University of Birmingham
Montag, Feb. 21st, 2000.

FTT Level 1
my personal view

André Schöning
ETH Zürich

- New idea! Digitalisation on Adapter cards using a drift chamber receiver chip?
  - hit ntuples
- L1 Design problems:
  - cabling problem (LVDS $< 2 - 5$ m)
  - calculation of z (before/after segment finding)
  - validation of track segments
    $\rightarrow$ CAMs versus LUT versus Logic
- L1 Trigger option
  - trigger decision algorithm
  - I/O problem

Hit Ntuples

- GHz ASIC could generate hit nutples: t_{hit}, charge, z-coordinate
- Chip proposed by Thom Wolff who is one of the inventors of the DCRphi trigger
- Chip development can be done by Thom Wolff (e·vision)
- Advantage: early digitalisation and no analog losses

Cabling problem

- Location of L1 crates and L2/L3 crates needs discussion
  - desirable scenario: L1 crates close to CJC FADCs and L2/L3 crates close to central trigger

    $\rightarrow$ distance about 10 m
- LVDS can drive signals (5+x) m. Better less than 5 m.

  $\rightarrow$ All FTT crates should stay together in one rack
- Alternative: Early digitisation of hits.

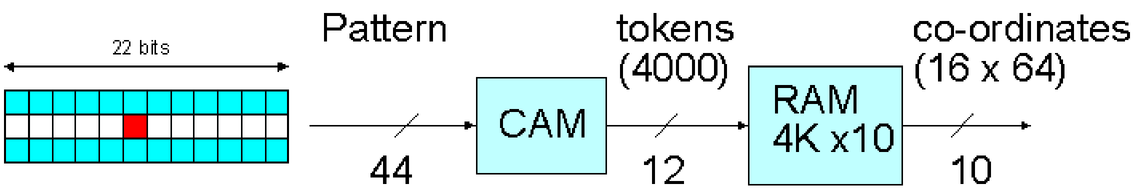
Determination of z

- Pipelined 8 bit fixed point division can be done at frequencies up to 100 MHz (8 steps pipelined) in APEX FPGA.
- z-division can start before track segment finding thus saving time but:
  - only on individual hits
  - overhead from not linked hits
- z-division starts after track segment finding:
  - loosing time
  - worse precision after charges are summed up
  - overhead from multiple used hits
- Conclusion: within one BC several z-divisions can be done, maybe more rapidly than hits are coming in!
  $\rightarrow$ Why not start immediately after QT
- Needs to be discussed...

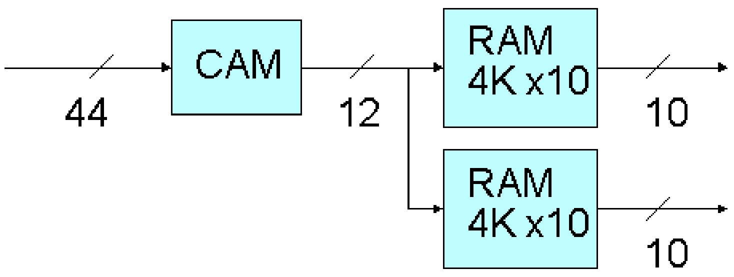
Track Segment Finding

- Concept with encoded CAMs
- Any alternatives?

Primary tracks for L1:



Secondary tracks for L2:



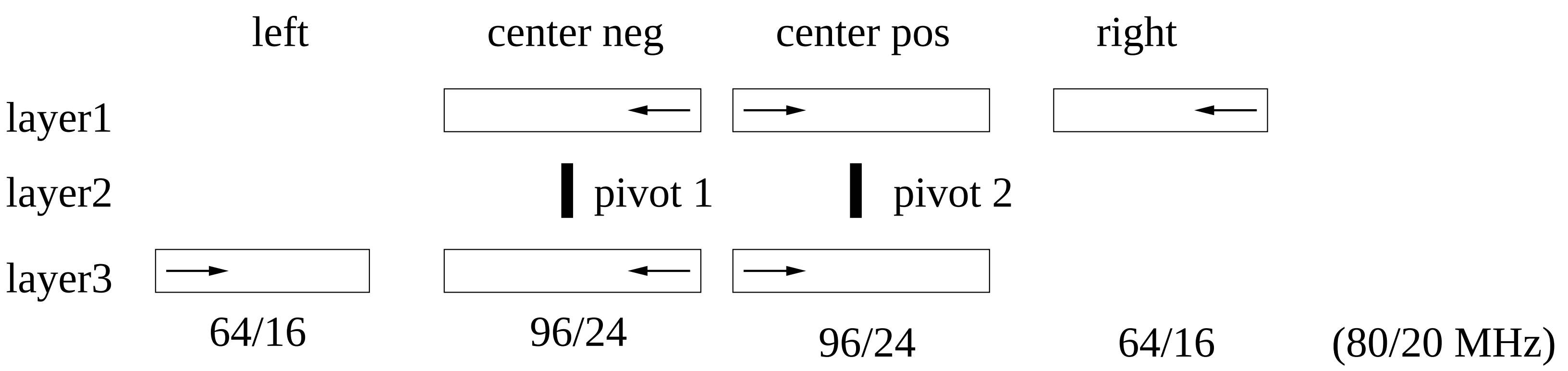
Validation of track segments

- How to perform everything in parallel?
  - CAM: encoded mode is deterministic (good) but can not treat multiple matches (bad). Therefore, several CAMs are needed in parallel with some load balancing
  - CAM: unencoded mode cannot run with tag fields (RAM access) if RAM is not 1 bit deep. This could be an attractive solution for the L1 trigger where kappa-phi histos have to be ORed (Logic OR of up to 1000 words).
  - LUT with output addresses pointing to RAM similar to encoded CAM mode but hardwired. Unvalid entries have no address pointer.
  - hardwired logic checks all possible hit combinations. Similar to unencoded CAM but hardwired.
- Timing
  - matched track segments are in general valid for more than one BC
  - after validation a bunch of track segments for different bunch crossings have to be treated $\rightarrow$ additional shift register

$\rightarrow$ More detailed ideas needed here!!!

Andre.Schoning@desy.de

Shift Register



combinations:	center x center	(24+24)*(24+24)	= (16+16+16)*(16+16+16)	9 CAMs (32bit)
	center x right	(24+24)*16	= (16+16+16)*16	3CAMs (32bit)
	left x center	16*(24+24)	= 16*(16+16+16)	3CAMs (32bit)

- 15 CAMS for each pivot element makes 30 CAMs in total
- probability of validating more than one track segement in CAM:

N_tracks	probability
0	0
1	0
2	0.03
3	0.1
4	0.2
5	0.3

University of Birmingham

Andre.Schoning@desy.de

University of Birmingham

Shift register:

A1	B1	C1	D1	E1	A2	B2	C2	D2	E2	A3	B3	C3	D3	E3	A4	B4	C4						

pivot element

A1	B1	C1	D1	E1	A2	B2	C2	D2	E2	A3	B3	C3	D3	E3	A4	B4	C4						

several CAMs containing combinations of hit patterns:

AA,AB,AC,AD,AE,BA,BB,BC, and so on...

L1 Trigger Option I

- Actual trigger algorithm is fast!
  - Cluster finding algorithm on a small histogram has been succesfully tested for L2 linking and could be run at up to 180 MHz. 5 cycles are needed.
  - Any projection on track numbers in the kappa-phi histo is done in one cycle with up to 200 MHz
  - 16 integer numbers (e.g. track multiplicites) with 6-10 bit representation can be added in two cycles with up to 200 MHz

  $\rightarrow$ No serious problems expected with the L1 track linking!!!

- I/O is the most severe problem for the design:
  - desirable size of kappa-phi histo is 60x16 (to trigger tracks down to 100 MeV)
  - data volume is 960 bits per layer without t_0 info
  - full data size is 4x2x960=7680 bit per BC corresponding to a data transfer rate of 9.6 Gbyte/s!
  - LVDS achieves at 100 MHz 600 Mbyte/s
  - data transfer into single FPGA with 500 pins at 100 MHz on board gives 6 Gbyte/s.

  $\rightarrow$ Mission Impossible!

Timing Specifications at Trigger Level 1

Process	**time [μs]**
drift time of most distant hits	1.0
$Q - t$ analysis	0.3
filling of shift registers	0.1
track segment finding	0.2
collecting track segments	0.2
L1 trigger processing	0.2
total	2.0

Table 1: Timing specifications at trigger level 1. Note that the track segment finding begins before all drift times are available, such that several bunch crossings are processed in parallel. The figure of $0.2\ \mu\mathrm{s}$ quoted for track segment finding thus represents the actual delay incurred, rather than the full time necessary to process a single event.

Linking Results

- e.g. elastic J/Ψ candidate in κ-ϕ plane ($\Delta\frac{1}{p_t}$ = 1.25/GeV)

```
event        11
linked segments:
 -7 000000000000000000000000000000000000000000000000000000000000
 -6 000000000000000000000000000000000000000000000000000000000000
 -5 000000000000000000000000000000000000000000000000000000000000
 -4 000000000000000000000000000000000000000000000000000000000000
 -3 000000000000000000000000000000000000000000000000000000000000
 -2 000000000000000000000000000000000000000000000000000000000000
 -1 000000000000000000000000000000000000000000000000000000000000
  0 000000000000000000000000420000000000000000000000000000000000
  0 000000000000000000000000000000000000000000000000003000000000
  1 000000000000000000000000000000000000000000000000011000000000
  2 000000000000000000000000000000000000000000000000000000000000
  3 000000000000000000000000000000000000000000000000000000000000
  4 000000000000000000000000000000000000000000000000000000000000
  5 000000000000000000000000000000000000000000000000000000000000
  6 000000000000000000000000000000000000000000000000000000000000
  7 000000000000000000000000000000000000000000000000000000000000

after weighting:
 -7 000000000000000000000000000000000000000000000000000000000000
 -6 000000000000000000000000000000000000000000000000000000000000
 -5 000000000000000000000000000000000000000000000000000000000000
 -4 000000000000000000000000000000000000000000000000000000000000
 -3 000000000000000000000000000000000000000000000000000000000000
 -2 000000000000000000000000000000000000000000000000000000000000
 -1 000000000000000000000000000000000000000000000000000000000000
  0 000000000000000000000000200000000000000000000000000000000000
  0 000000000000000000000000000000000000000000000000002000000000
  1 000000000000000000000000000000000000000000000000000000000000
  2 000000000000000000000000000000000000000000000000000000000000
  3 000000000000000000000000000000000000000000000000000000000000
  4 000000000000000000000000000000000000000000000000000000000000
  5 000000000000000000000000000000000000000000000000000000000000
  6 000000000000000000000000000000000000000000000000000000000000
  7 000000000000000000000000000000000000000000000000000000000000
```

$\Rightarrow$ two high p_t tracks with opposite charge found

L1 Trigger Option II

- Constraints from a optimistic scenario
  - kappa-phi histo only 30x8 bins with $p_t > 200$ MeV (not bad by the way) giving 2.4 Gbyte/s
  - Each FEM has to send (5+2)*8*2=112 bits per BC corresponding to 140 Mbyte/s (1 optical link or LVDS)
- Maybe generic L2 merger boards can be used to collect from one layer 30*8*2=480 bits per BC corresponding to 600 Mbyte/s output (LVDS 100 MHz). But latency is induced by receiver/sender
- Alternative 1: kappa-phi histo can be split into several phi sectors e.g. on 3 or 6 boards
- Alternative 2: Use 30 optical links (1 fiber) at a frequency of 1.12 GHz and 16 fold multiplexing. That gives 30*26=480 lines on the trigger board at 56MHz

$\rightarrow$ Mission NOT Impossible!