trigger Data(3)

- **Hit data that are not included in the FL frame because exceeding the frame size can be recovered by transmitting a FL frame immediately after the first if their timing can be specified as a negative relative timing in the hit timing bits**



- **Frame size and link speed can be fixed to fit the required transmission efficiency (i.e.: “hit loss rate”)**

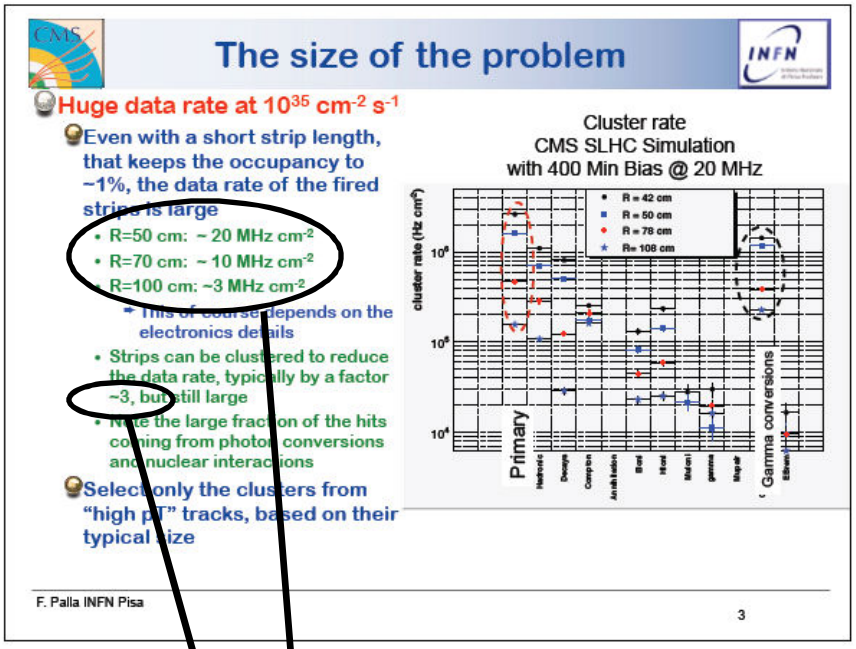
FL Frames & Trigger Data(4)

- **Link speed = 320 Mbps – Frame Size = 3 clock cycles (18 bits) → 2 Hits**
  - **Hit Address → 5 bits - Hit Timing (0,1,2) → 2 bits ⇒ Hit Data (2x7)= 14 bits**
  - **Hit Number (1,2 – Hamming 1/3) → 3 bits – Parity → 1 bit**

- **Link speed = 320 Mbps – Frame Size = 5 clock cycles (30 bits) → 3 Hits**
  - **Hit Address → 5 bits - Hit Timing (0,…,4) → 3 bits ⇒ Hit Data (3x8)= 24 bits**
  - **Hit Number (1,2,3 – Hamming 2/5) → 5 bits – Parity → 1 bit**

- **Link speed = 320 Mbps – Frame Size = 8 clock cycles (48 bits) → 5 Hits**
  - **Hit Address → 5 bits - Hit Timing (0,…,7) → 3 bits ⇒ Hit Data (5x8)= 40 bits**
  - **Hit Number (1,2,3,4,5 – Hamming 3/7) → 7 bits – Parity → 1 bit**

FL Frames & Trigger Data(5)



- R= 100cm → cluster_rate/FE ~ 5 MHz
  - Ncl → 2 clusters / 3 clock cycles
  - Data Rate = 180 Mbps
- R= 70cm → cluster_rate/FE ~ 15 MHz
  - Ncl → 3 clusters / 5 clock cycles
  - Data Rate = 300 Mbps
- R= 50cm → cluster_rate/FE ~ 30 MHz
  - Link speed = 320 Mbps
  - FRM Size = 5 (30 bits/3 clusters)

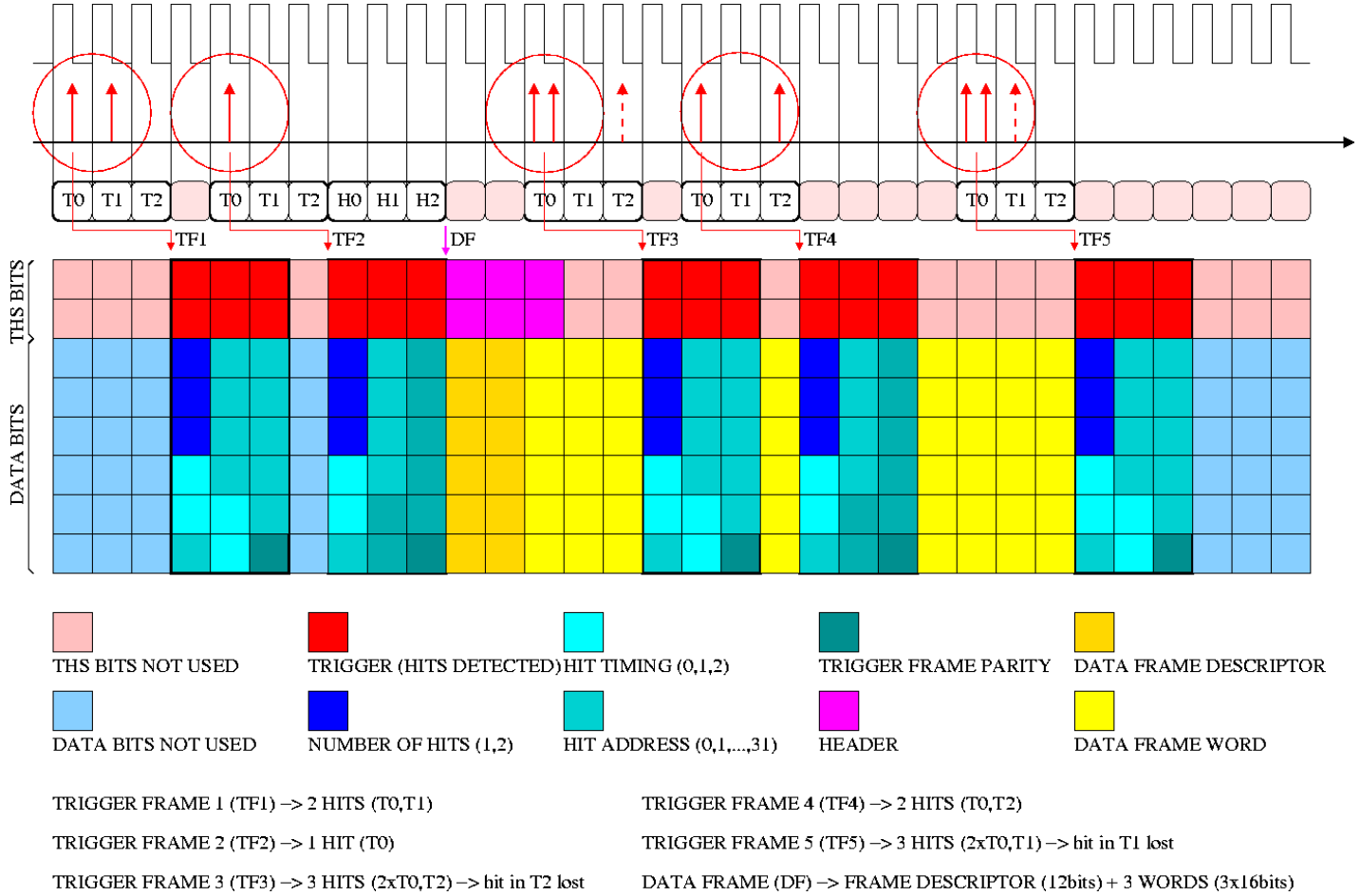
FE Area ~ 4,6cm² ⇒

FL Frames & Trigger Data(6)

- Different possible configurations w.r.t. link speed and frame length have been simulated in order to evaluate their efficiency in the transmission of trigger data.
- Values of the Trigger data Transmission Efficiency (TTE) for different values of Link Speed (LS), frame length expressed in clock cycles (NC) and maximum number of transmitted hits (NH) are here summarized.
- These results have been obtained assuming a Poisson distribution for hits (0.125 hits/clock cycle) and hits generated by one FE circuit in simulations with LS = 8xF and by four FE circuits in simulations with LS = 16xF.

LS	NC	NH	TTE
8x	3	2	96,57%
8x	5	3	98,53%
16x	4	7	97,29%
16x	8	13	98,94%

FL Frames & Trigger Data(7)



TRIGGER FRAME 1 (TF1) –> 2 HITS (T0,T1)

TRIGGER FRAME 2 (TF2) –> 1 HIT (T0)

TRIGGER FRAME 3 (TF3) –> 3 HITS (2xT0,T2) –> hit in T2 lost

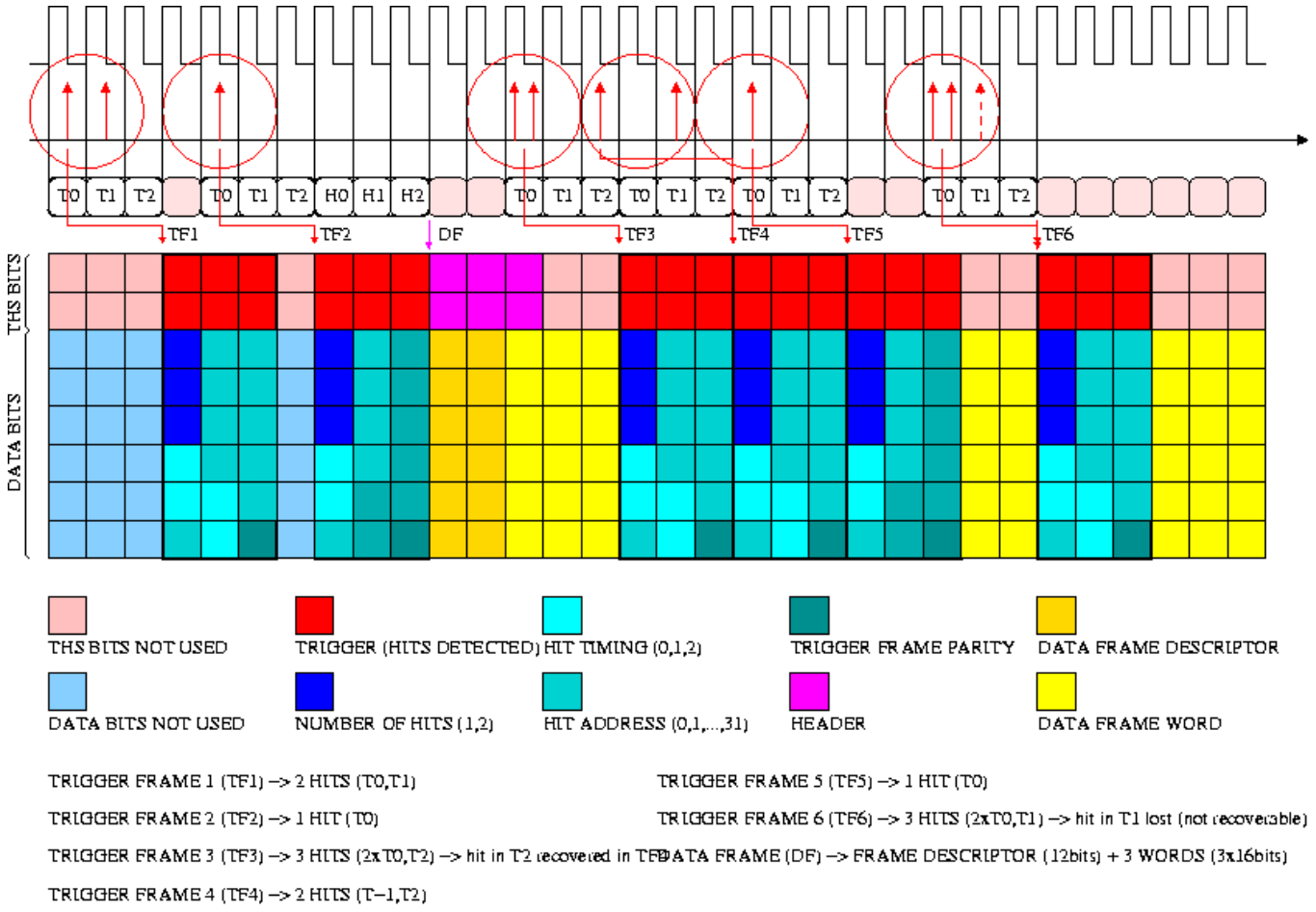
TRIGGER FRAME 4 (TF4) –> 2 HITS (T0,T2)

TRIGGER FRAME 5 (TF5) –> 3 HITS (2xT0,T1) –> hit in T1 lost

DATA FRAME (DF) –> FRAME DESCRIPTOR (12bits) + 3 WORDS (3x16bits)

Link speed = 8xF, frame size = 3 cycles (24 bits), 7 bits/hit (2 bits for hit timing, 5 bits for hit position) → up to 2 hits transmitted in each FL frame, no hit recovery

FL Frames & Trigger Data(8)

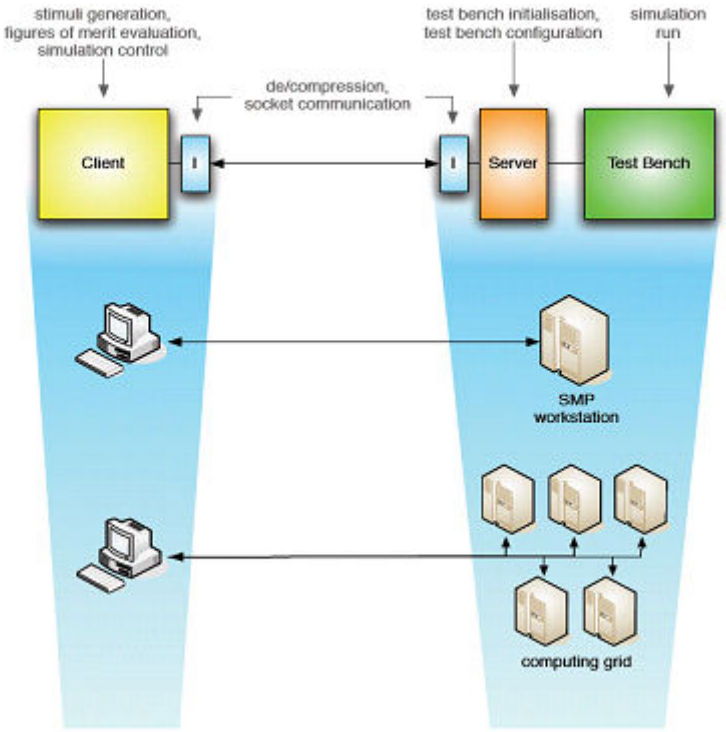


Link speed = 8xF, frame size = 3 cycles (24 bits), 7 bits/hit (2 bits for hit timing, 5 bits for hit position) → up to 2 hits transmitted in each FL frame, hit recovery

High Level Simulations (1)

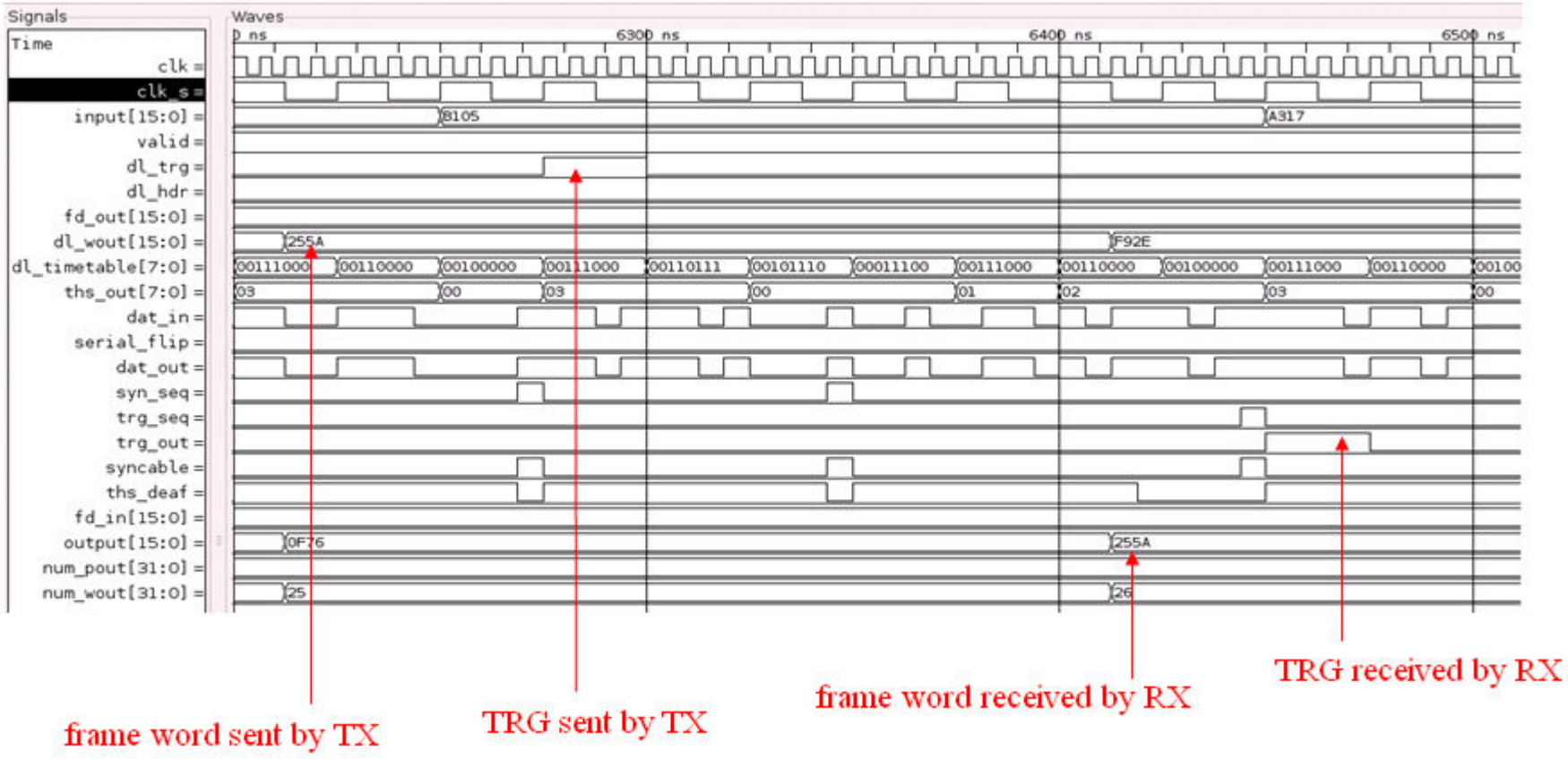
High Level Simulator

- High-level System-C models of TX and RX interfaces to validate the protocol, including its robustness against transmission errors and evaluate figures of merits (e.g.: Trigger Loss Rate, Packet Loss Rate, Packet Latency).
- Client-server architecture to take advantage of socket communication and to provide a scalable environment suitable for Symmetric Multi-Processing (SMP) workstations or computing grids (e.g. 1500-processors grid at INFN-Pisa).
- Reusability of the simulator in FPGA based emulators by replacing the server section with the emulator without modifying the client section.



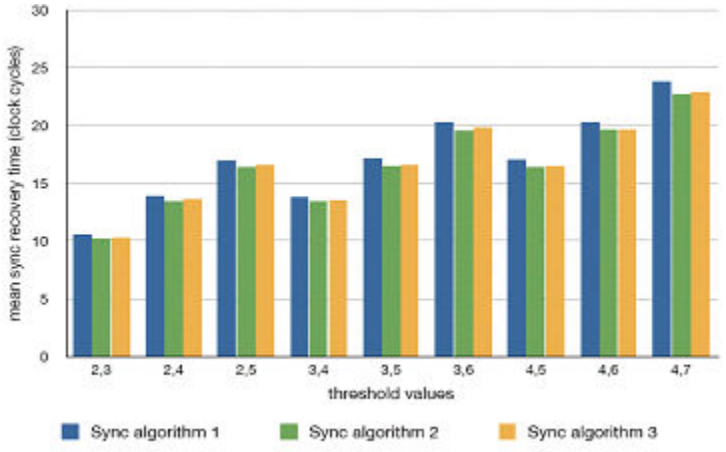
High Level Simulations (2)

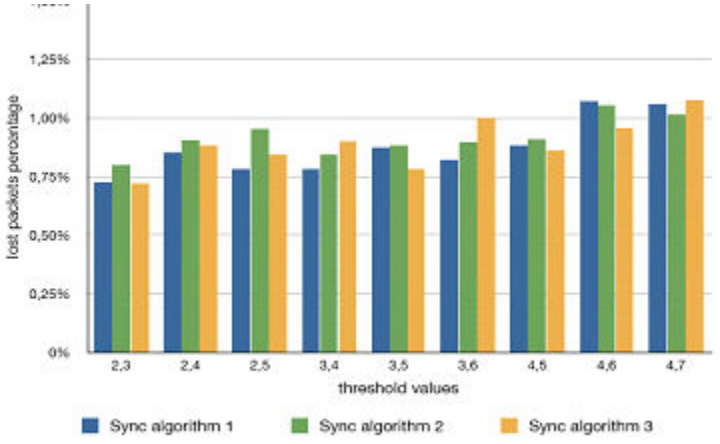
- High-Level simulations give the opportunity to analyze in detail the protocol behavior …



High Level Simulations (3)

- … and to tune protocol parameters according to results of performance analysis.
- E.g.:
  - three different algorithms for the synchronization between TX and RX interfaces (i.e.: detection of the THS channel and recovery of the reference clock) have been evaluated, based on different thresholds in pattern counting
  - plots show mean Synchronization Recovery Time and Packet Loss Rate due to errors on the clock line (error rate = 10-6) with different threshold values





Integrated Simulation Environment (1)

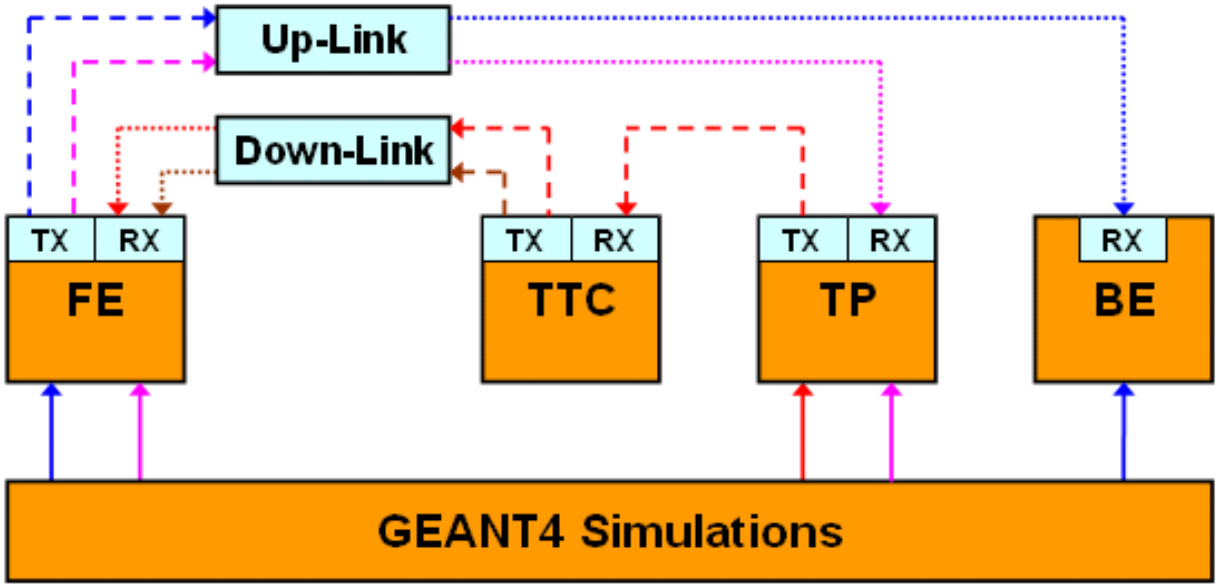
Integrated Simulation Environment (ISE)

- **Physics simulations (GEANT4) performed with different sensor geometries (e.g.: pixel or strip) and different values of the luminosity and of the distance from the interaction point (i.e.: r and z).**
- **Models of Front-End (FE) electronics with input data from physics simulations and emulating different algorithms for the data generation (e.g.: zero-suppression for "raw" data, clustering of pixel or strips for "trigger" data).**
- **Models of the L1 Trigger Processor (TP) implementing different possible trigger algorithms (e.g.: track finding on "trigger" data from different layers in order to identify high-Pt tracks).**
- **Models of the Back-End (BE) electronics emulating "off-line" data analysis**
- **Models of the TTC system (TTC) emulating the distribution of triggers (from TP) and control signals.**
- **Models of the links (Up-Link and Down-Link): TX and RX interfaces embedded in FE, TP, TTC and BE models and optical and electrical links including error injection due to radiation and noise.**

Integrated Simulation Environment (2)

Developed for the validation of the FF-LYNX protocol

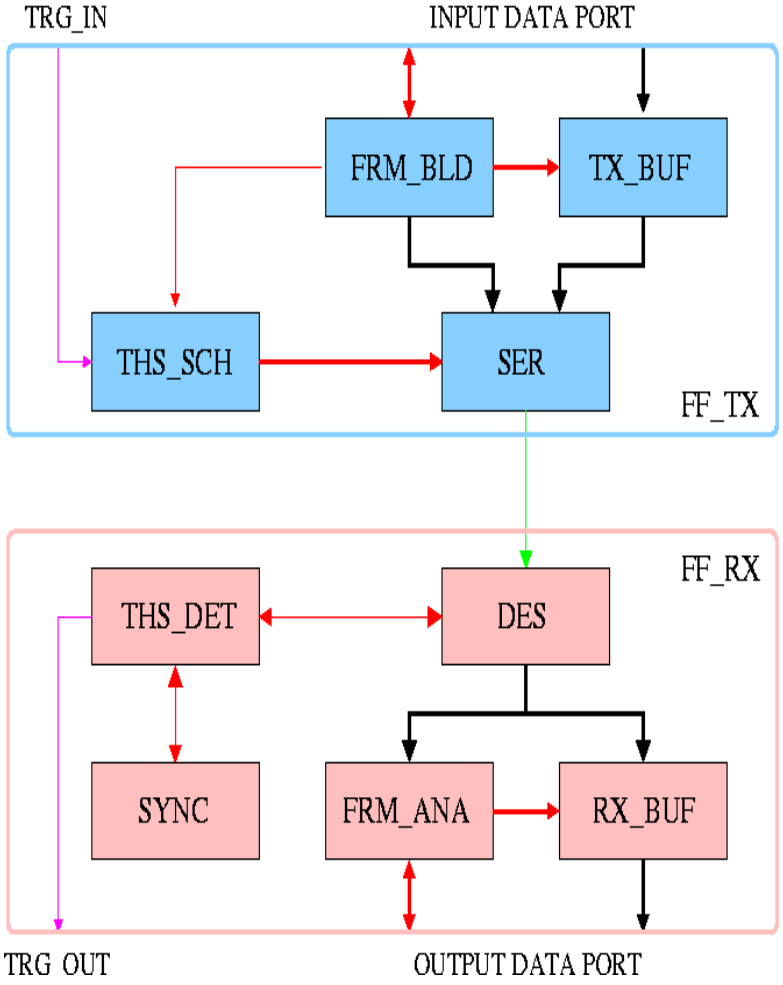
Not yet developed



- Blue → "raw" data
- Magenta → "trigger" data
- Brown → commands
- Red → L1 triggers

- ⁄ → ⁄ → ⁄ → "data losses"

TX and RX interfaces



Architecture of the FF-TX and FF-RX interfaces

FF-TX

- **TX Buffer (TX_BUF) → Buffering of input data**
- **Frame Builder (FRM_BLD) → Assembly of data frames**
- **THS Scheduler (THS_SCH) → Arbitration between Triggers and Frame Headers**
- **Serializer (SER) → Generation of the output serial stream**

FF-RX

- **Deserializer (DES) → Extraction of THS and FRM data from the input serial stream**
- **THS Detector (THS_DET) → Detection of patterns associated to Triggers, Frame Headers and Sync commands in the THS channel**
- **Synchronizer (SYNC) → Recovery and synchronization of the reference clock and detection of THS and FRM channels in the input stream**
- **Frame Analyzer (FRM_ANA) → Analysis of the frame descriptor and control of the data transfer to the output buffer**
- **RX buffer (TX_BUF) → Buffering of output data**

HDL Simulations

- **A VHDL test bench based on the control and readout system of the CMS pixel detector has been developed for the functional verification and the performance evaluation of the VHDL models of the FF-LYNX interfaces**

Input: "Hit file" describing hit pixel addresses and amplitudes in a simple custom text format.

Example:

[Time index] – [L1T] – [1st Hit Data] ... [last Hit Data] ;
where
[Hit Data] = ([ROC num] , [column addr (DC number)] , [row addr] , [hit ampl])

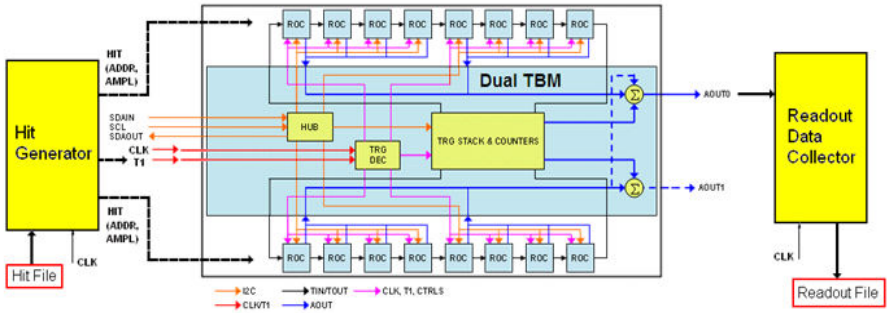
```
11 - 0 - (0, 12, 01, 3)(2, 20, 140, 3);
30 - 0 - (1, 3, 158, 4)(1, 15, 59, 3);
40 - 0 - (0, 8, 152, 6)(0, 17, 18, 5)(2, 6, 47, 3);
50 - T - (0, 14, 52, 5)(1, 11, 9, 5)(1, 18, 112, 5)(2, 7, 129, 5)(2, 4, 146, 5)(2, 10, 147, 5);
63 - 0 - (0, 2, 15, 5)(1, 21, 123, 3)(2, 24, 99, 3);
70 - T - (0, 11, 151, 5)(2, 4, 14, 3);
80 - T - (0, 4, 116, 5)(1, 6, 85, 3)(1, 15, 73, 3)(1, 22, 47, 3)(2, 16, 95, 3);
90 - 0 - (2, 2, 147, 5);
100 - 0 - (1, 23, 114, 5)(1, 15, 78, 3)(2, 17, 55, 3);
110 - T - (0, 3, 75, 5)(0, 24, 26, 3)(1, 16, 117, 3)(1, 14, 35, 3)(1, 18, 79, 3)(1, 5, 47, 3)(2, 9, 98, 3);
130 - T - (0, 19, 47, 5)(2, 20, 59, 3)(2, 17, 72, 3);
204 - 0 - (0, 16, 46, 5);
```

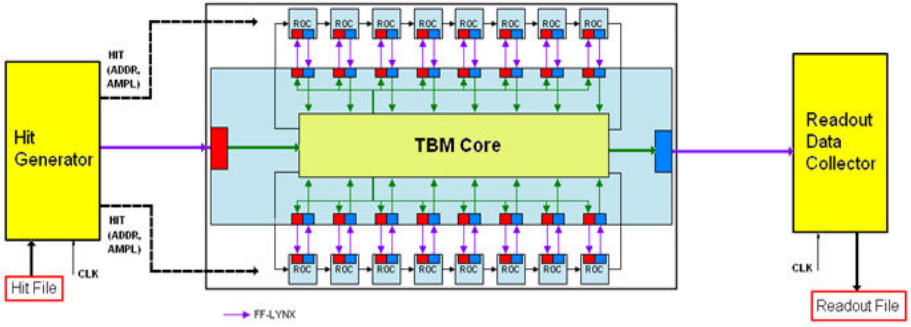
Output: "Readout file" listing readout data in a simple custom text format.

Example:

[Event number] [1st Hit Data]...[last Hit Data] [Status byte] ;

```
1 (0, 14, 52, 5)(1, 11, 9, 5)(1, 18, 112, 5)(2, 4, 146, 5)(2, 7, 129, 5)(2, 10, 147, 5)00000000 ;
2 (0, 11, 151, 5)00000000 ;
3 (0, 4, 116, 5)(1, 6, 85, 3)(1, 15, 73, 3)(1, 22, 47, 3)(2, 16, 95, 3)00000000 ;
4 (0, 3, 75, 5)(0, 24, 26, 3)(1, 5, 47, 3)(1, 14, 35, 3)(1, 16, 117, 3)(1, 18, 79, 3)(2, 9, 98, 3)00000000 ;
5 (0, 19, 47, 5)(2, 17, 72, 3)(2, 20, 59, 3)00000000 ;
```





FPGA Emulator (1)

- **An emulator of links implementing the FF-LYNX protocol has been developed on an "off-the-shelf" platform (Altera EP3SL150 - Stratix III)**
- **Advantages:**
  - **Real-Time system → much faster than simulations running on PCs**
  - **Reusable as "test bed" for physical links (e.g.: cables) and test circuits**

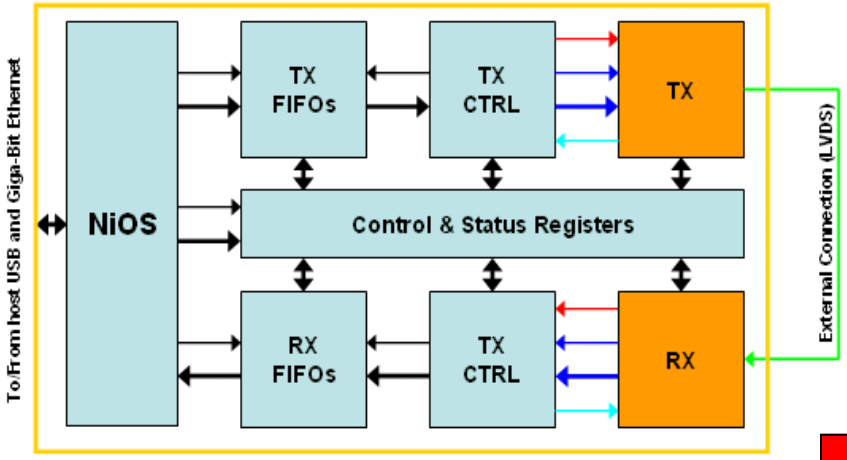
High Level Simulator

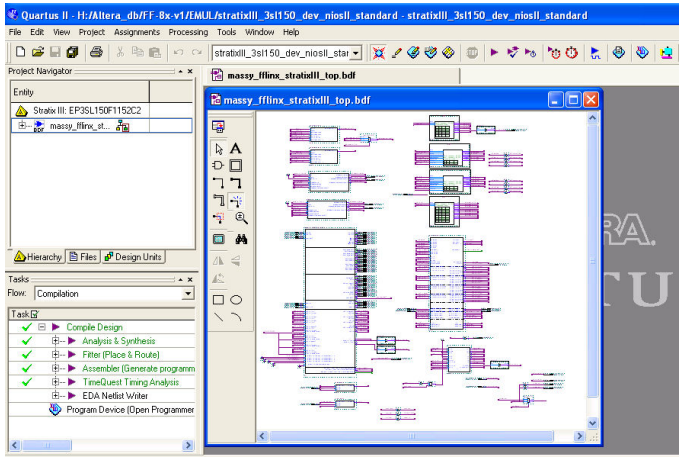
Tsim = 60 s, Speed: 4x
TRG rate: 400 KHz
PCK rate: 400 KHz - PCK size: 5
Execution Time
Intel Core 2 Duo @ 2.5 GHz → 11h 47m 18s
Intel Xeon 8-core @ 1.6 GHz → 2h 21m 23s
INFN GRID (400+ CPUs) → 3m 21s

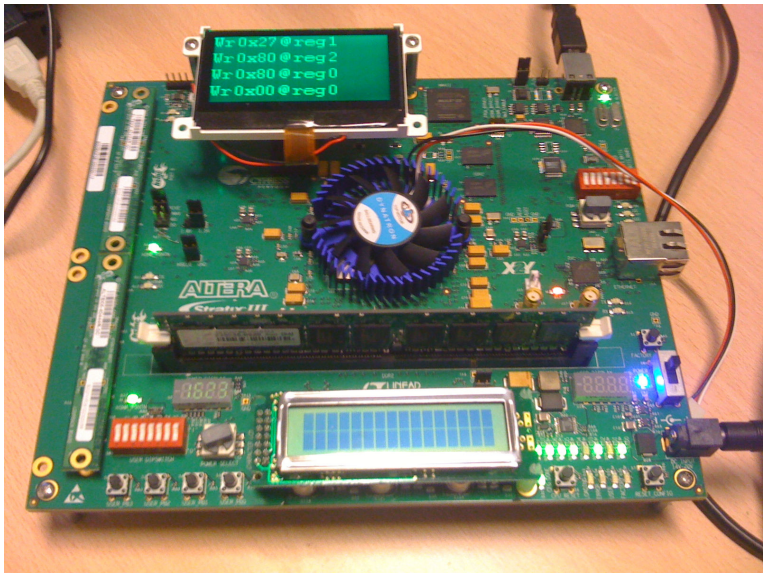
FPGA Emulator

Tsim = 60 s, Speed: 8x
TRG rate: 133 KHz
PCK rate: 133 KHz - PCK size: 8
Execution Time → 6m 40s

FPGA Emulator (2)

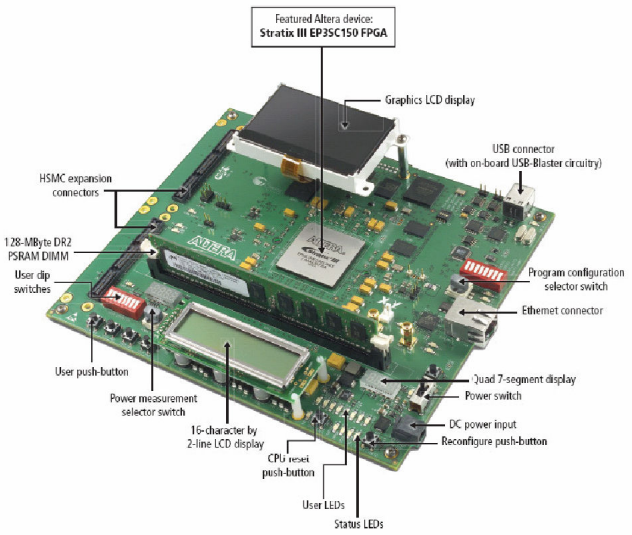


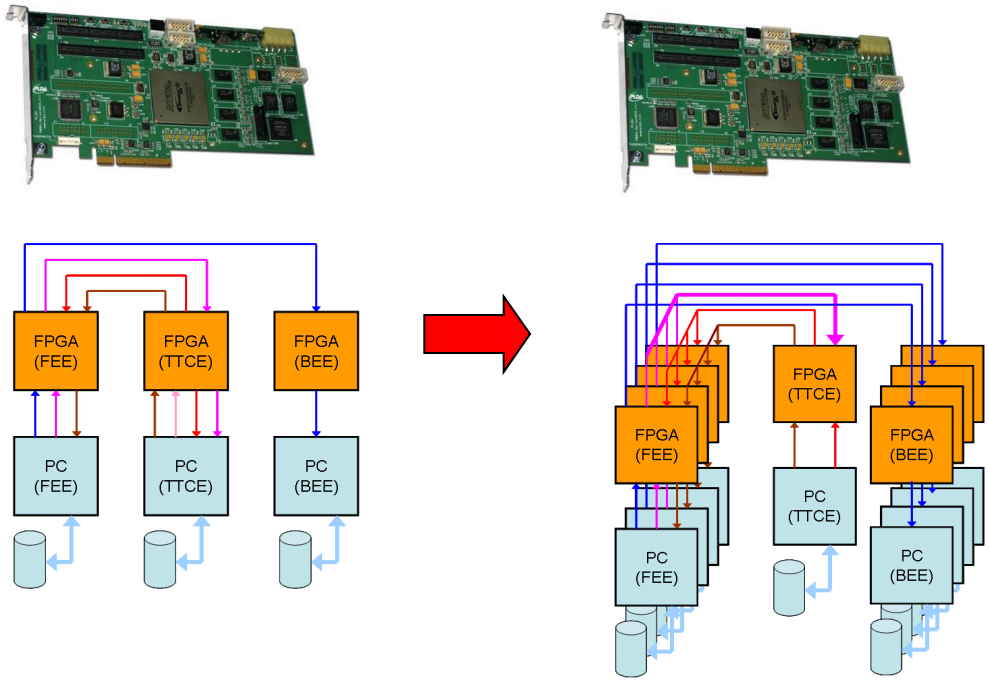




General Purpose Emulator

- FPGA “test-bed” toward a modular and scalable General Purpose Emulator based on PCI-Express FPGA platforms (FEE = Front-End Emulator, BEE = Back-End Emulator, TTCE = TTC + Trigger Processor Emulator)



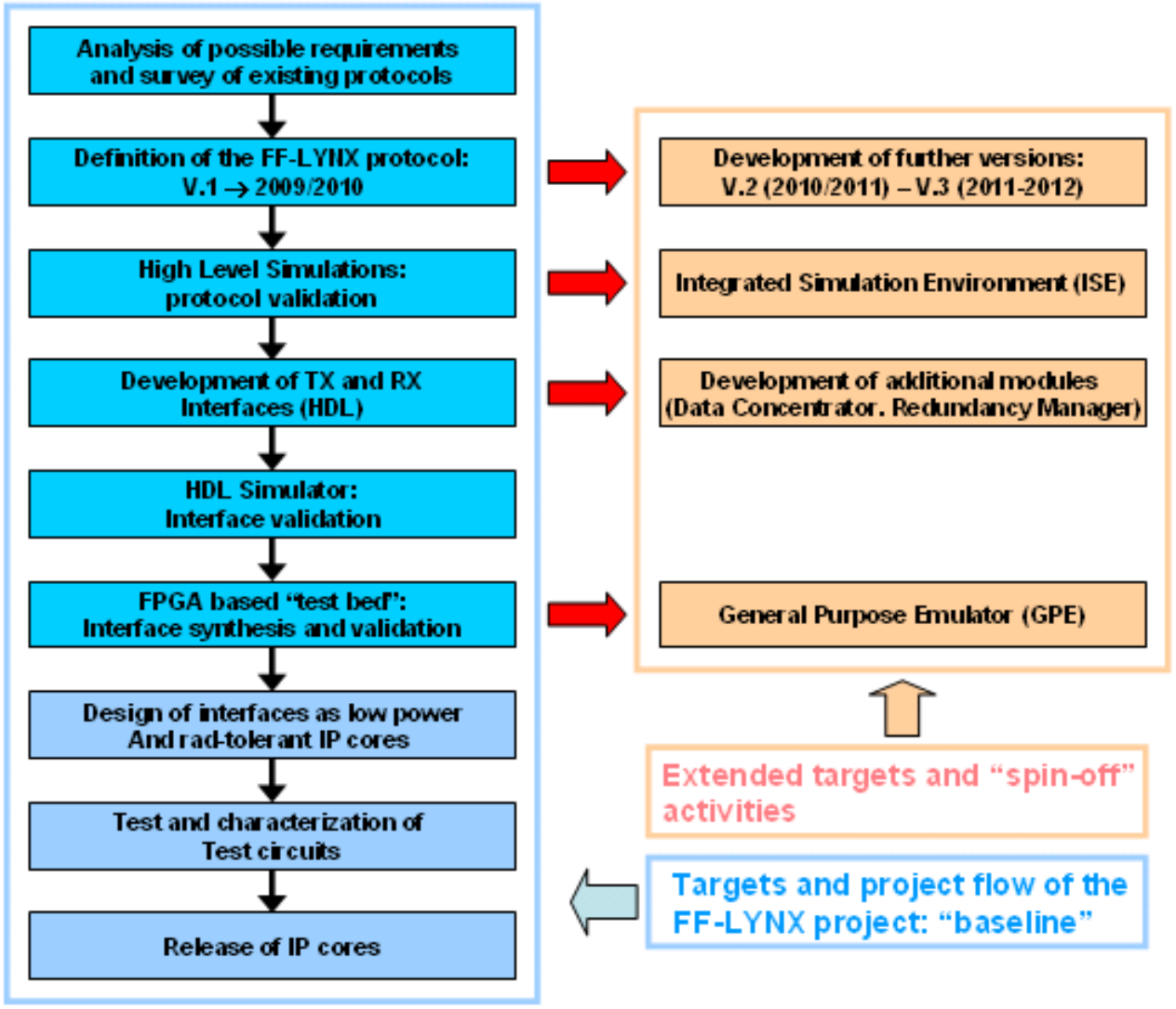


Conclusions

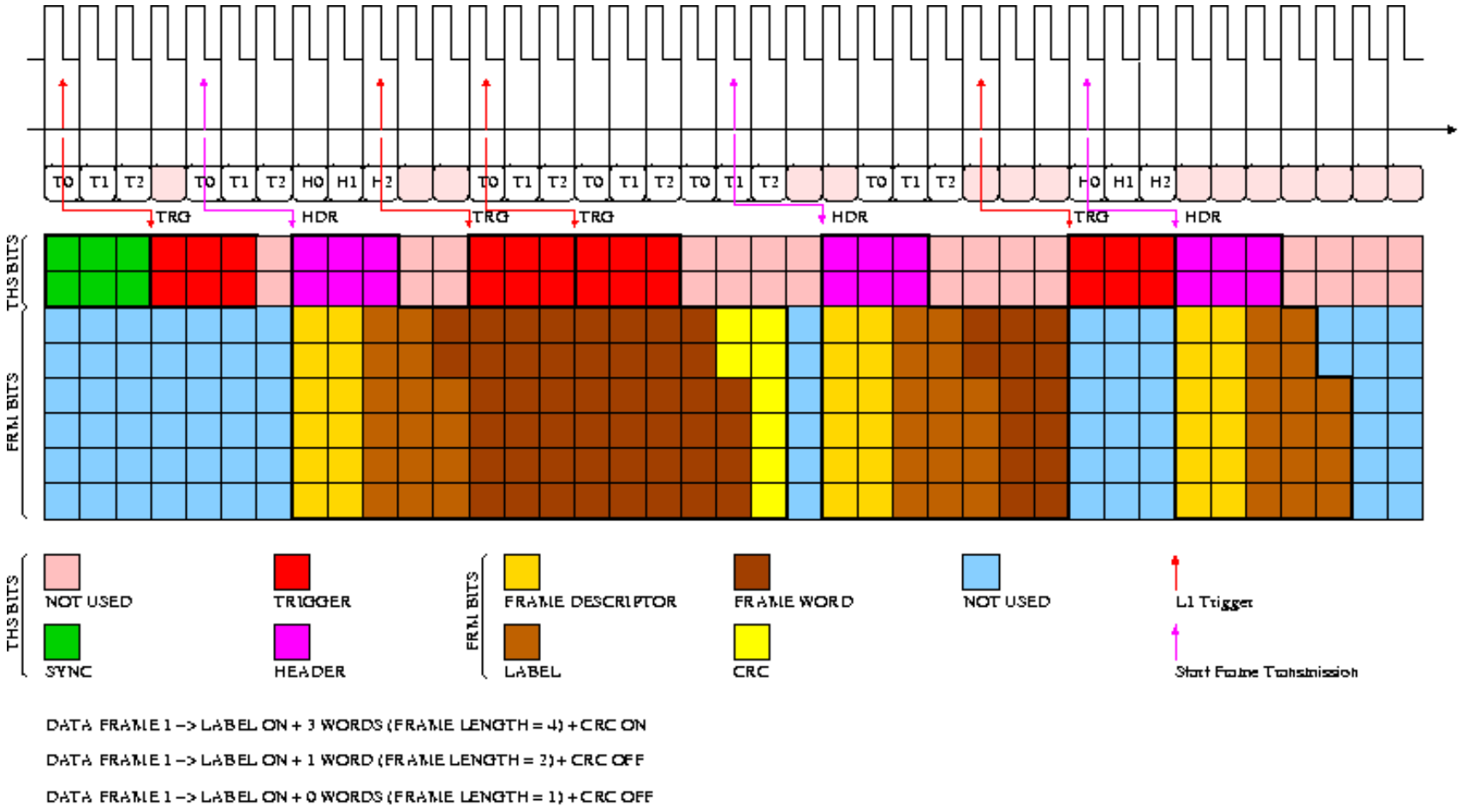
- A flexible protocol for the integrated distribution of TTC signals and the data acquisition has been defined and validated through high level simulations.
- The protocol is tailored on the requirements expected for future HEP experiments and it can support a new communication standard featuring the required functionalities and providing the necessary degree of flexibility with respect to different data formats and system architectures.
- HDL models of TX and RX interfaces implementing the protocol have been developed and functionally validated through HDL simulations and in an FPGA based emulator.
- The design of a test circuit with prototypes of the interfaces implemented with solutions at the architectural, circuital and layout level that will ensure the required radiation hardness and power dissipation is planned in Q2/Q3 2010
- This activity will bring to the development of reusable and configurable intellectual property (IP) macro-cells implementing the designed interfaces, available to the designers of ICs for HEP experiments.

Backup slides

FF-LYNX: project flow



Down-Link (to detectors)



DATA FRAME 1 –> LABEL ON + 3 WORDS (FRAME LENGTH = 4) + CRC ON

DATA FRAME 1 –> LABEL ON + 1 WORD (FRAME LENGTH = 2) + CRC OFF

DATA FRAME 1 –> LABEL ON + 0 WORDS (FRAME LENGTH = 1) + CRC OFF

Up-Link (from detectors)