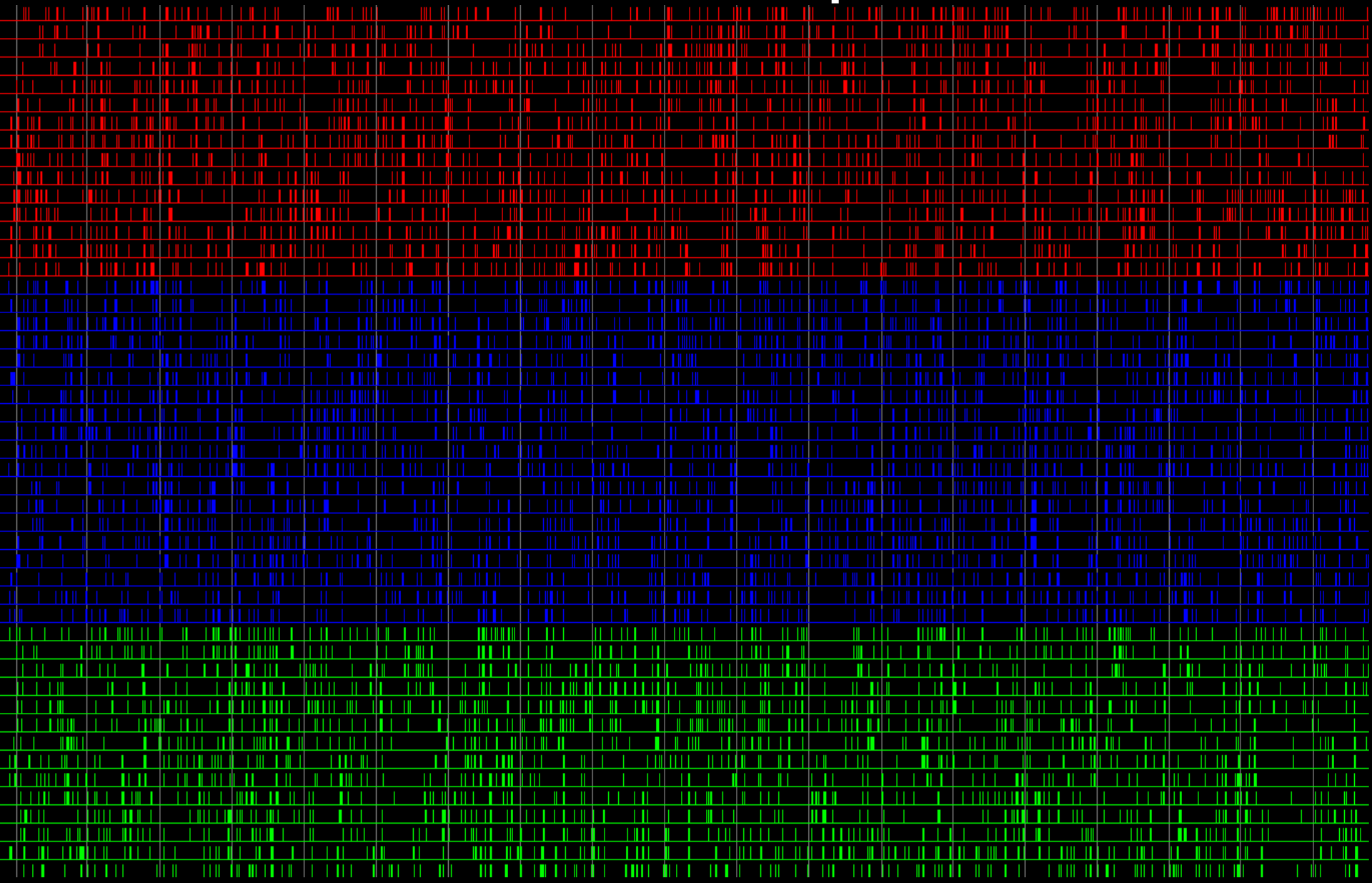
Simulation updates



12-11-2010 F.Giorgi

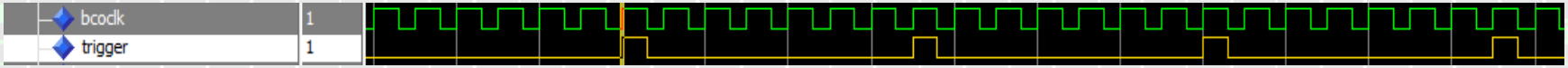
What's new

- Architecture
  - New Barrel1 and Barrel2 optimized for synthesis compilation time and number of cells
  - New Sweeper featuring:
    - Optimized sweeping algorithm (compilation and cells)
    - Data Push/Triggered selectable working mode, selectable trigger latency
  - BC and trigger signals: doubly registered on input (RDclk)

- Test bench
  - More information stored after each run (logs, reports, raw data, waveforms, check results… → optimized sim/human-check cycle: night-long simulations/day analysis)
  - New cross check algorithms (python script to be converted and integrated with previous C++ compiled routines).
    - TS order check
    - Search for triggered hits not found on output
    - Search for not-triggered hits found on output

Trigger working mode

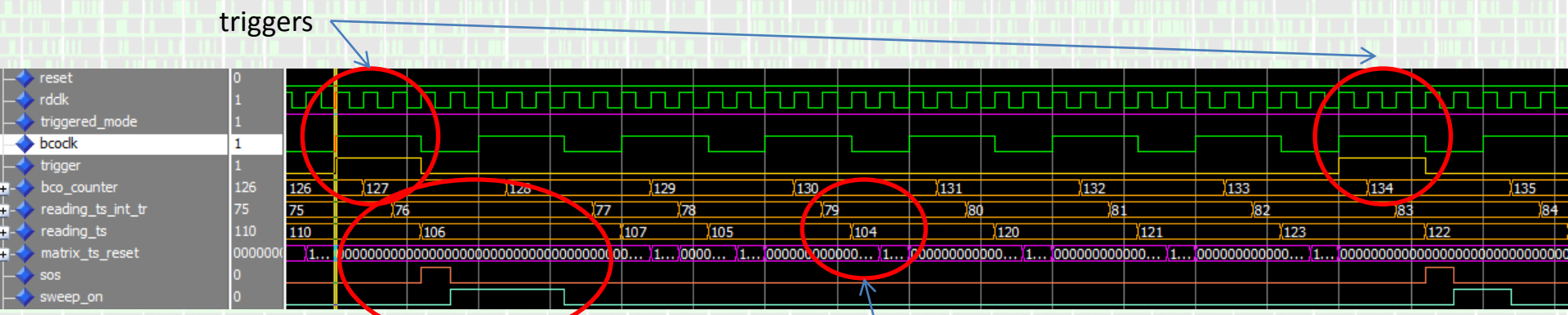
- Trigger interface fixed in agreement with M. Villa
- Machine trigger processed by DAQ boards, and “translated” to FE chips
  - Trigger logic as simple as possible on chip
  - Programmable trigger logic on DAQ boards
- Trigger signal aligned with BC clock:



- Trigger latency selectable via I2C slow control
  - 0 -255 value range (Time counter range)
  - units of BC periods
- Triggered TS → corresponding hits are swept out of the matrix
- Not triggered TS → corresponding hits are deleted on the matrix

Trigger working mode

Waveform example



- BC period 100 ns
- Trigger latency 50 BC (5 us)
- Trigger frequency: 1 trigger every 7 BC (1.4 MHz) (random trigger implemented also)

Tests

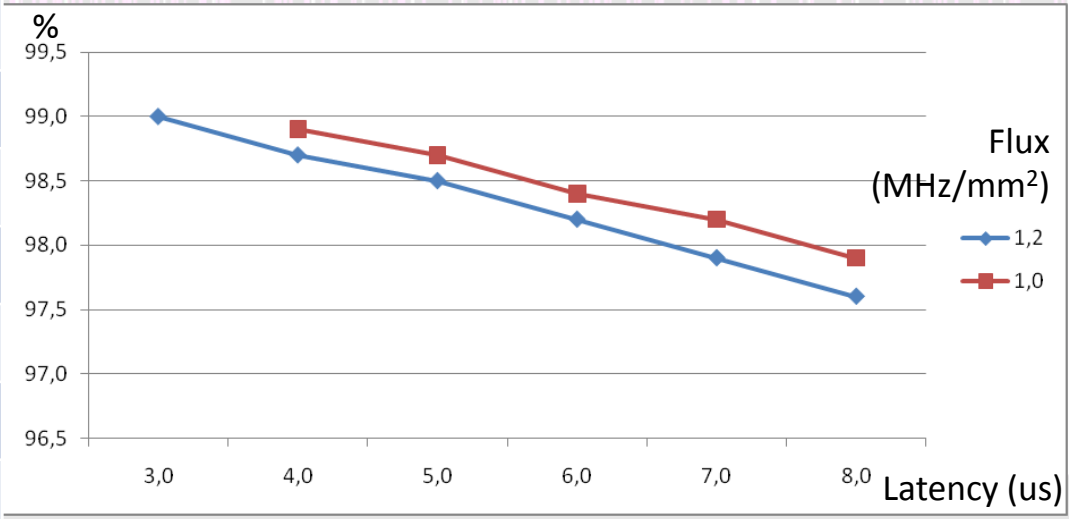
- 1st test: does what used to work (Data Push) still work fine?? → hits check passed
- Many short functional tests (200 us → ~20k hit each)
  - Many latencies
  - Random trigger and modulo trigger
  - Latency=0
  - Latency=0 ,Trigger "all-in" → data push, sort of.
  - → hits check and waveform inspection passed
- Long runs for efficiency estimation
  - 2 latency scans performed in the 4 → 8 us range (1.2 MHz/mm^2 and 1.0 MHz/mm^2)
  - 3 ms simulated for each run (~half million events each)
  - hits check passed

Triggered mode efficiency results

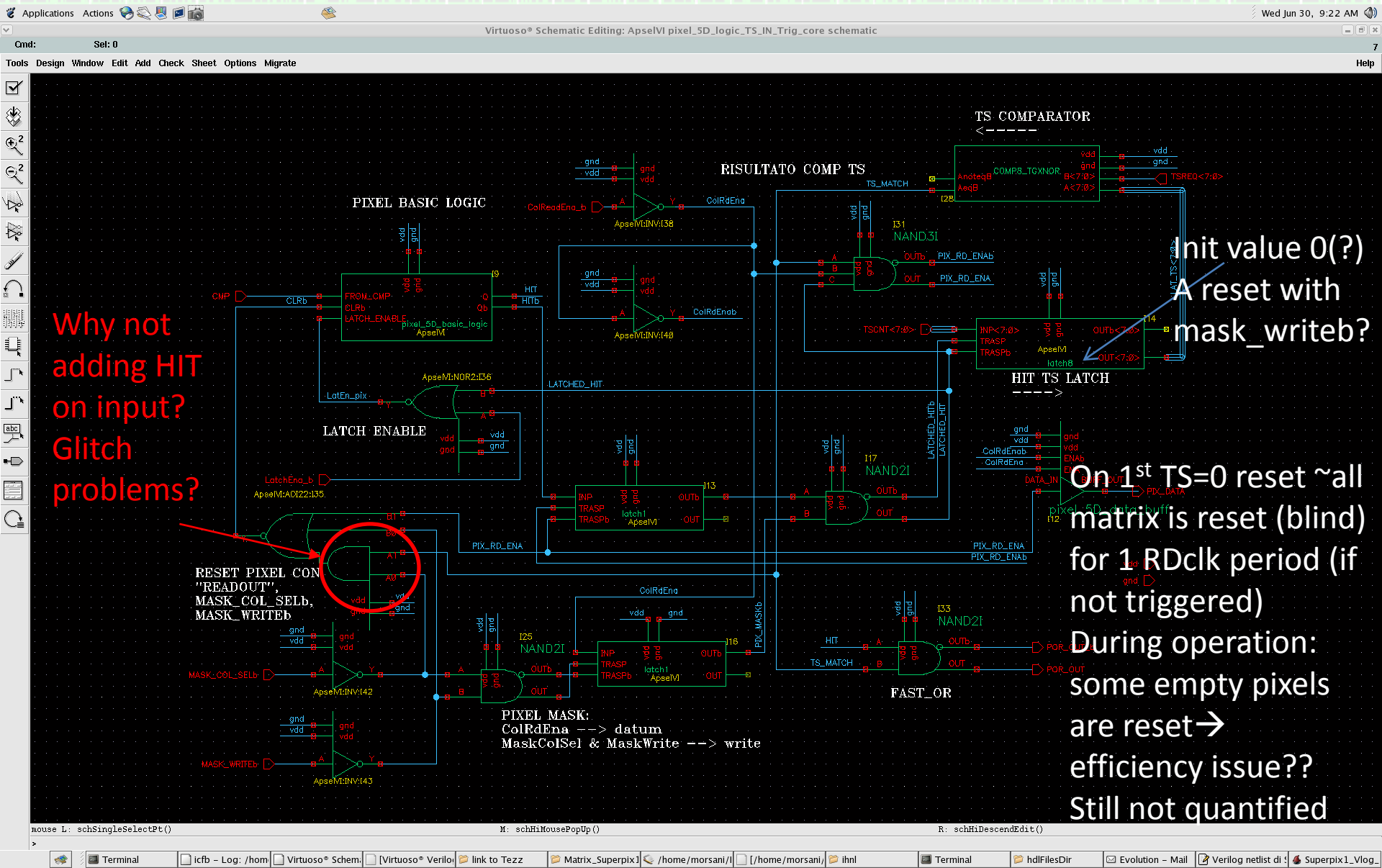
RUN	Lat (us)	Flux(MHz/mm^2)	efficiency
453	3,0	1,2	99,0
468	4,0	1,2	98,7
469	5,0	1,2	98,5
470	6,0	1,2	98,2
471	7,0	1,2	97,9
472	8,0	1,2	97,6
496	4,0	1,0	98.9
497	5,0	1,0	98.7
498	6,0	1,0	98.4
499	7,0	1,0	98.2
500	8,0	1,0	97.9

BC 100 ns
Trigger rate 2,5 MHz
RDclk 50 MHz
3 ms each run

hits check passed



Matrix behavior on first TS=0 reset



Conclusions

Many thanks to student **F. Conti** (code optimization, triggered mode, check scripts)

- What did we achieve?
  - Enforced test bench
  - Triggered mode implemented and preliminary tested
  - Preliminary measurements of efficiency carried out
  - Barrel and sweeper code optimizations tested

- Is the efficiency acceptable at the expected latency of 6 us?
- Is the pixel TS-dependent reset a real issue?
- Close to come:
  - Architecture tailoring on submission dimensions
    - Code branching node → as late as possible
    - Mandatory for real matrix model test → as soon as possible

AARGH!

  - Synthesis on Synopsys DC.