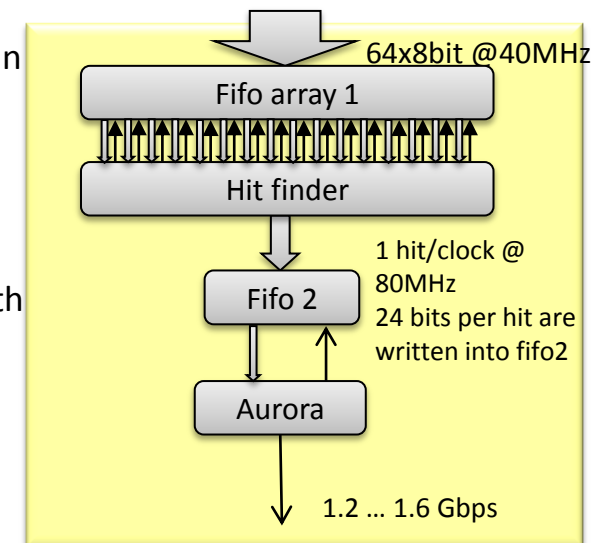


DHP 1.0 data-loss estimation

Mikhail Lemarenko

- What losses are expected for the DHP 1.0 chip, given it would need to handle occupancies up to 3%?
- Assumptions:
 - Input data and triggers (first order approximation) poissonian distributed
 - Trigger rate $\sim 30\text{kHz}$, can intersect each other (frame rate $\sim 50\text{kHz}$)
 - Input 64x8bits channels @ 40MHz ($\Leftrightarrow 256\text{ch. @ } 10\text{MHz}$)
- Current DHP structure:
 - Zero suppressed data fall to the first Fifo array , only non-zero values remain to be searched by the hit finder
 - Hit finder finds one value per clock@80MHz. One clock is additionally needed to go from one row to the next one, even though if it is empty. Found hits are sent to the next Fifo.
 - Hits are removed from the output fifo @ 1.6Gbps, With effective bandwidth of $1.6\text{Gbps} / \frac{24\text{bits}}{\text{hit}} \cdot 80\% = 50\text{MHits/sec}$
 - Current fifo size are for a while fixed to:
 - Fifo array 1: 64 inputs wit the depth of 8
 - Fifo 2: depth 512.



- External

- For the given input and output rates, assuming that the DHP itself does not produce any losses except those linked with bandwidth limitations, it can be estimated that our maximum acceptable occupancies assuming the lossless transmission are:

- Non-trigger mode: $P \sim 2.1\%$

- Triggered @ 30kHz: $P \sim 5\%$

- Internal

- DHP core has 3 different zones with varying processing times. Losses on the borders of these zones are dependent on 3 main factors:

- Fifo1 depth(before hit finder)

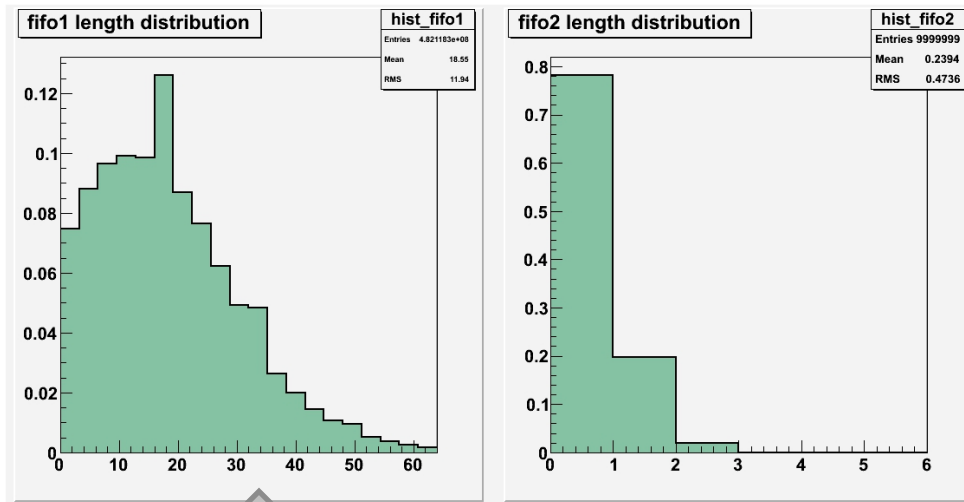
- Fifo2 depth(output fifo)

- Hit finder performance

- Hit-finder efficiency is one of critical factors. Depending on its performance, different fifo sizes are necessary to get less than 1% of losses for the 3% of signal occupancy. Fifos can not be very deep for the reason of limited hardware resources
- In the following report 3 simulated scenarios shall be presented for different implementations of hit finder and fifos.

Current hit finder(implemented on DHP0.1) profile @3% occupancy

Fifo profile distribution @3% occupancy (minimum necessary estimated)

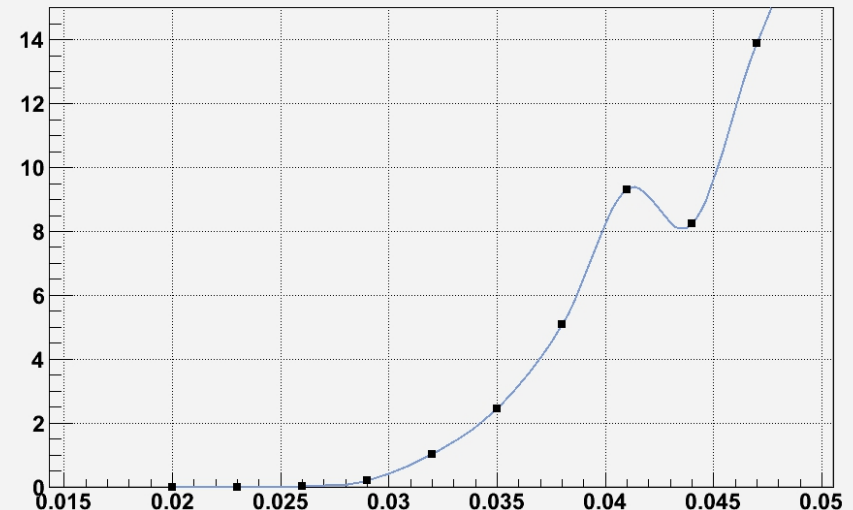


- This hit finder is quite slow, so the main waiting queue is situated in front of it (in the first fifo)

Parameters

- Minimum Fifo parameters to achieve 1% of losses:
 - Fifo array 1 depth : 64(73kbit)
 - Fifo 2 : 4(96bit)
 - Losses : 0.15%
 - Total estimated area : 0.4mm²

Losses vs occupancy

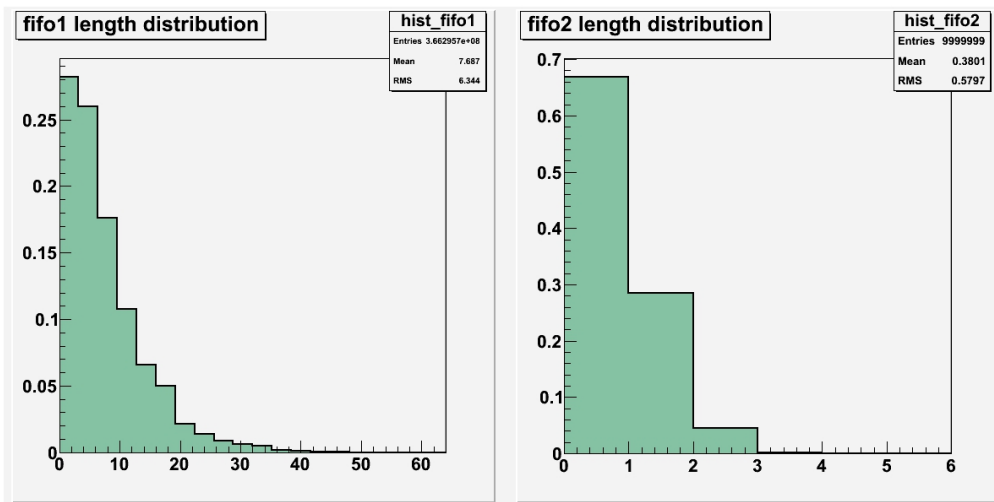


Upgraded hit-finder profile @3% occupancy

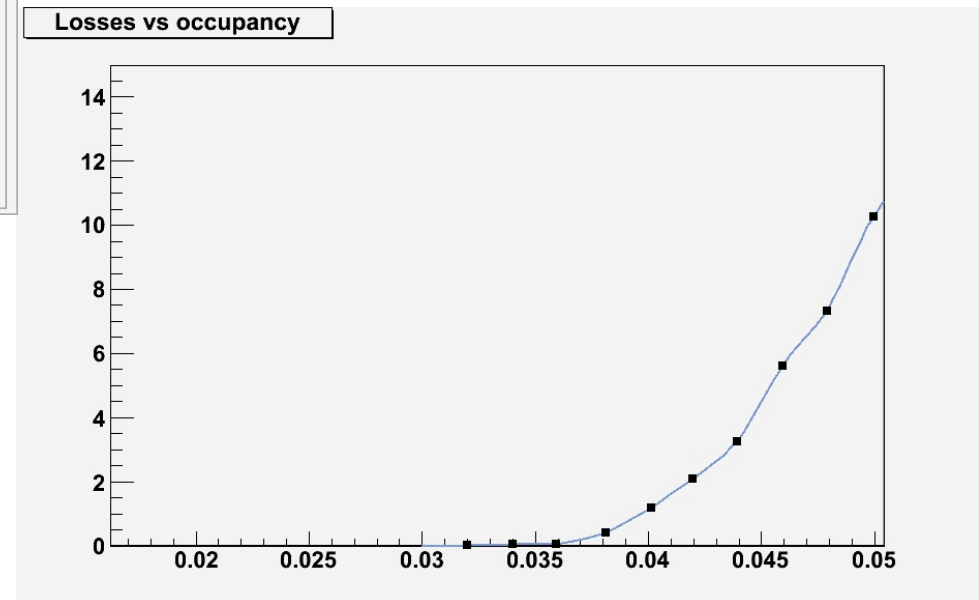
Fifo profile distribution @3% occupancy

Parameters

- Same as in previous scenario

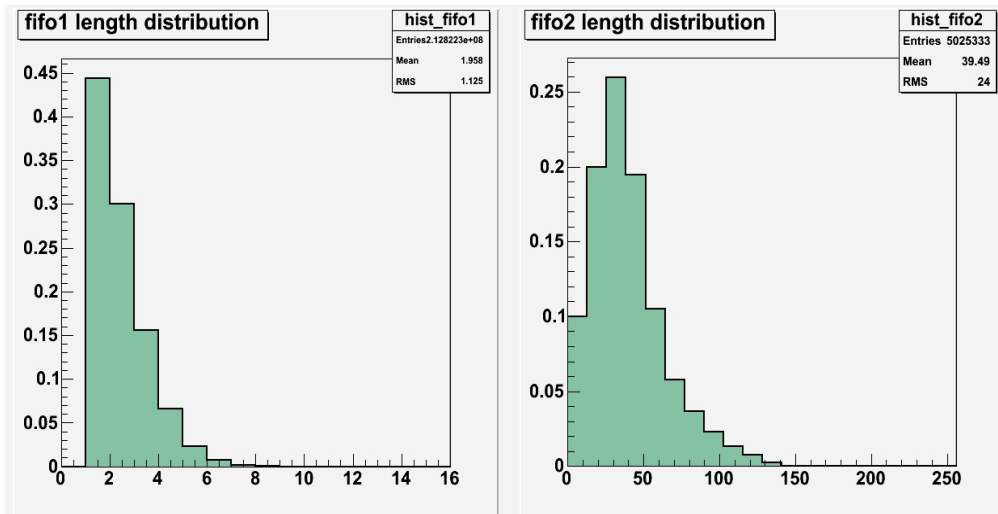


This hit finder need only one clock cycle to pass from one row to the next one



Clever hit-finder scenario profile @3% occupancy

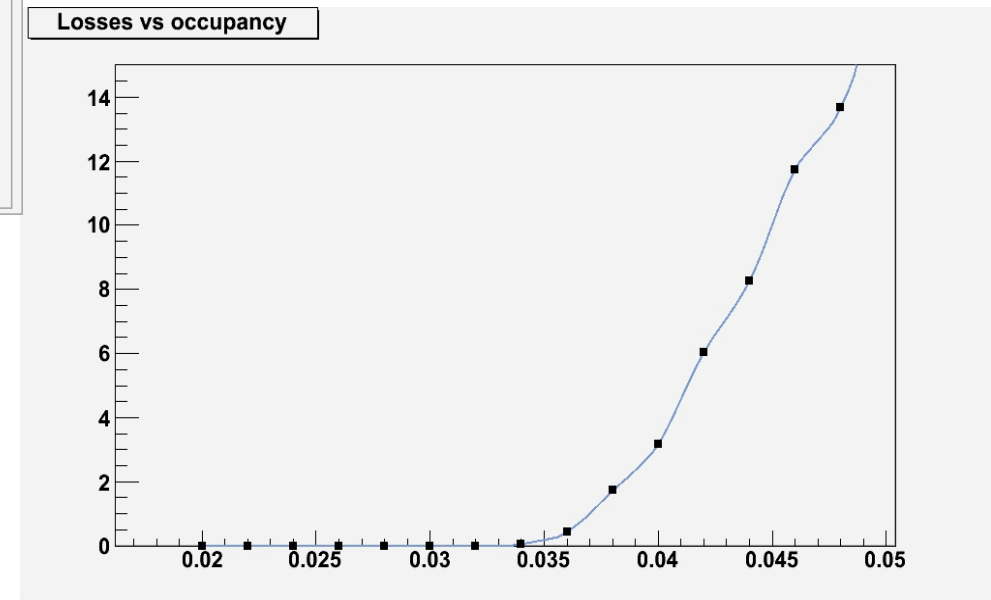
Fifo profile distribution @3% occupancy



Here we assume we can get one hit per clock cycle regardless how hits are distributed

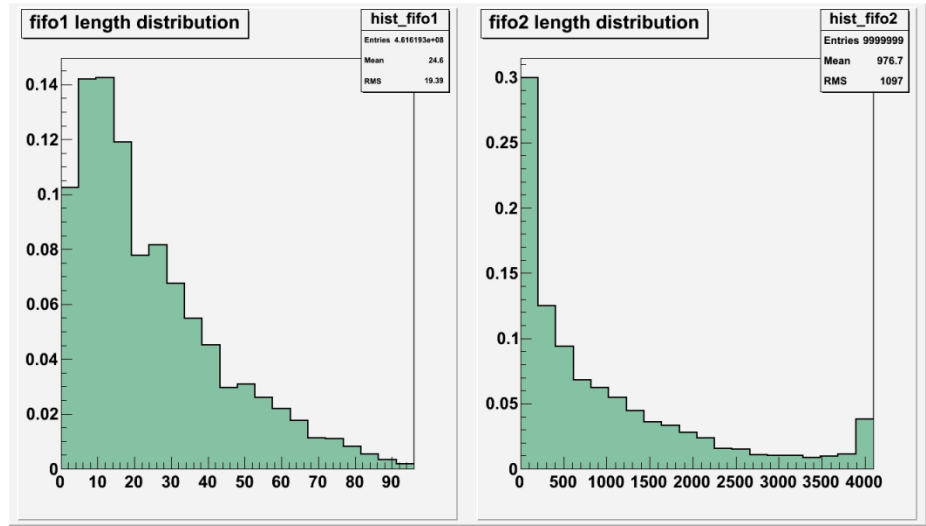
Parameters

Fifo array 1 depth	:	16(18kbit)
Fifo 2	:	256(6kbit)
Total required area	:	0.14mm ²



Resources needed for the almost perfect performance

Fifos length probability profile@5%



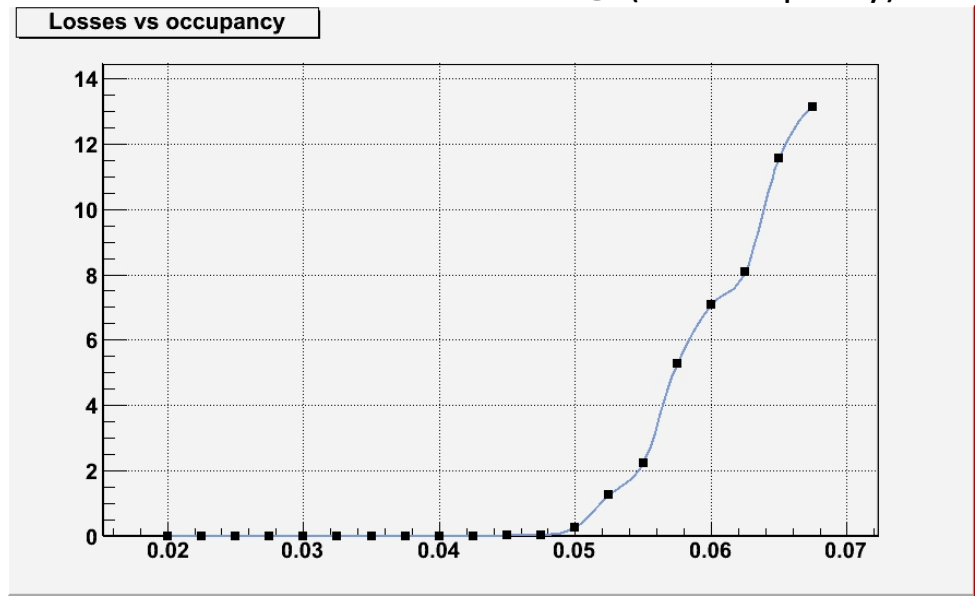
Estimated resources

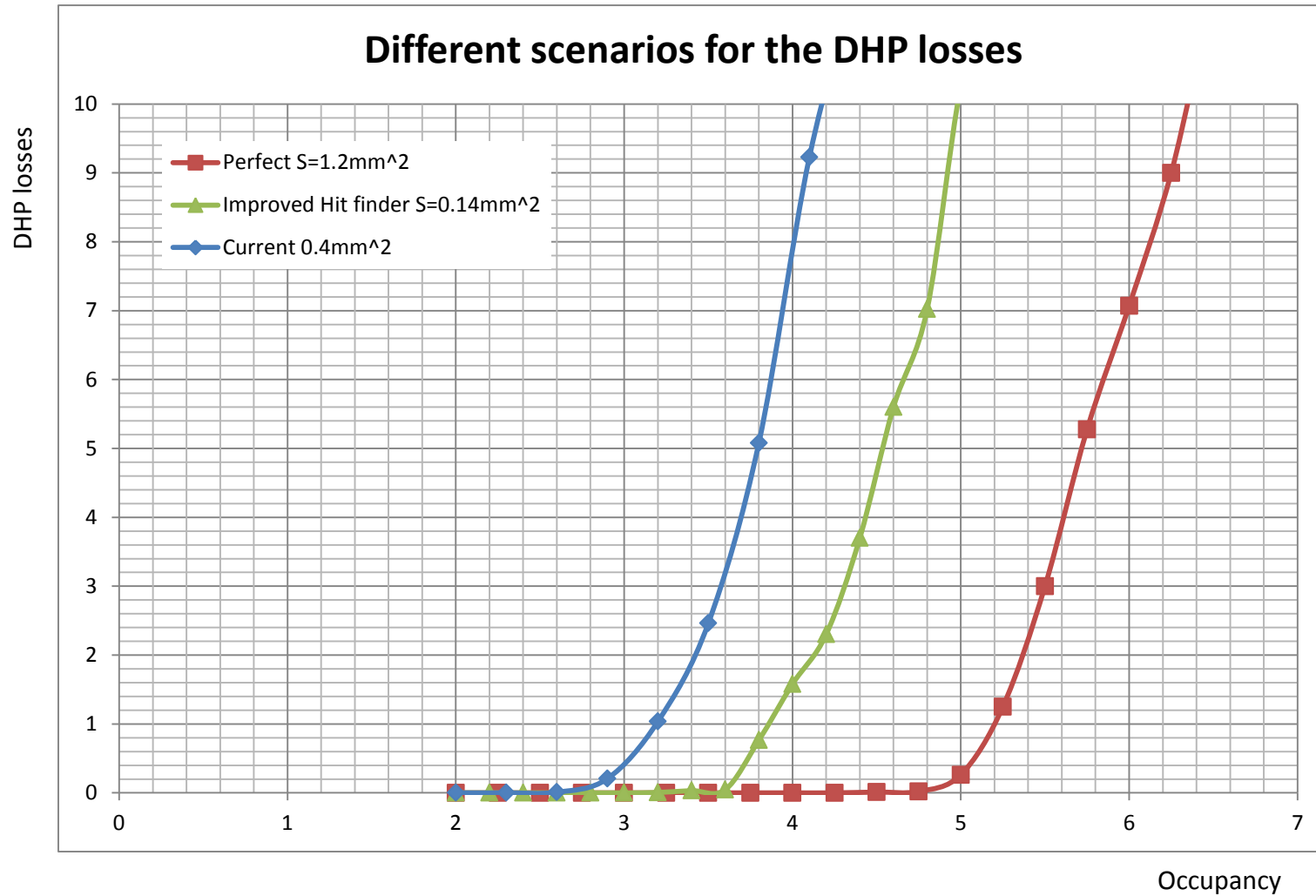
- Fast hit-finder (1 removal/clock)
- Fifo1 array : 96 (110kbits)
- Fifo2 : 4096 (96kbits)
- Total area needed (assuming 8um²/flip-flop): 1.6mm²

Total losses would be: 0.46%
@ (5% occupancy)

Drawbacks:

- Resources hungry scenario
- Not sure scenario if it is technically achievable.





Further ideas for the better DHP performance → data compression

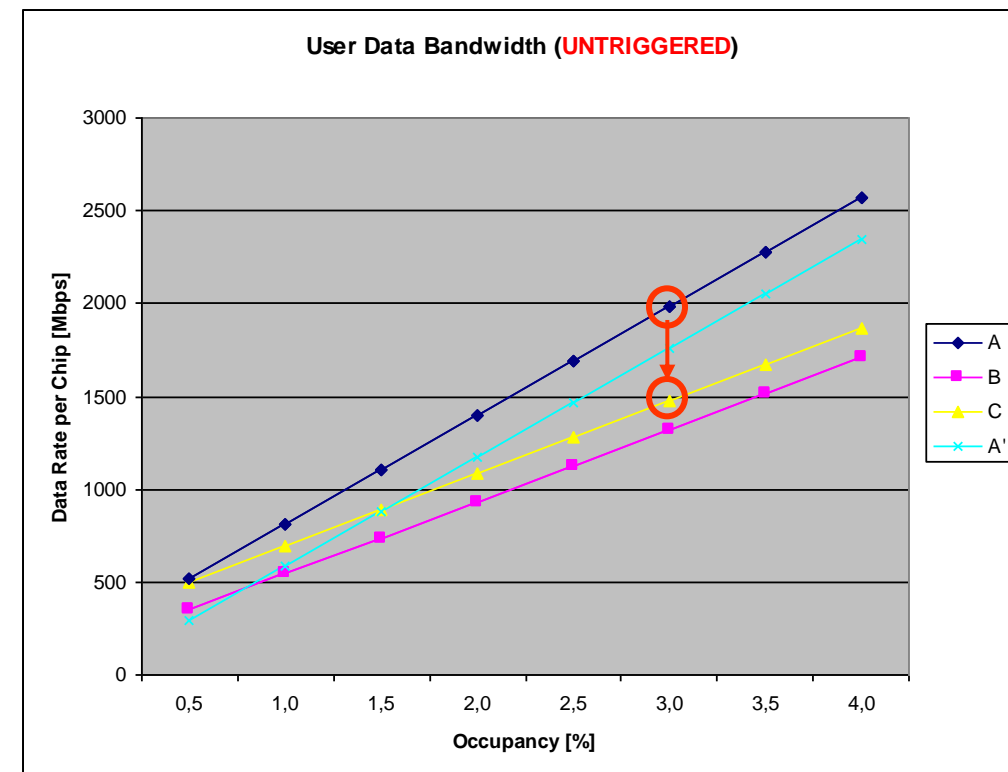
- Address Naming Convention
 - Electronic channels:
 - 8 bit Switcher channel address: 0..191 (6 x 32 ch. Switcher)
 - 8 bit DCD channel address: 0..255 (one DCD → one DHP → one Link)
 - Physical (pixel) addresses:
 - 10 bit pixel row address: 0..767
 - 6 bit pixel column address: 0..63
 - this format is used for the hit data words inside the DHP

- Current data format (DHP0.1)
 - generic 24 bit hit data format: <10b row><6b col><8b ADC>
 - 10 bit row address: 768 pixels
 - 6 bit column address: 64 pixels
 - 8 bit ADC range: 0..255
 - want/need to add common mode value once per 256 pixels (every four rows)
 - send c.m. as data word (keep 24 bit alignment) → waste of bandwidth
 - alternative: store c.m. values for one frame and send the data in a special block → makes the data path more complicated
- Idea 1
 - proposal: “row header” format
 - send row address only once and send data words with column address and ADC value
 - 16 bit length for both words
 - row header (including c.m.): < 10b row><6b cm>
 - data word: <6b col><8b ADC><2b reserved>
 - disadvantage: one cannot distinguish row headers from data words by their structure (maybe heuristics can do)

- Idea2
 - reorder addresses bits to define flags for bits row headers and data words
 - row header: <row flag = 0><9b row><6b cm>
 - data word: <data flag = 1><7b col><8b ADC>
 - send row header every 128 (and not 256) pixels (adds some overhead)

- Data formats shown in the plot
 - A: generic 24b data (+ 24b c.m. word per row)
 - A': A without c.m. word
 - B: 10b/6b row/col addresses
 - C: reordered 9b/7b row/col addresses

➔ data reduction of 26% @ 3% occ.
for a change from A to C



- Frame Header (32 bit): <data type><res.><chip ID><frame ID>
 - data type (3 bit): [raw data, processed data]
 - reserved (5 bit)
 - chip ID (8 bit)
 - frame ID (16 bit)
 - ➔ will always be send at start of a new frame, independent of trigger
- Row Header (16 bit): <flag><row address><common mode>
 - flag (1 bit): 0 ➔ row header
 - row address (9 bit)
 - common mode (6 bit)
 - ➔ will only be send if hit data for the active row is available
- Data Word (16 bit): <flag><column address><ADC>
 - flag (1 bit): 1 ➔ hit data
 - column address (7 bit)
 - ADC (8 bit)

- With the DHP0.1 we are not yet at the point to be able to digest 3% of input data occupancy. However we do not see a big challenge in achieving that result. We have several options:
 - Memory resources increase
 - Hit finder optimization
 - Different data format

- We are likely to be even better than that