Test benches / custom hardware status for the review

- SLINK64
  - Merger, Cables, Fedkit
- FRL
  - Measurements
- RU bench
  - Configurations, Measurements
- FMM
- TTCrx

Review Topics

- Overview on column components and test benches
- Summary of last years status to emphasize Δ
- Introduce GIII : the base of ongoing tests
  - Explain the Fedkit and the relevance for CMS (number of ordered/sold modules)

Discuss also today:

- SLINK64 bench, FRL bench, RU bench status
- FMM, TTCrx status

Glossary

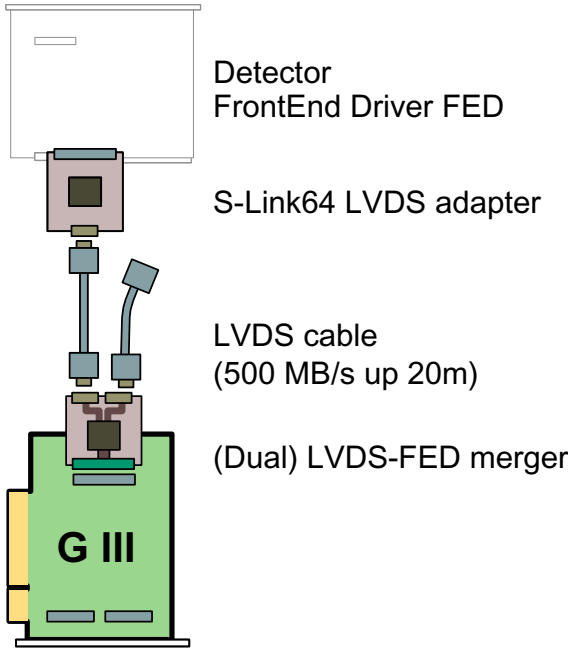
- PROTOTYPE : aims to meet final requirements
- DEMONSTRATOR: study to demonstrate feasability with todays means. Might need some more development to meet the final requirements

GIII / SLINK64 - production

Board	Version	Quantity	Use
Generic III (old)	First version	10 pieces	All
Generic III	Actual version	30 pieces 32 pieces	10 INFN 5 FED_KIT LVDS 4 for debug new design 5 user fed-kit 6 free New production (January 2003)
LVDS link	First version	2 kit	1 cern 1 Ohio J.Gilmore ENDCAP muon
LVDS link	Actual version (with merge)	5 kit 18 kit	For FED-KIT LVDS New production (January 2003)

SLINK64 prototype and test bench

- New :
  - Full SLINK64 constructed: intelligence in sender and receiver needed to meet specification.
    - Test-mode,
    - Resynchronization of flow control signals
  - Still: Too tight specification for FED developers to handle flow-control
    - trivial to change
  - Merger : Add on with FPGA and Fifos to receive up to two SLINK64 data channels
    - Today: Plugs on GIII which emulates FRL
    - Future: Plugs on FRL or is implented in FRL
  - Cable tests
    - A variety of tests with cables of 4 different vendors



Producer	Length	Test	Test length	Frequency	Remarks
AMP	2 meters	OK	10^{15} (1 month)	100 MHz x 64-bit	
AMP	7,5 meters	OK	1 day	100 MHz x 64-bit	
AMP	(2 + 7,5 + 7,5)=17 meters	OK	1 day	66 MHz x 64-bit	
Amphenol	20 meters	KO (wrong bits)		66 MHz and 33MHz	
Amphenol	15 meters	OK	70 x 10^{12}	66 MHz x 64 bits	
3M	15 meters	OK	Short test	66 MHz x 64 bits	
3M	10 meters	OK	Short test	66 MHz x 64 bits	
AMP	2 x 7,5 meters = 15 meters	OK	Short test	66MHz x 64-bit	
Amphenol	20 meters/15meters Universal SCSI	KO		66MHz and 33 MHz	Doesn't follow our specifications
Israel	20 meters	KO		66MHz and 33MHz	We haven't any specifications for this cable

RESULT :

- An SLINK64 prototype has been successfully built.
- The Specifications of SLNK64 are fully met
- The data throughput of **528 MB/s** (66MHz / 64 bit) has been achieved with a long cable **(15m)**

Current Activity : Fedkit with SLINK64

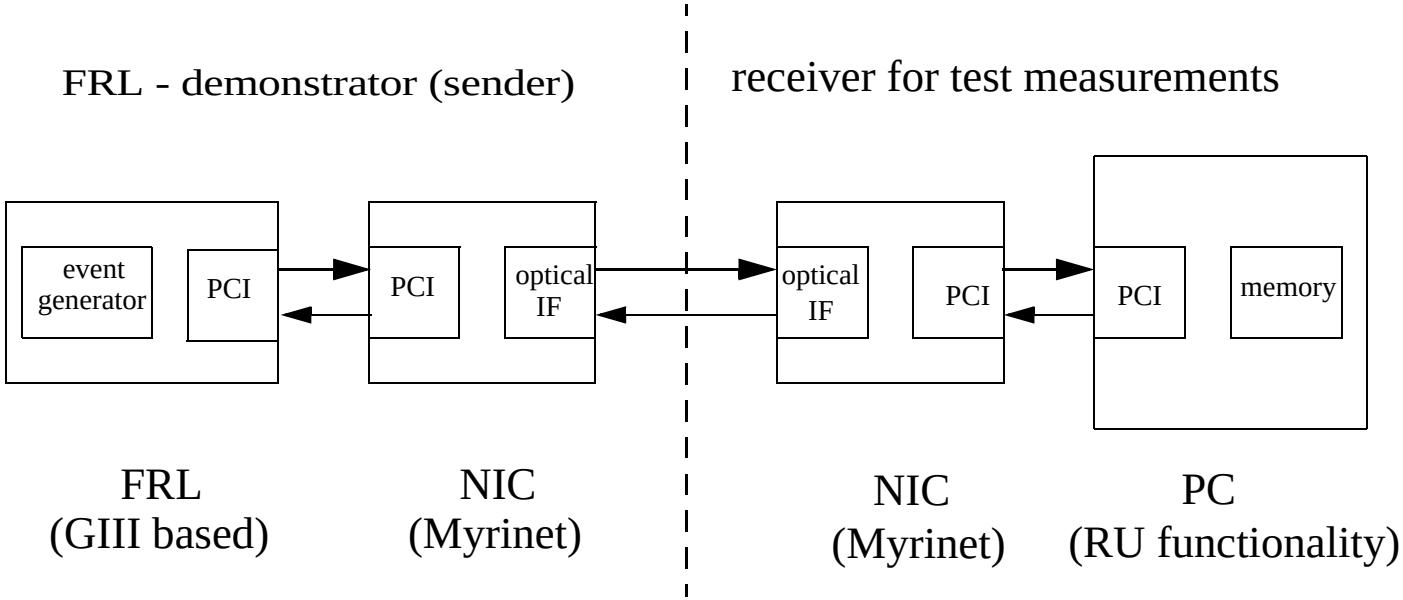
Three modes of operation

- Event Generator (working)
  - One GIII plugged into PC; event parameters are written into GIII by host PC
  - Full SLINK, sender card contains event generator.
- Master Mode (in progress)
  - Full Slink (sender and receiver)
  - Sender is GIII card and initiates DMA to get data from PCI bus
- Slave Mode (in progress)
  - Full Slink (sender and receiver)
  - Sender is GIII and external device initiates DMA into GIII

Tracker wants to use Slave Mode in test beam

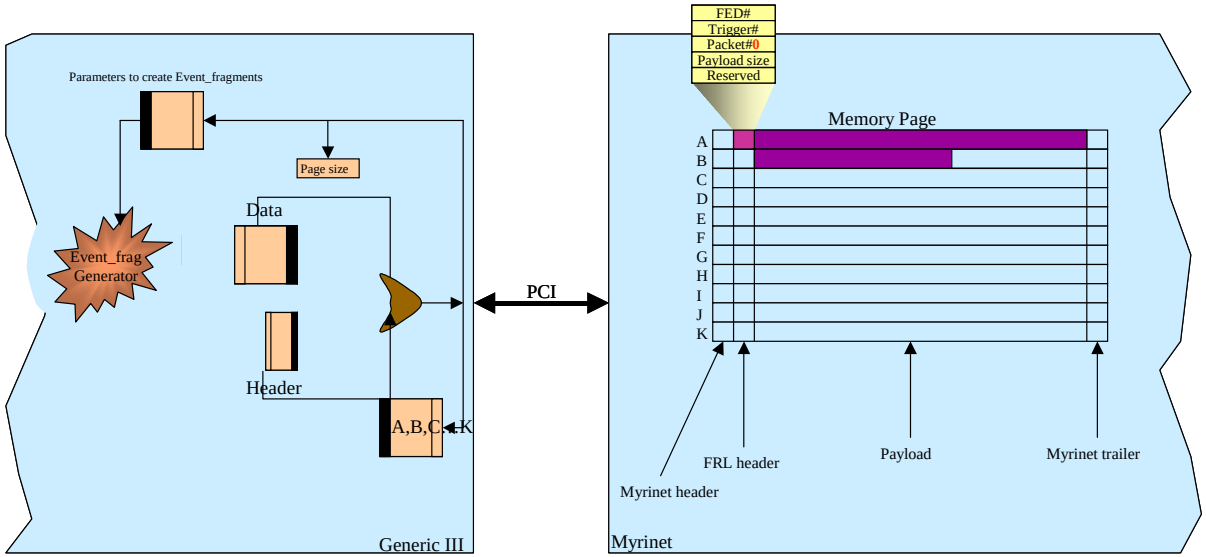
FRL Demonstrator

- Testbench Architecture

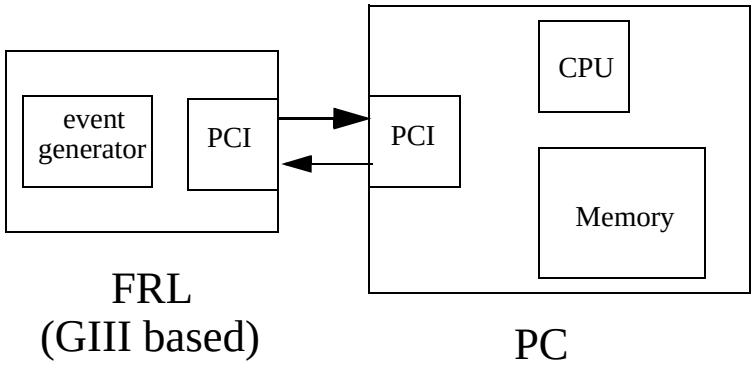


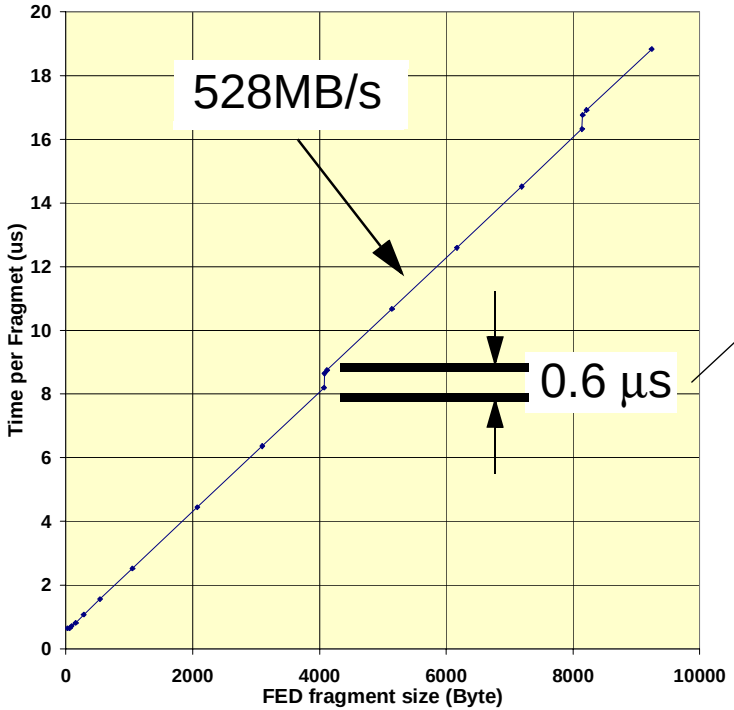
- Hybrid out of GIII and NIC
- Test bench for GIII - NIC protocol --> no real data input
- Data input (Merger) has been tested as receiving side of SLINK64

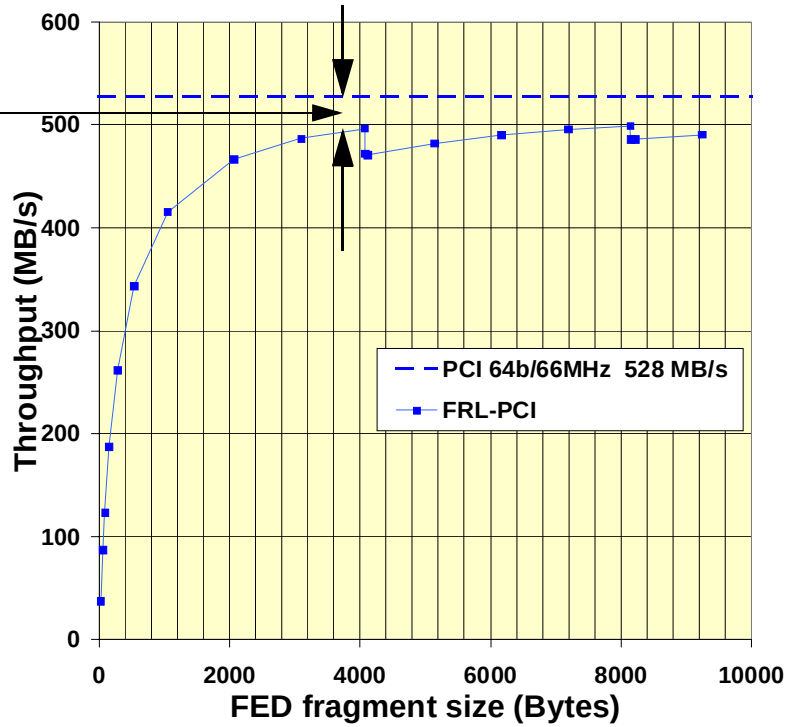
- Protocol
  - Myrinet card AND GIII only support DMA bursts
  - prefer DMAs and poll local memory than single word accesses
  - Protocol features:
    - Buffer loaning avoids unnecessary copying
    - chains of fixed size buffers (fragment sizes are not known when fragment arrives in FRL)
  - Protocol features:
    - Buffer loaning avoids unnecessary copying
    - chains of fixed size buffers (fragment sizes are not known when fragment arrives in FRL)
    - Header space for FRL and Myrinet



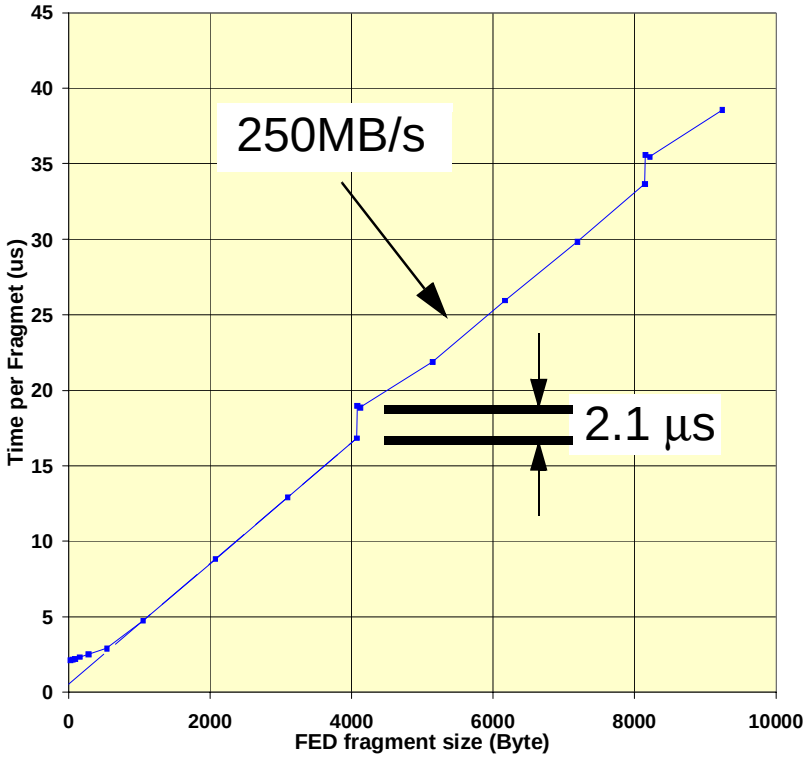
- Test 1 : PC emulates NIC card
  - Measures performance of FRL (PC is much faster than NIC)
  - details: 4kB pages, 66Mhz 64 bit PCI-bus, 1024 page entries in GIII, fragment parameters for 512 fragments in GIII fifo by PC

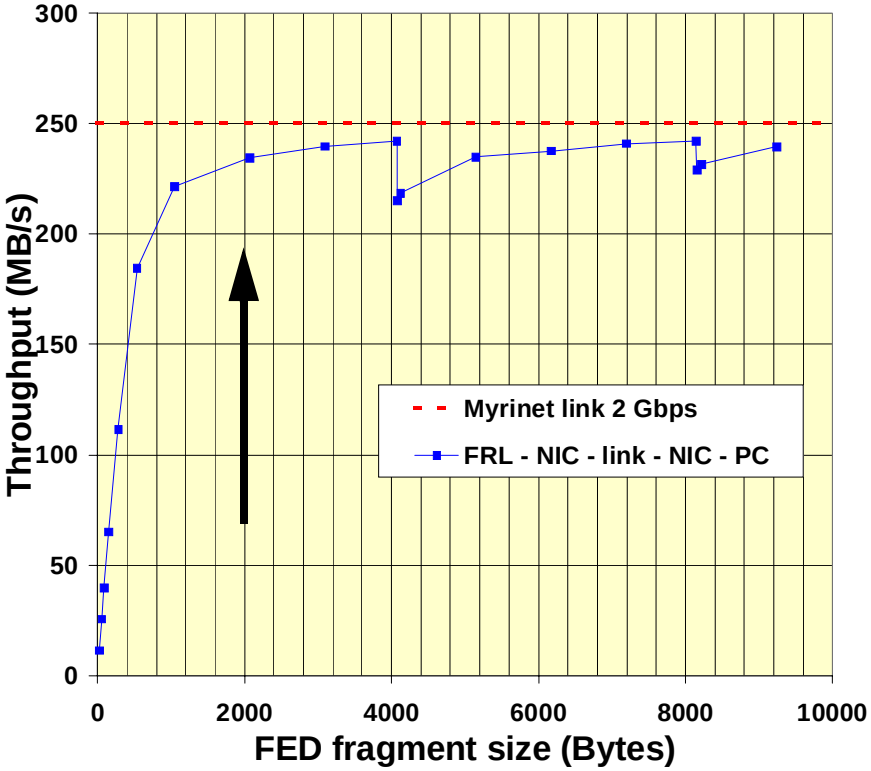






- Test 2 : FRL - NIC - NIC - PC
  - theoretical maximum now 250 MB/s (Myrinet)
  - offset 2.1 µs is partly shadowed (CPU works while DMA goes on)
  - 0.5 µs “irreducible” offset



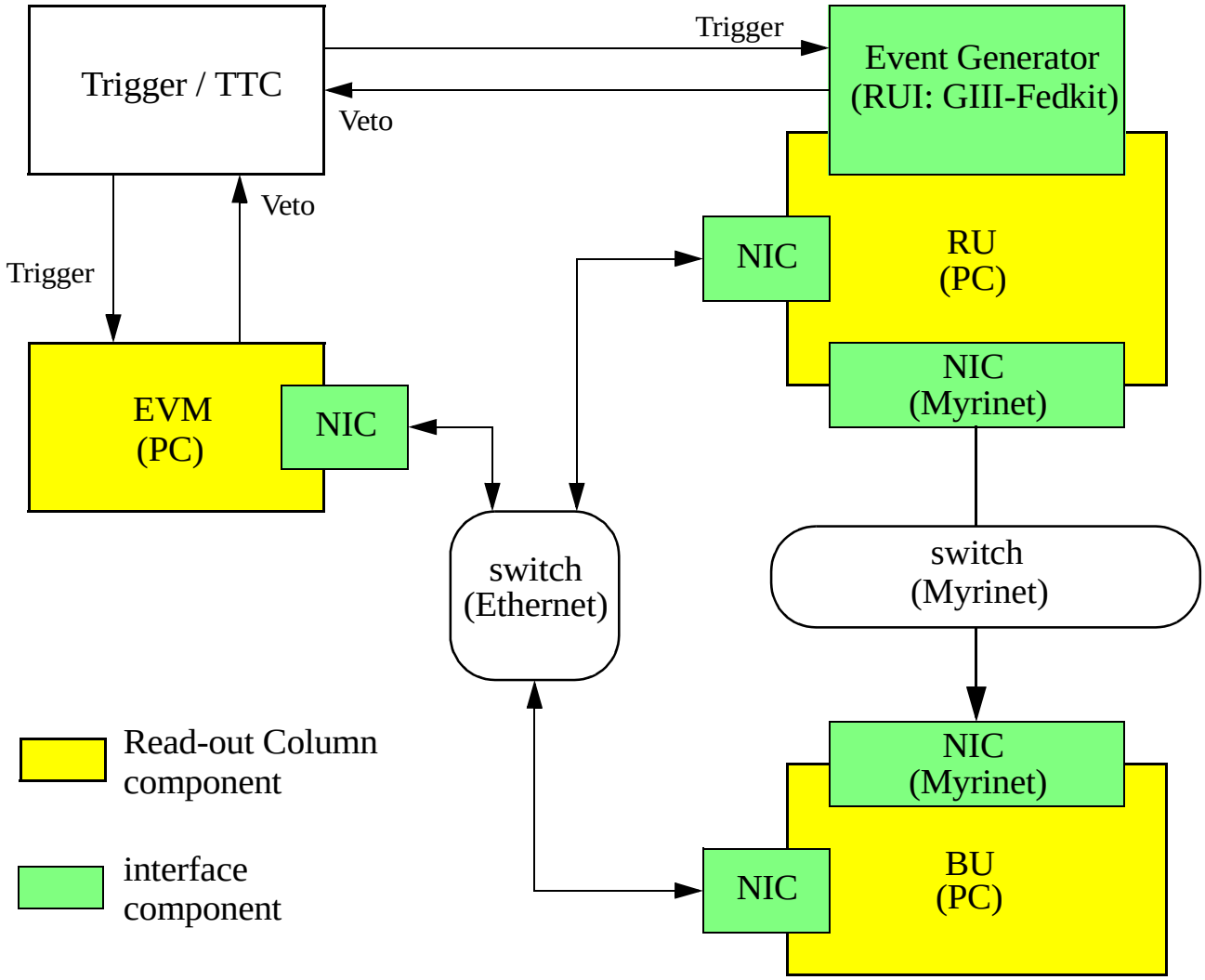


CRUDE : Column for Readout Unit DEvelopment

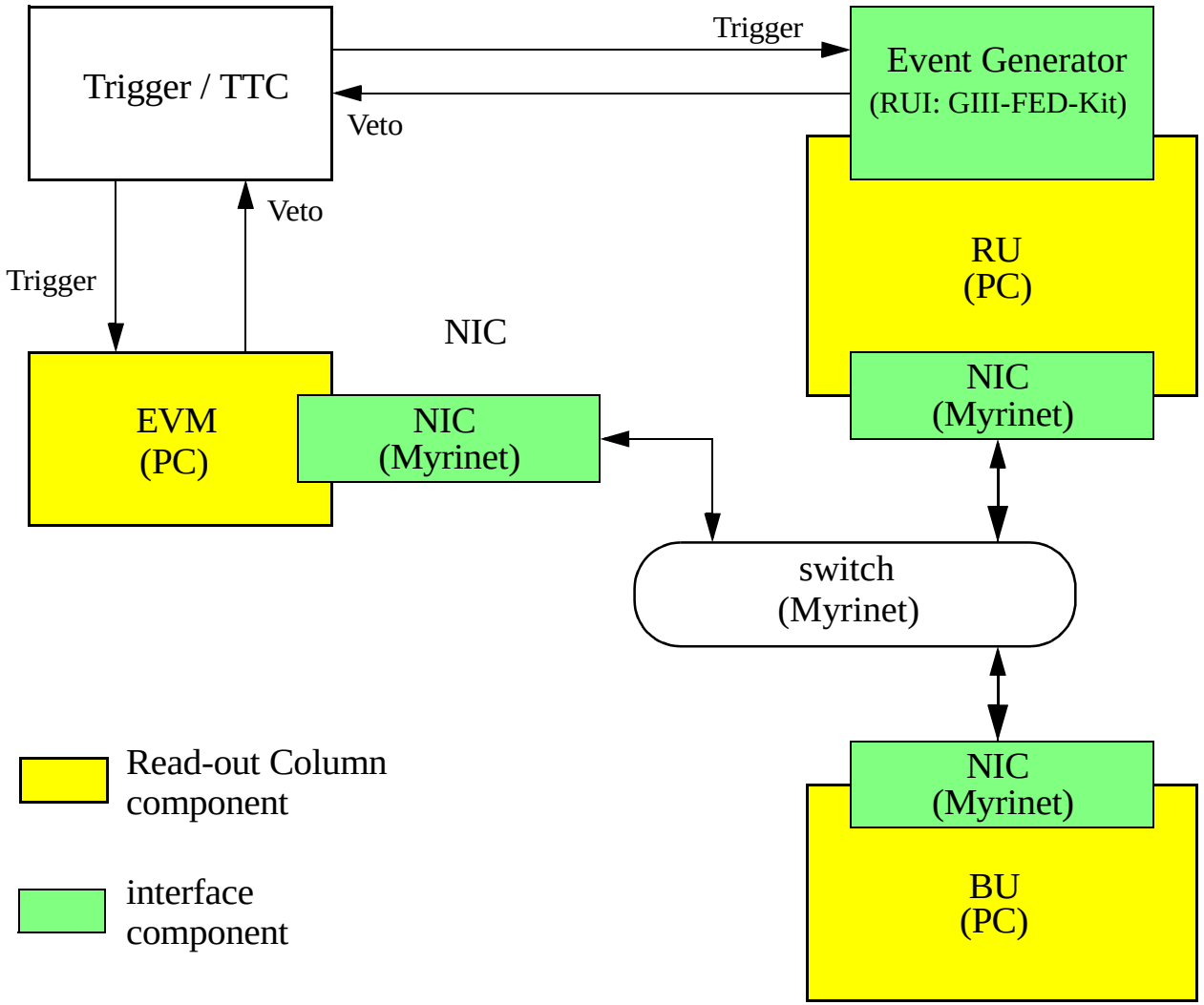
- Test bench features for RU-PC
  - XDAQ software environment
  - XDAQ applications used and modified as needed
    - zero copy implementation using buffer loaning
    - indirect mode
  - Data source (RUI replacement) : modified Fedkit (with trigger and backpressure)
    - works similar to current view of RUI
    - uses buffer loaning scheme with memory allocated by XDAQ (required by GM)
    - data shuffled by DMA of external card
    - backpressure latency compensated for with internal trigger counter
  - Data output : BU
    - Myrinet card
    - Protocol GM (4 kB buffer size)

- Implemented data checks:
  - event numbers (always, in various places)
  - data can be checked in BU (optional; not done if speed is measured)
  - no error recovery tried
  - Myrinet card

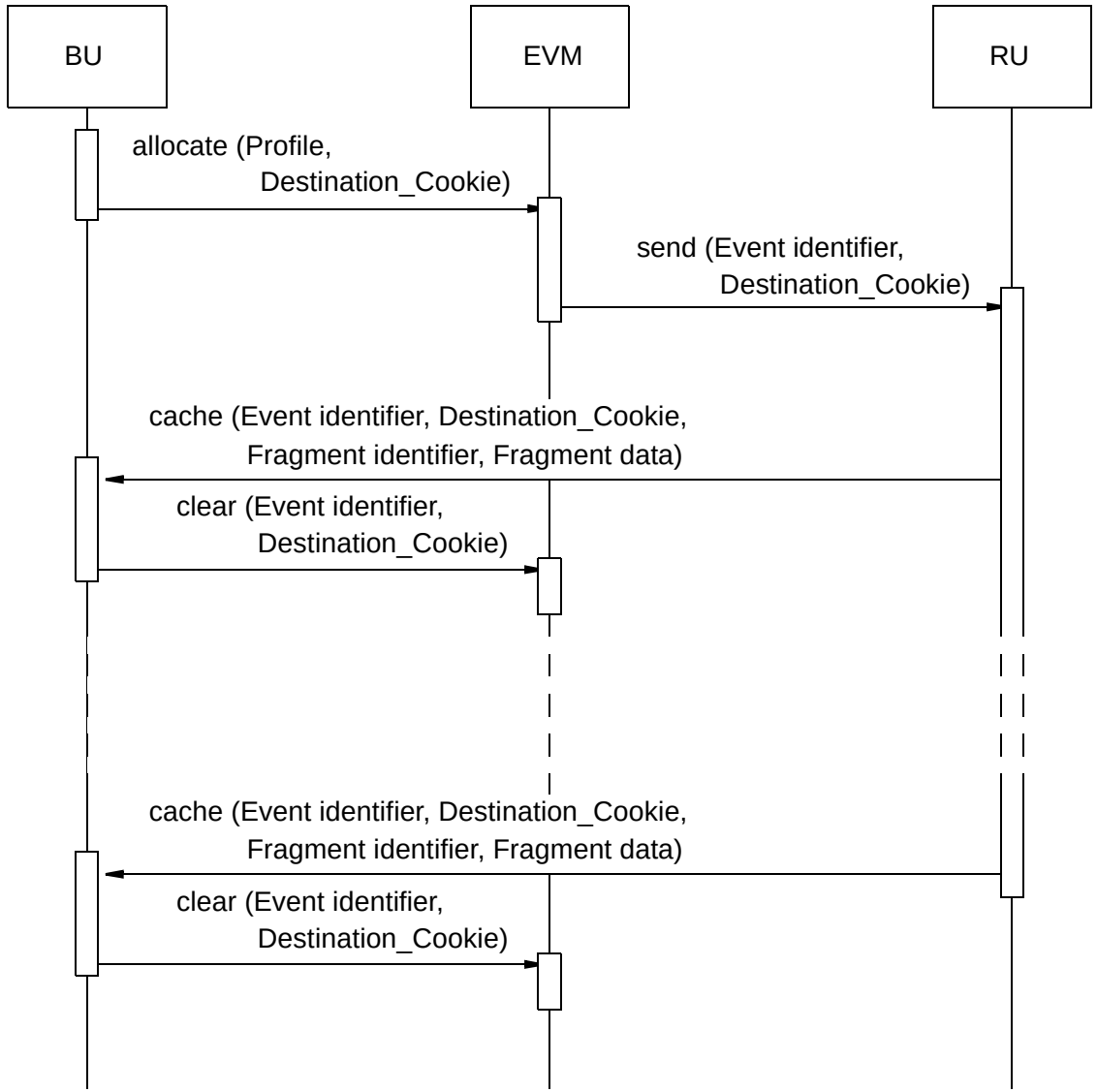
CRUDE Layout 1 :



CRUDE Layout 2 :



EVM : indirect mode



Simplification

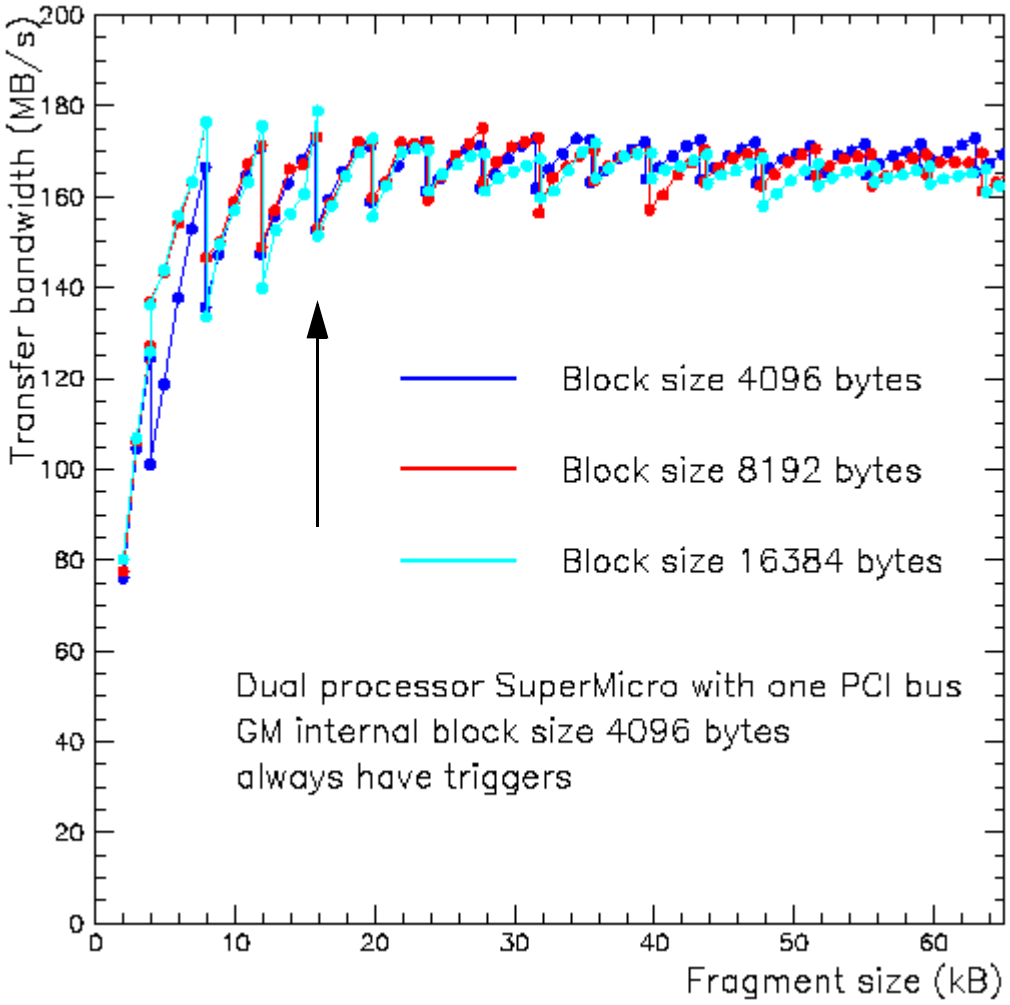
- EVM assumes there are always triggers
  - naive readout of TTCrx setup too slow.
  - need Fedkit like architecture :
    TTCrx readout via DMA: CPU can work in parallel
    Multiple buffers: No backpressure if other tasks are scheduled
- New design is on the way.
  - If done with TTCrx then it needs DMA (not working yet)
  - In GIII it needs a Fedkit interfaced to TTCrx (done)
  - Alternative: Use a slightly modified Fedkit (similar to RUI: with trigger and backpressure)

Possible Parameters to adjust

- General system parameters
  - Bigphys RAM-size
- Fedkit
  - blocksize: these blocks are used in Rui, RU and RUO.
  - maximum number of blocks circulating
  - (mean) fragment size
  - fragment size distribution: constant / table driven
- EVM protocol
  - RU : bundeling of send request,
  - BU : bundeling of requests,
  - Trigger readout: bundeling of readout (CURRENTLY NOT USED (see simplification))

Results : Super Micro differnt block size

- Fixed block sizes
- GM works with fixed blocksize of 4 kB -> sawtooth
- At 16 kB :
  160 MB/s +/- 13 MB/s

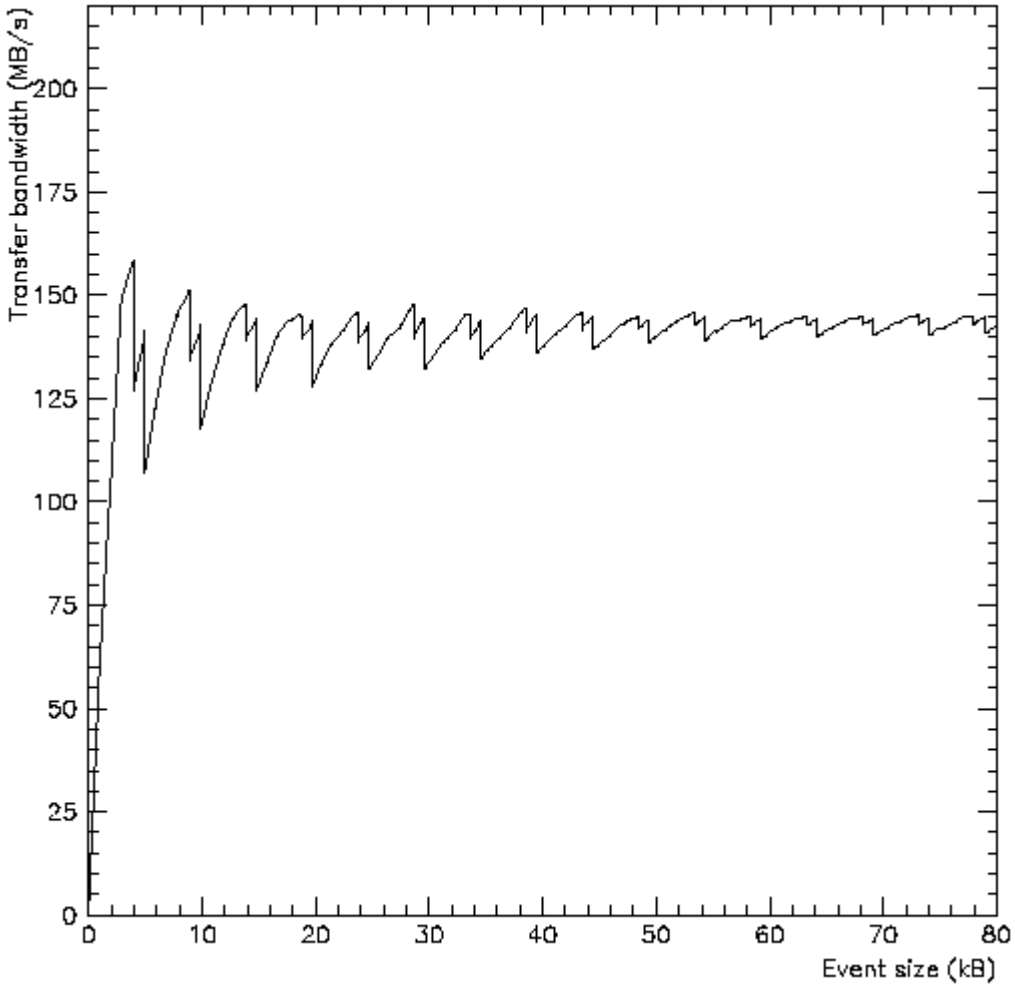


Results : Blocksize of 5 kB

- Example for “wrong” parameter choice:
  - GM works with fixed blocks of 4 kB
  - For each 5 kB XDAQ block it sends a full 4 kB block and a block filled with only 1 kB

    ==> lower total throughput

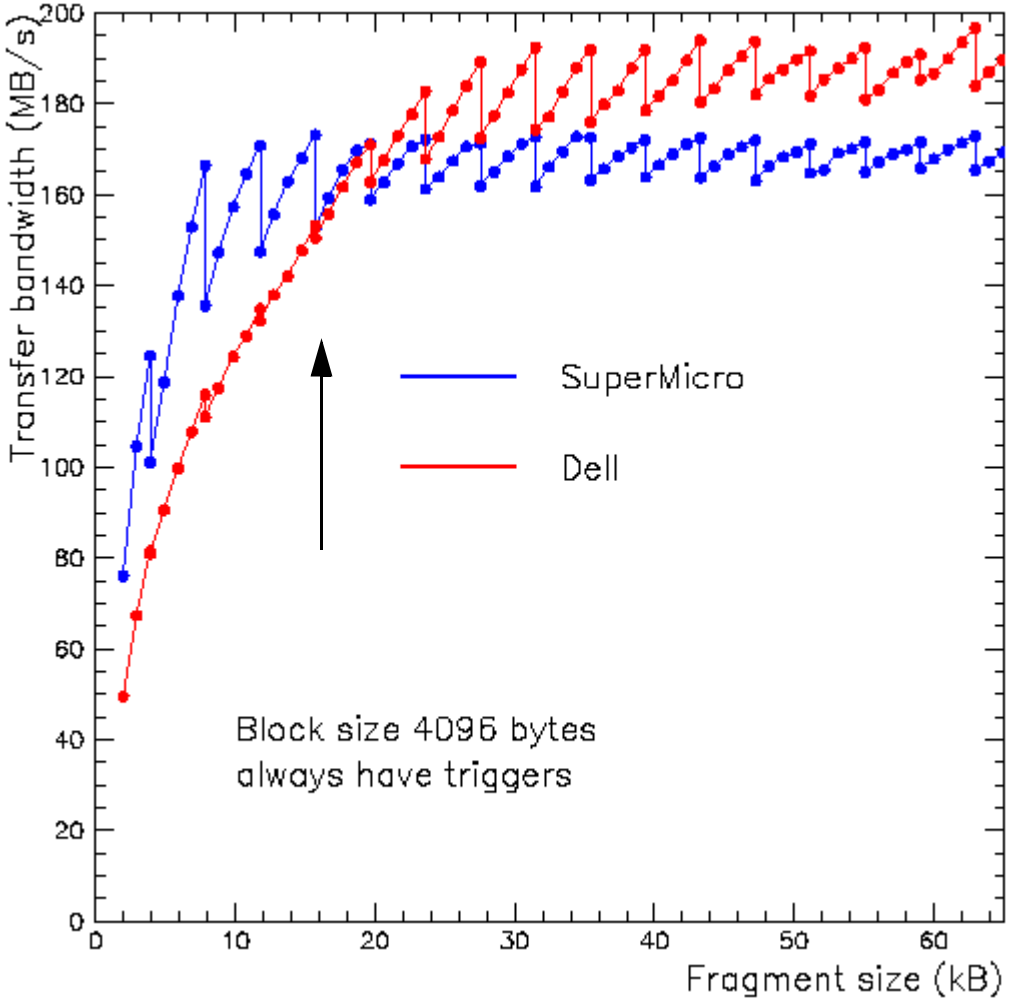
    ==> double sawtooth



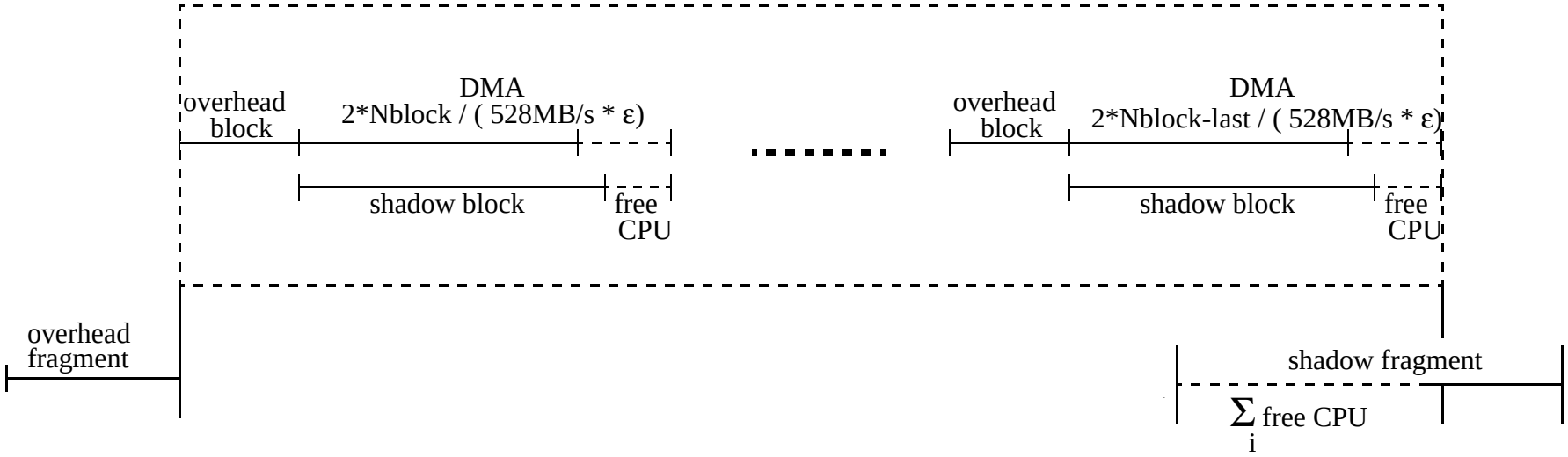
Results : Comparision Dell Power Edge - Super Micro

- high throughput for large frags
  - DELL has 2 pseudo - independent PCI busses (Internally in the bridge they merge to the same bus)
  - Might be reason for higher throughput for large fragments ?
- slow rise:
  - no sawtooth in slow rise (overhead for new block is hidden)

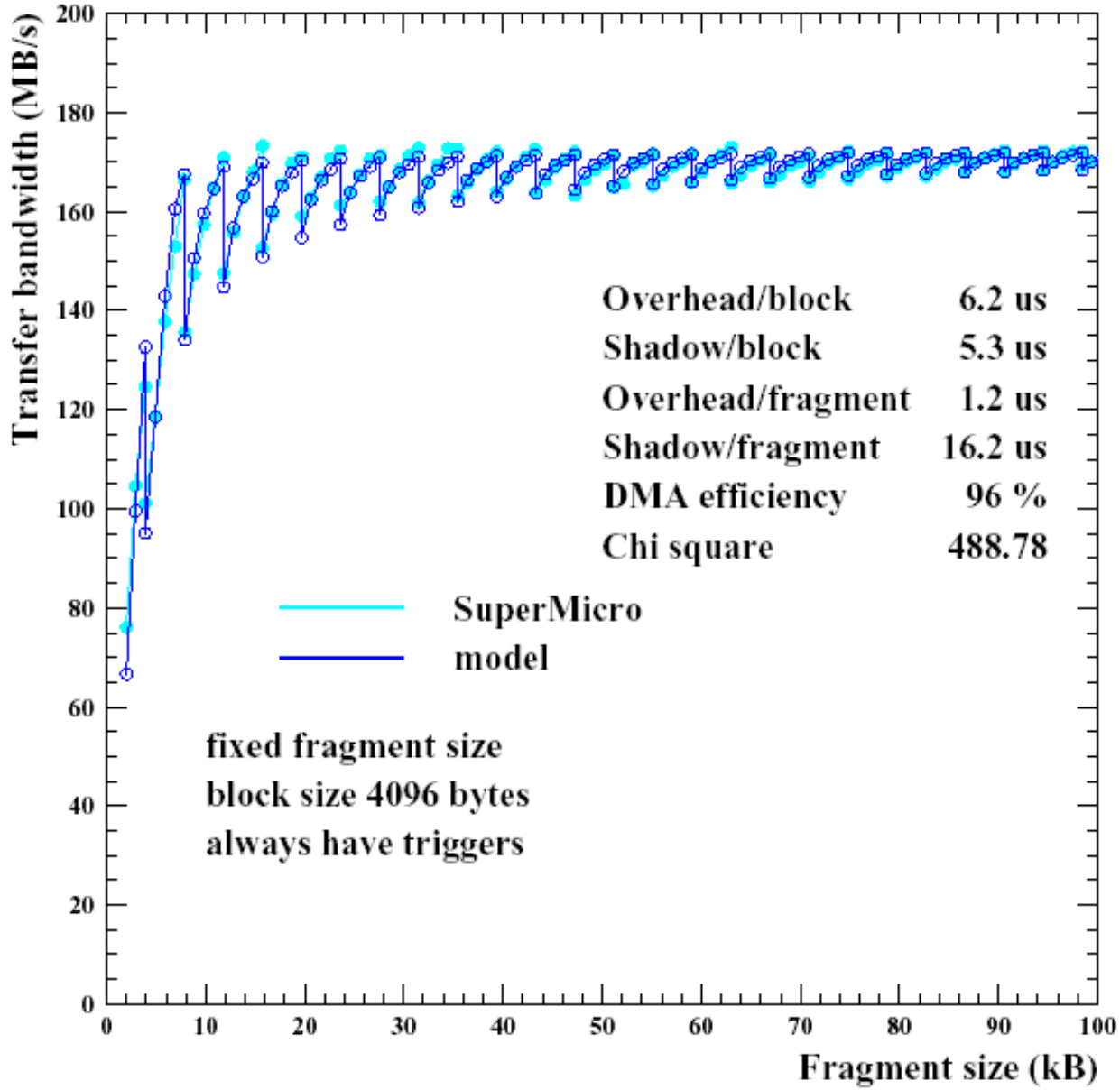
slow rise for small fragments unexplained



Results : Comparision with a simple model

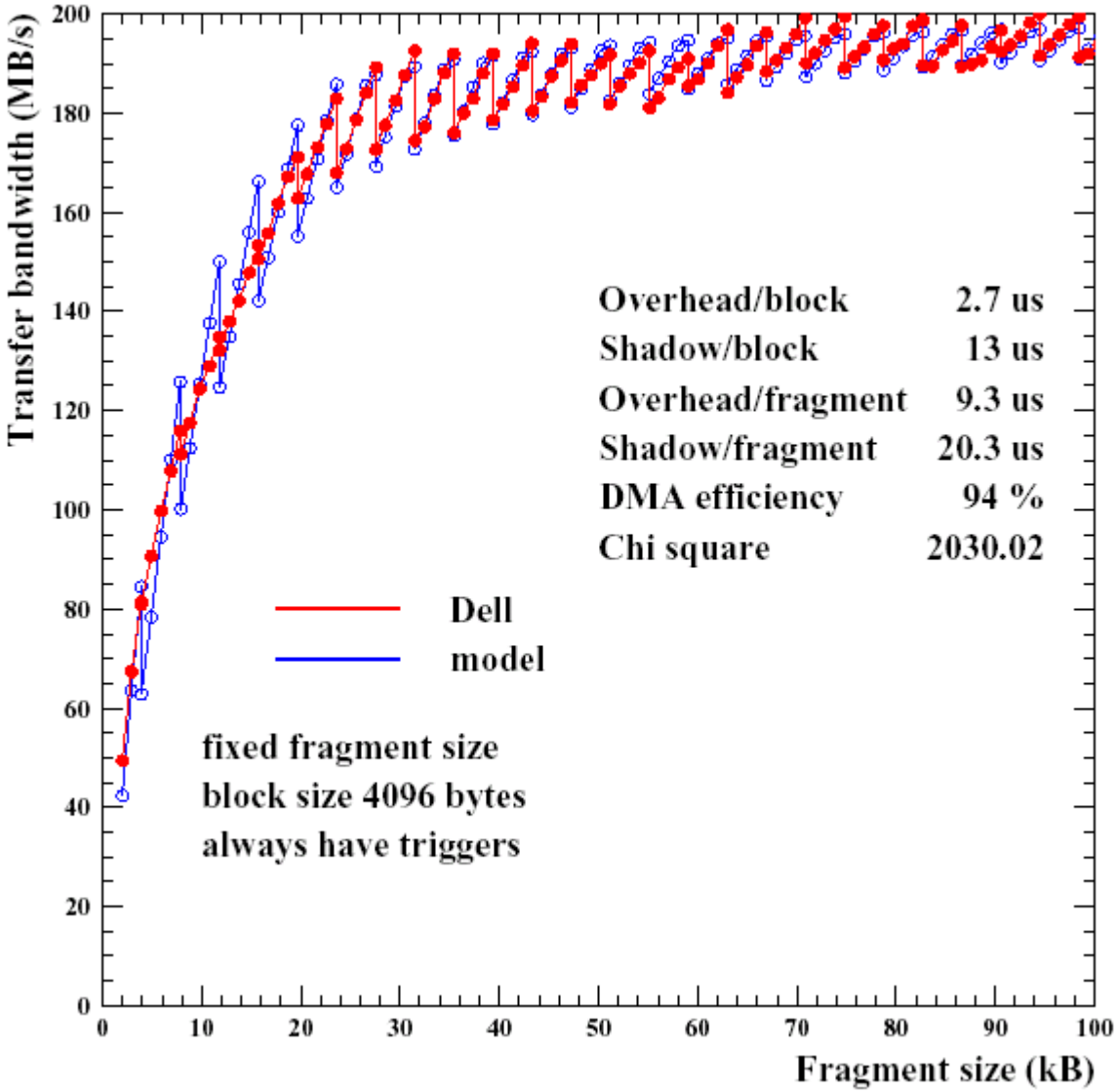


- try to fit to this model with 5 parameters...

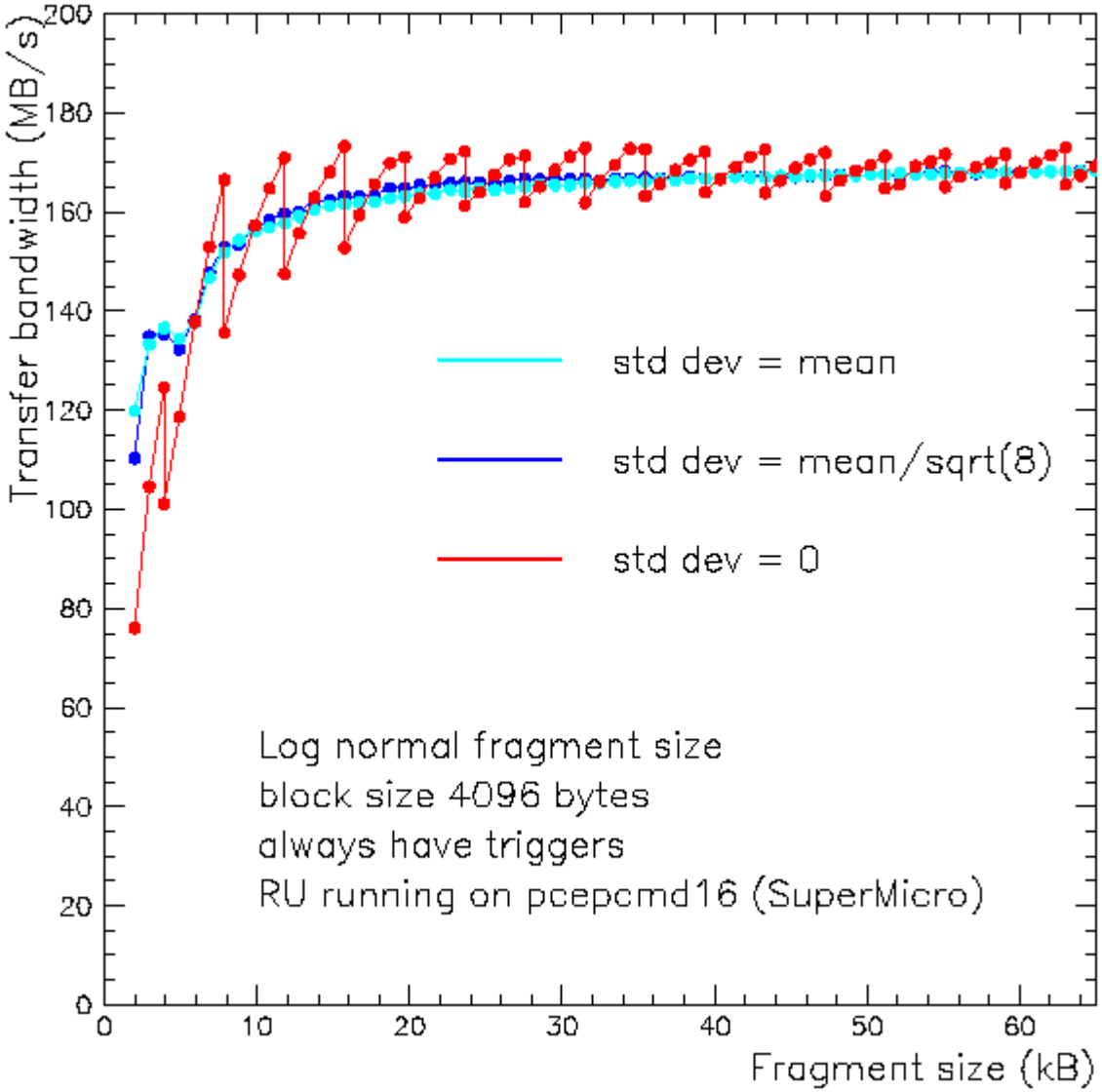

Transfer bandwidth (MB/s)
Fragment size (kB)
Overhead/block 6.2 us
Shadow/block 5.3 us
Overhead/fragment 1.2 us
Shadow/fragment 16.2 us
DMA efficiency 96 %
Chi square 488.78
SuperMicro
model
fixed fragment size
block size 4096 bytes
always have triggers

- NOT correctly modelled : the two pseudo independent PCI busses.
- 4096 bytes at 250MB/s needs 15.6 µs
  13 µs + 2.7 µs = 15.7 µs
  ==> Network limited?
- Large Shadow/block gives large jumps
- small overhead/block gives high throughput
- model does not describe slow rise details

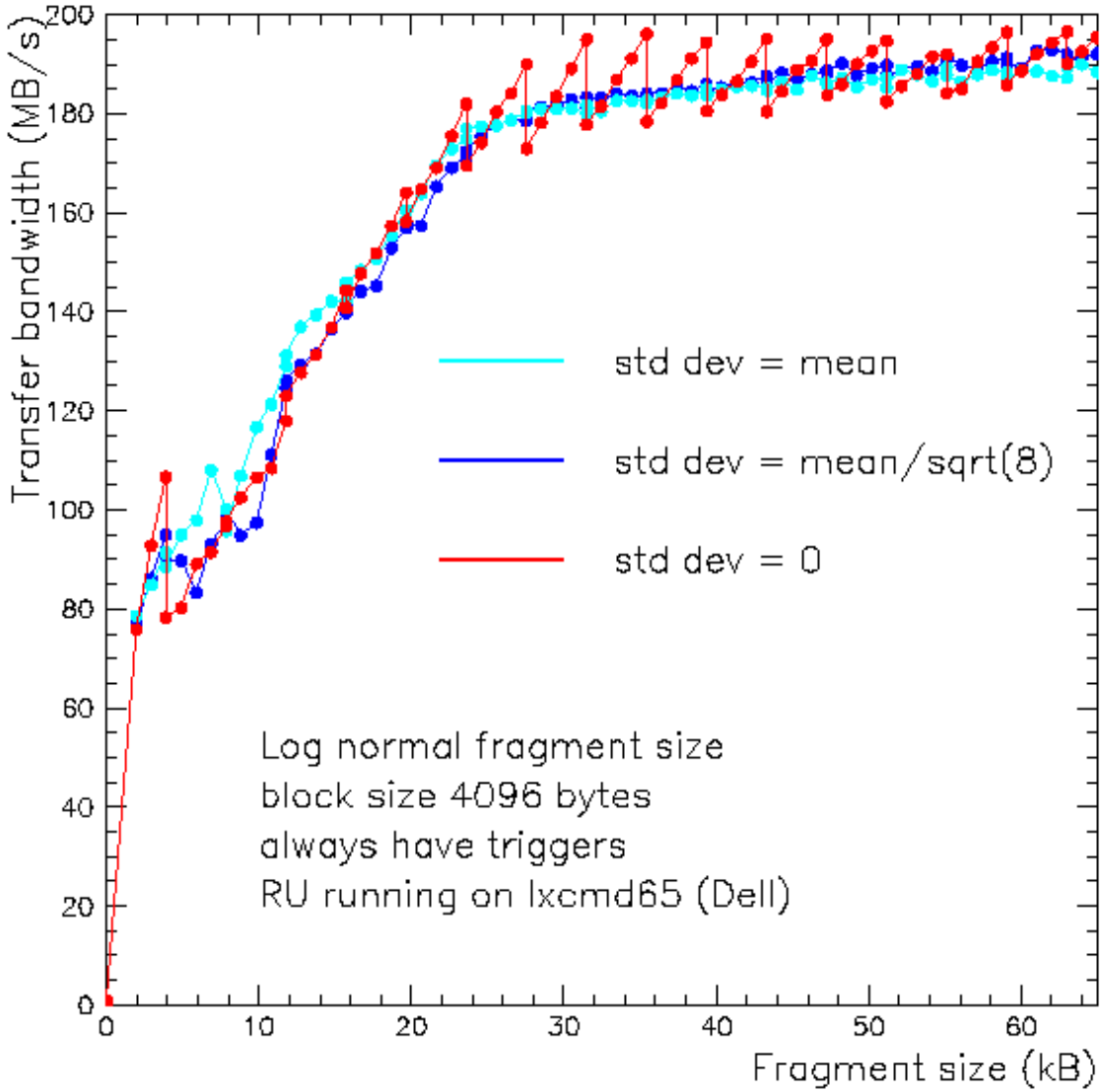
No clear understanding yet



Results : variable fragment size SuperMicro



Results : variable fragment size DELL Power Edge

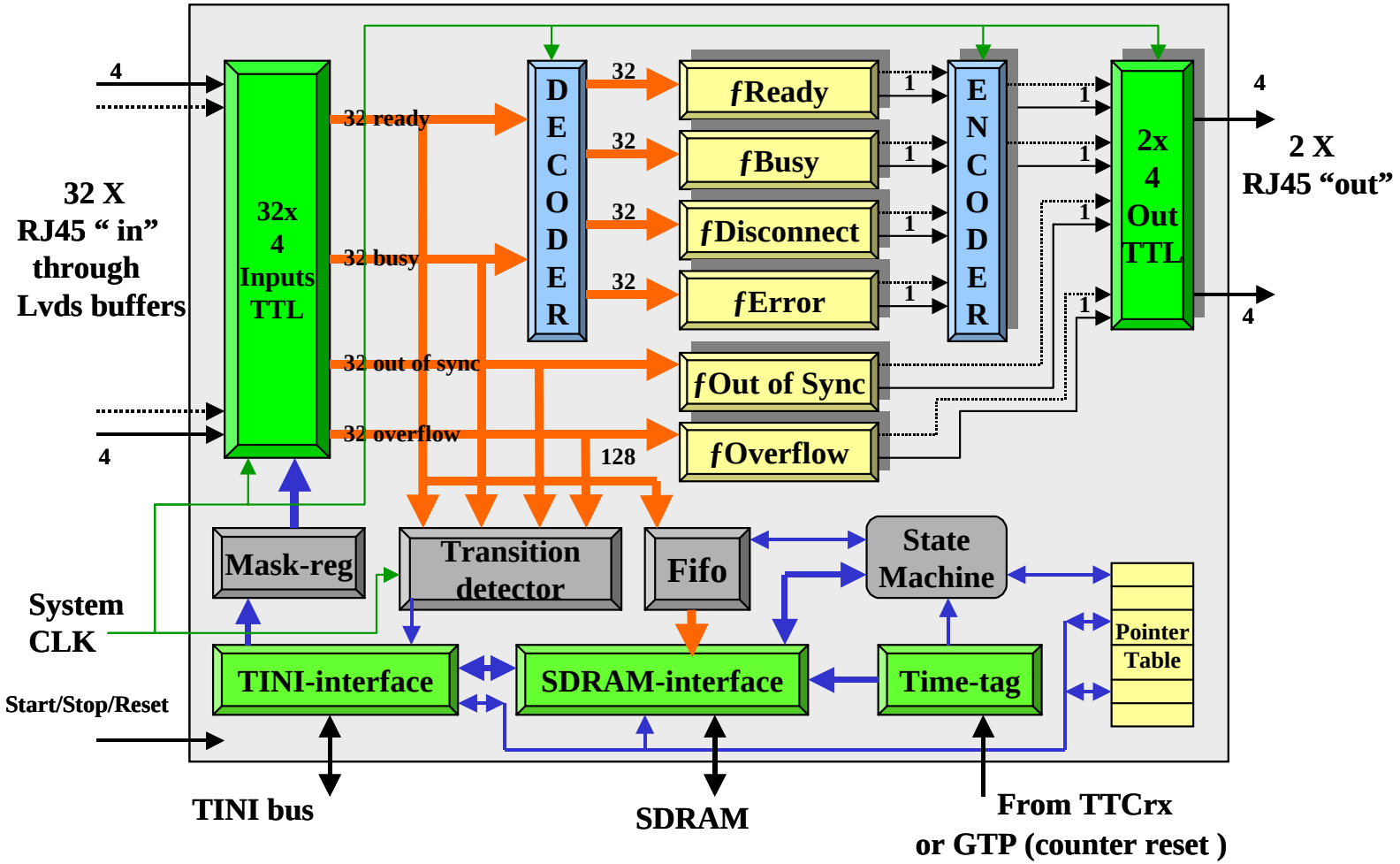


Future Activites

- Try to understand DELL
  - why is the rise at low fragment sizes so slow
- Upgrade the Event Manager
  - Fedkit like architecture should eliminate current bottle neck of TTCrx readout
  - Do systematic measurements changing bundling parameters, and number of available Fedkit blocks
- Implenent direct EVM protocol
  - This will be done if Steve's EVM application is ready to use.
- Merge RU-bench with FRL bench
  - Complete system with SLINK - Merger - FRL - Myrinet Fed-Builder / RUI - RU - BU
  - Implement MAZE (library for Barrel shifter Event Building Protocol)
- Merge with Filter Unit

FMM Fast Merger Module

- under development



TTCrx Tester

- Currently DMA is being implemented
  - needed for efficient readout
  - can be used in a Fedkit configuration which reads out the TTCrx data
  - Data has to be bundled in order to increase efficiency