Progress on Pixel Region Optimization and SystemVerilog Simulation

E. Conti, S. Marconi, P. Placidi
DIEI – University and INFN Perugia (Italy)

J. Christiansen
CERN

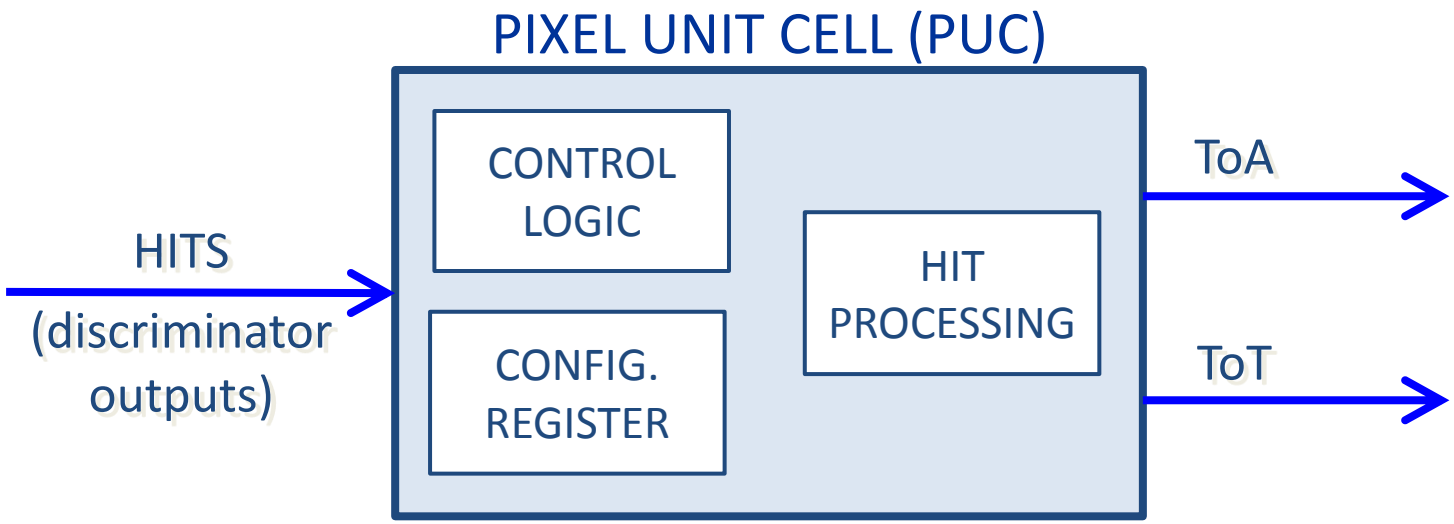
elia.conti@diei.unipg.it

Current Status

- New data available coming from MC simulation (M. Swartz) related to cluster footprints
  - statistical analysis on pixel grouping will be carried out from new data using the analytical approach developed in 2012
- A triggered high level pixel model with basic functionality is currently being developed using SystemVerilog
  - preliminary version of buffering and triggering logic has been implemented
  - module hierarchy has been instantiated (pixel unit cell, pixel chip and pixel region): easily extendable from 1 to NxM pixels
- Work to be validated
  - simulate pixel grouping using a devoted verification environment for a high level pixel chip model (SystemVerilog/UVM)

Implemented High Level Pixel Chip Model
Pixel Unit Cell (PUC)

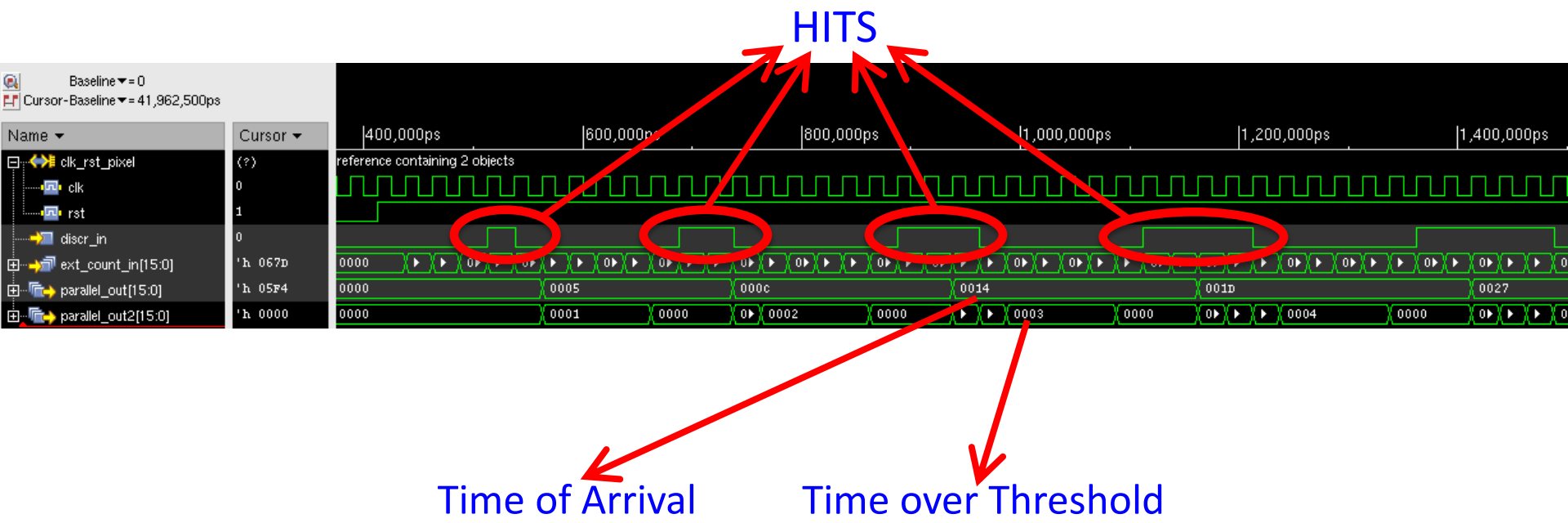
- Basic functionalities implemented in the pixel
  - Hit processing (ToA, ToT)
  - Pixel masking (basic configuration)



Implemented High Level Pixel Chip Model PUC subsystem (1)

HIT PROCESSING LOGIC

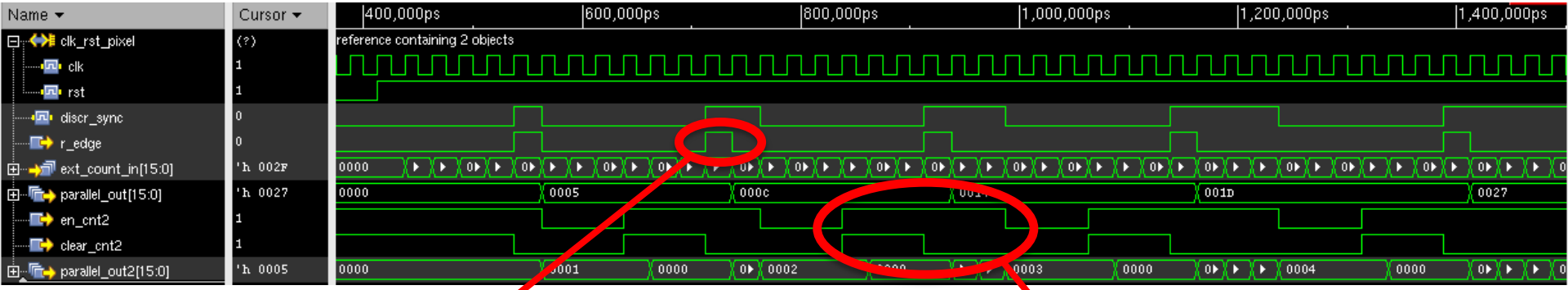
- ToA is determined by copying in a register an externally counted value related to the bunch crossing clock cycle
- ToT is carried out by a devoted counter



Implemented High Level Pixel Chip Model PUC subsystem (2)

CONTROL LOGIC

- Finite state machine providing control signals for hit processing logic



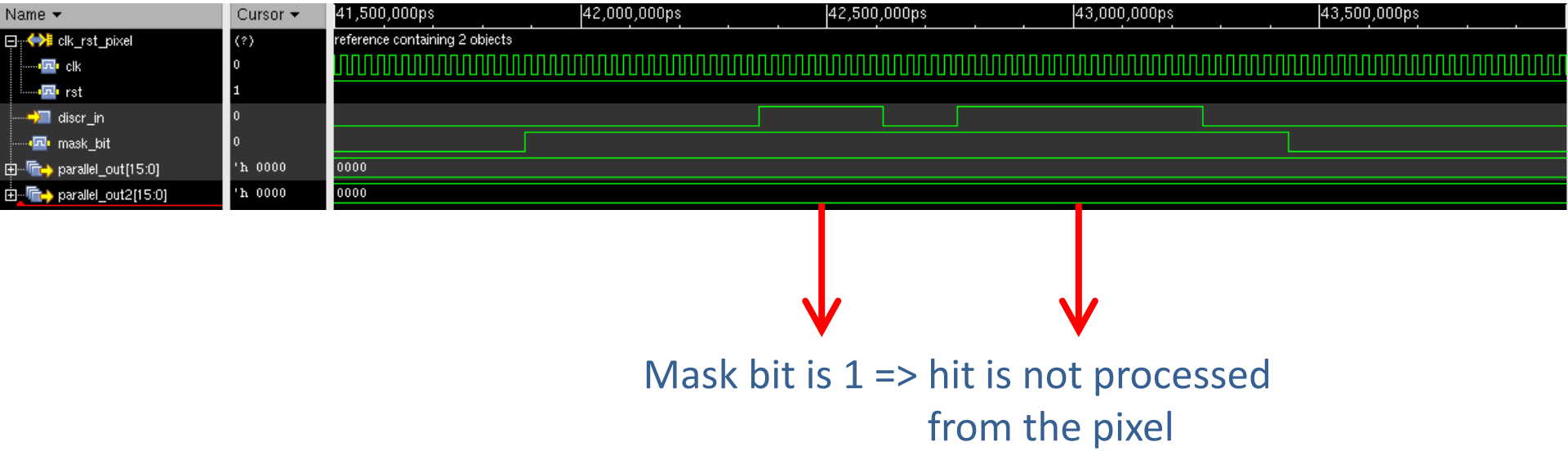
Control signal for ToA value copy

Enable and clear signals for ToT counter

Implemented High Level Pixel Chip Model PUC subsystem (3)

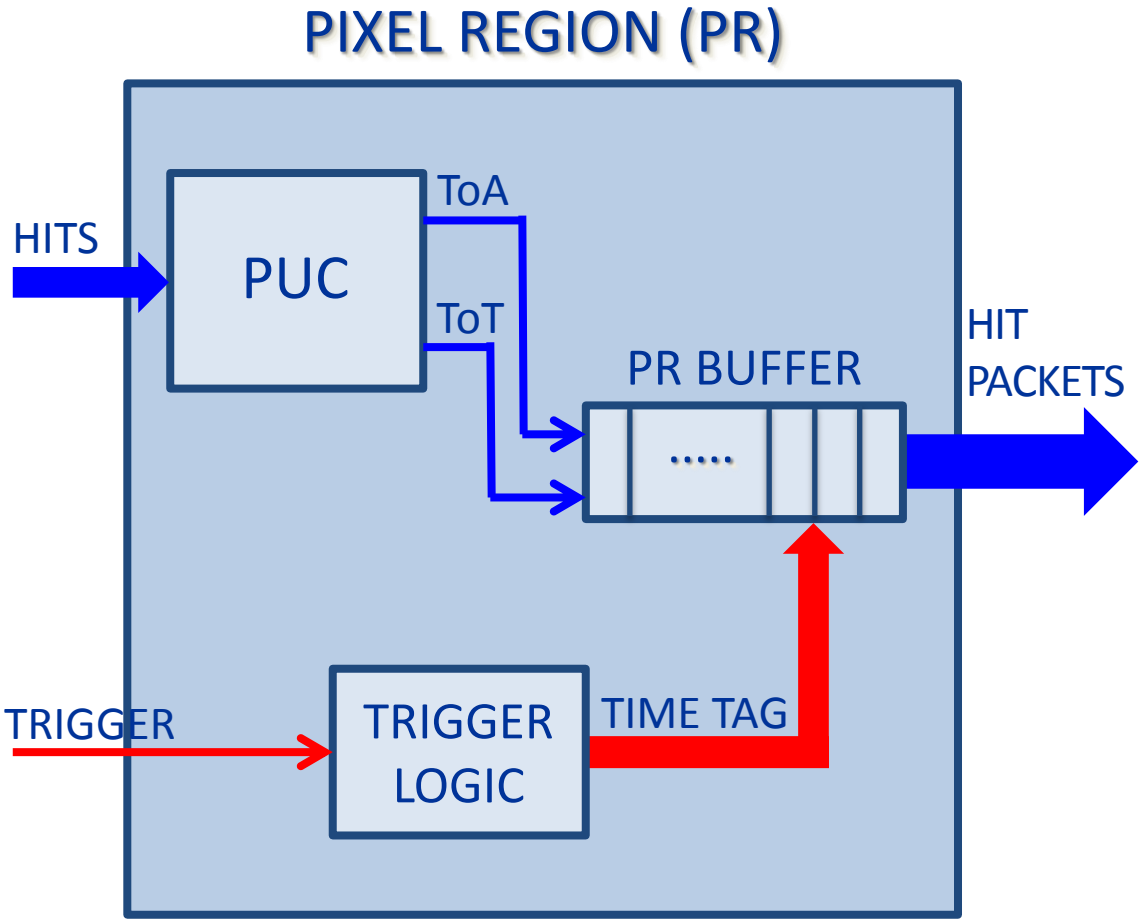
CONFIGURATION REGISTER

- Holds configuration bits – not important at this stage of pixel design.
- We have only included support for pixel mask bit as a basic functionality



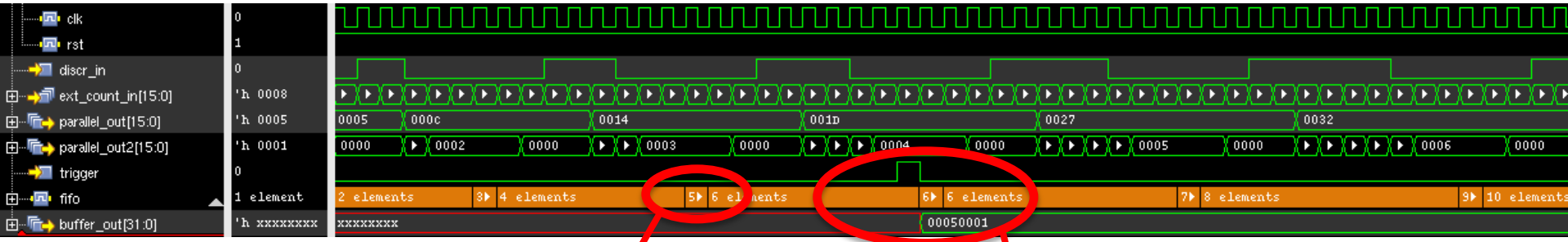
Mask bit is 1 => hit is not processed from the pixel

Implemented High Level Pixel Chip Model
Pixel Region (PR) (1)



- Buffer depth (D) and trigger latency (L) are programmable
- PR buffer is modeled as a FIFO (SystemVerilog queue) where ToA + L is compared to the time tag coming from the trigger

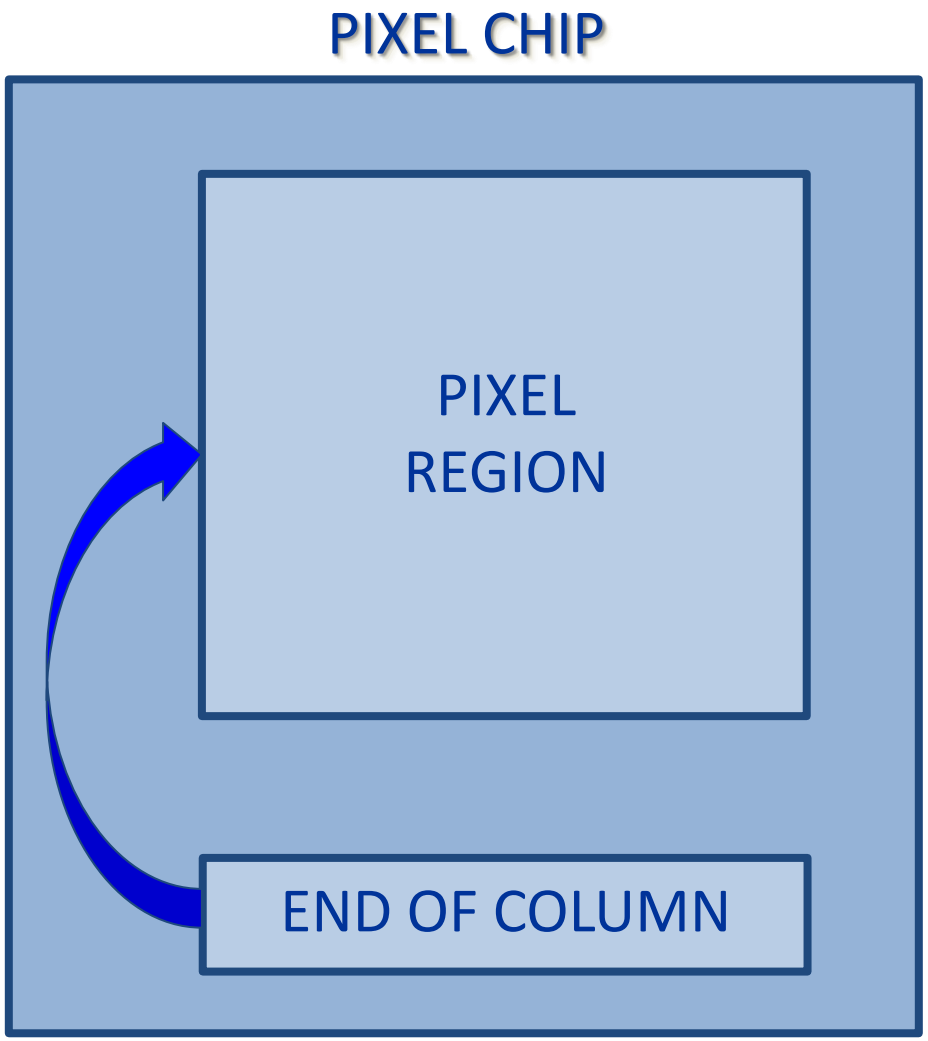
Implemented High Level Pixel Chip Model Pixel Region (PR) (2)



Hit packets (ToA + ToT) are stored in the buffer (still some details to be fixed)

Trigger selects data from the buffer

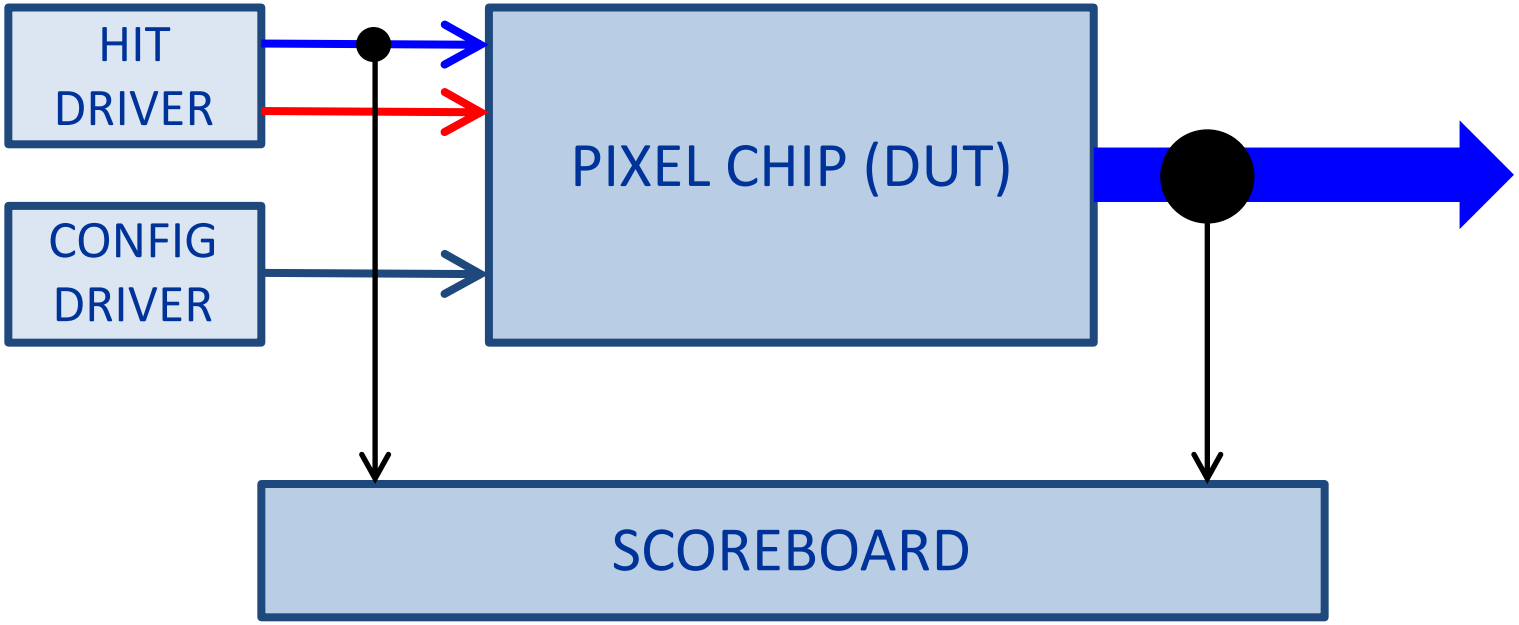
Implemented High Level Pixel Chip Model
Pixel Chip



- Pixel chip contains
  1 pixel region
- Pixel region contains
  1 pixel unit cell (PUC)
- EOC provides external counter for determining ToA
- With such hierarchy it is possible to easily extend to pixel regions made by NxM pixels and pixel chips containing multiple regions

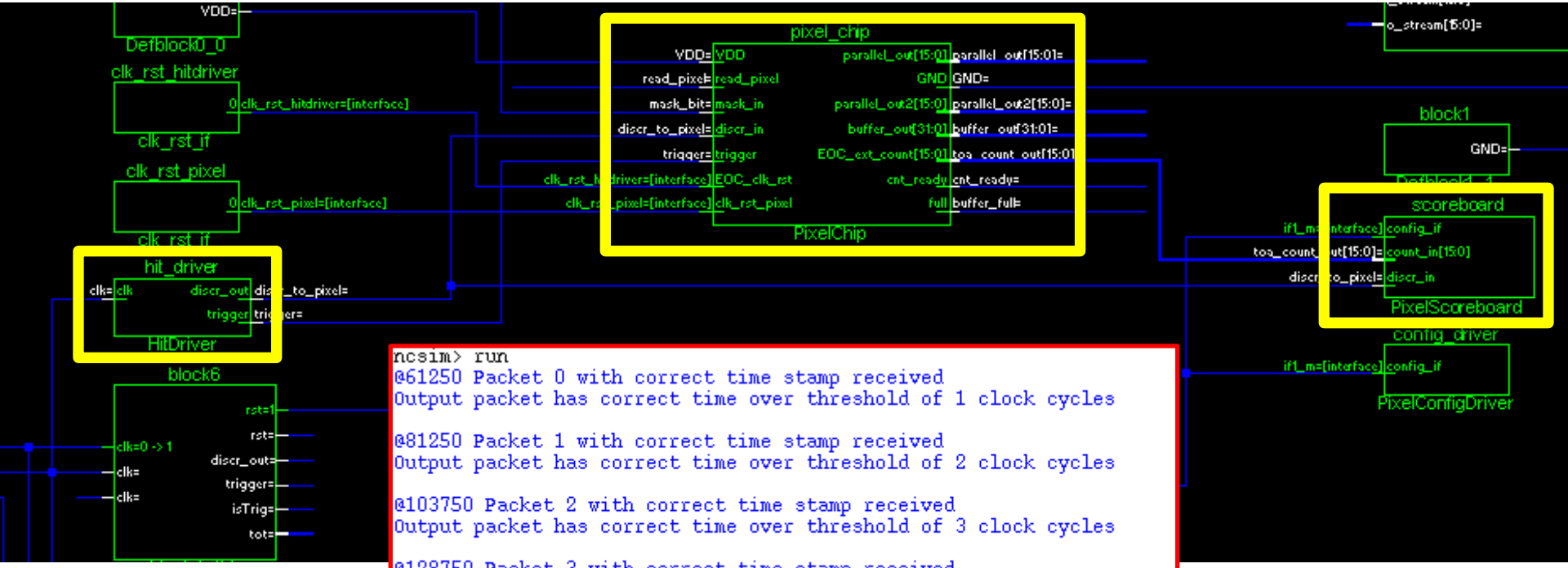
Implemented High Level Pixel Chip Model Verification Environment

- Custom module-based testbench with drivers and scoreboard



- **DRIVER**: Arbitrary hit and trigger generation
- **SCOREBOARD**: Conformity checks between input and output hits (checks not involving buffering/triggering yet)

Implemented High Level Pixel Chip Model Example of Simulation

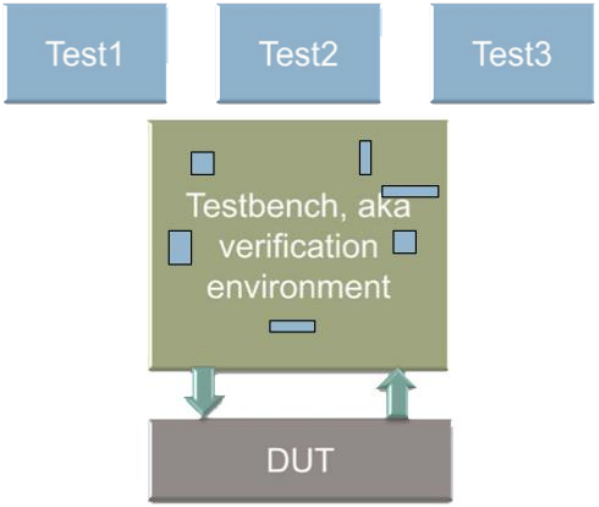



```
ncsim> run
@61250 Packet 0 with correct time stamp received
Output packet has correct time over threshold of 1 clock cycles

@81250 Packet 1 with correct time stamp received
Output packet has correct time over threshold of 2 clock cycles

@103750 Packet 2 with correct time stamp received
Output packet has correct time over threshold of 3 clock cycles

@128750 Packet 3 with correct time stamp received
Output packet has correct time over threshold of 4 clock cycles
```

Console output

Further Developments

- Standardize verification environment into class-based SystemVerilog/UVM for pixel chip containing a single pixel



  - devoted hit driver for automated generation of constrained random stimuli and injection of data from MC simulations
  - coverage based simulation
  - reusability during pixel design flow

- Replicate PUCs and start simulating pixel region configurations