

L2STT Track Fitting

Topics

1. Review
2. News
3. Updated Block Diagram
4. Plans

Track Fit Card

- Receives road+hit information from trigger card
- Coordinate conversion (via lookup)
- Filters multi-hits/layer
- Perform fit to linearized circle

$$\phi(r) = b/r + \kappa r + \phi_0$$

(r, ϕ) = hit position

(b, κ, ϕ_0) = track parameters

This trivially reduces to

$$\begin{pmatrix} b \\ \kappa \\ \phi_0 \end{pmatrix} = \begin{pmatrix} \sum 1/\sigma_i^2 & \sum r_i^2/\sigma_i^2 & \sum r_i^1/\sigma_i^2 \\ \sum r_i^2/\sigma_i^2 & \sum r_i^4/\sigma_i^2 & \sum r_i^3/\sigma_i^2 \\ \sum r_i^1/\sigma_i^2 & \sum r_i^3/\sigma_i^2 & \sum r_i^2/\sigma_i^2 \end{pmatrix}^{-1} \begin{pmatrix} \sum \phi_i r_i^1/\sigma_i^2 \\ \sum \phi_i r_i^3/\sigma_i^2 \\ \sum \phi_i r_i^2/\sigma_i^2 \end{pmatrix}$$

Done obvious things:

$1/\sigma_i^2$ is constant multiplier (actually two, σ_{CFT} and σ_{SMT})

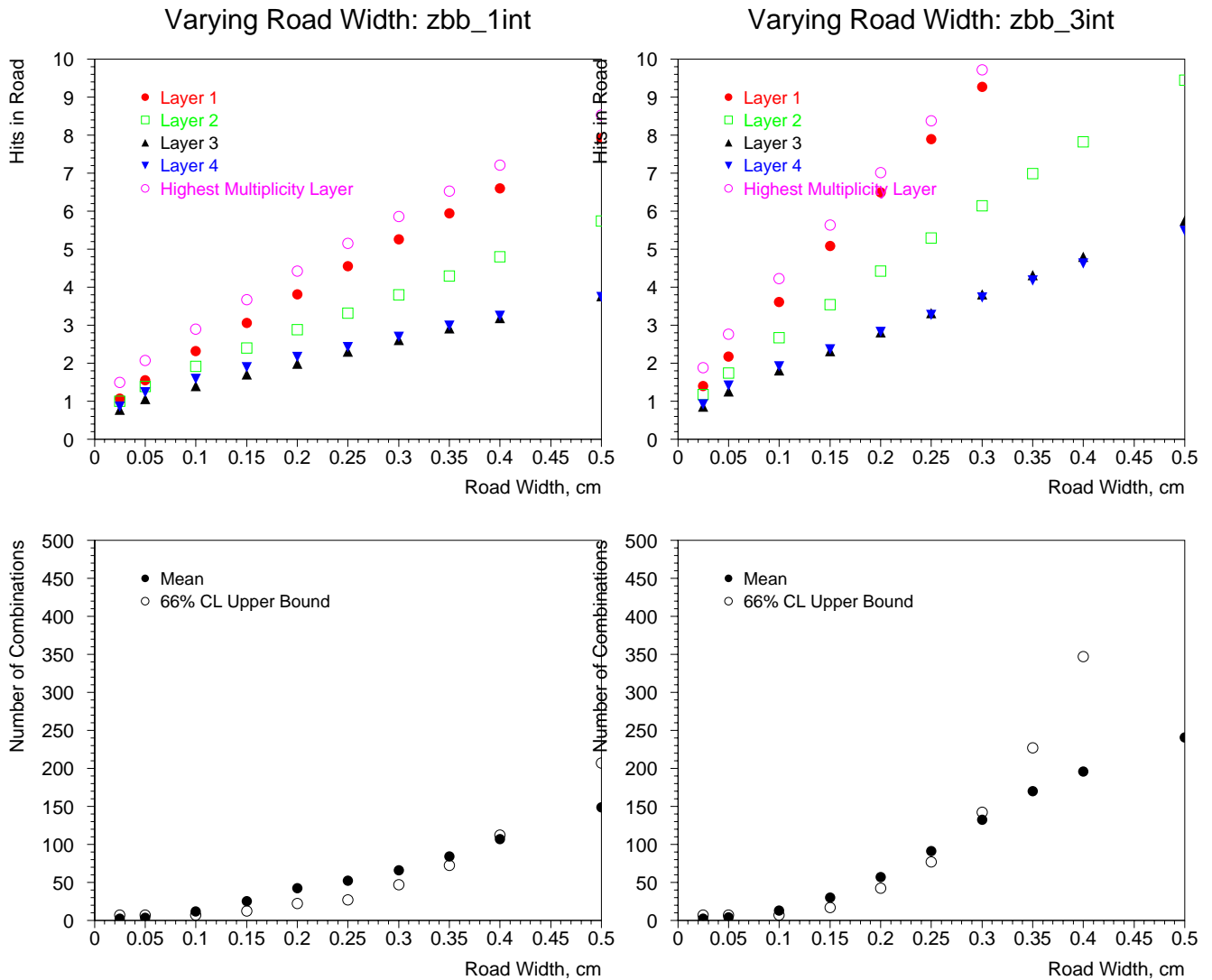
Order to minimize operations

Running products, $r^5 = r \times r^4$

...

The hit multiplicity problem

Which hits do we use?



Filtering algorithms, $< 25\mu\text{s}$ for processing

1. **Static Road-Center:** Use hits closest to circle defined by the CTT hits and $(0,0)$
2. **Dynamic Road-Center:** Use hits closest to circle defined by the CTT hits and hits in SMT layer 4 (Looping...)
3. **All Combinations:** All combinations of hits are fit. Choose best χ^2 .
4. **Best Combination at Layer:** Moving from the outer SMT layer inwards the fit is performed at the current layer using the best result from performing fits on all combinations in the preceding layer.

Fraction of tracks with hits from one MC Track

Fit Algorithm/ Minimum Hit Layers	$Z \rightarrow b\bar{b}$	
	1 int.	3 int.
Static Road/4	0.53	0.43
Static Road/3	0.76	0.69
Dynamic Road/4	0.41	0.28
All Combinations/4	0.65	0.60
All Combinations/3	0.79	0.71
Best at layer/4	0.51	0.43

Doing $\chi^2(\text{Fit-True})$ and rate vs. efficiency (undergrad)
” Floating point precision(JDH, Paris?)

Fit processors: Alpha

- Time (@500 Mhz)

Execution times/track (μ s)

Fit Algorithm/ Minimum Hit Layers	$Z \rightarrow b\bar{b}$	
	1 int.	3 int.
Static Road/4	3.8	4.1
Static Road/3	12.1	12.5
Dynamic Road/4	7.1	8.2
All Combinations/4	16.7	18.7
All Combinations/3	33.4	38.3
Best at layer/4	10.5	11.2

⇒ Can do 3-6 tracks/CPU

- Space, DØ standard is 1 CPU/board
- Power. DØ says OK...
- Cost. (\$5k-\$6k/CPU board)*

Cost of hypothetical system/16 tracks fit,
3 boards \times \$5k \times 6 sectors = \$90k + crates + misc
(MBT)

However, space for racks not there...?

*Previously, \$15k.

Fit Processors: TI320C6x DSP

- Time for algorithm static road center
 - Hal, et. al.
 - straight C code to *fixed-point* DSP (no hand optimization),

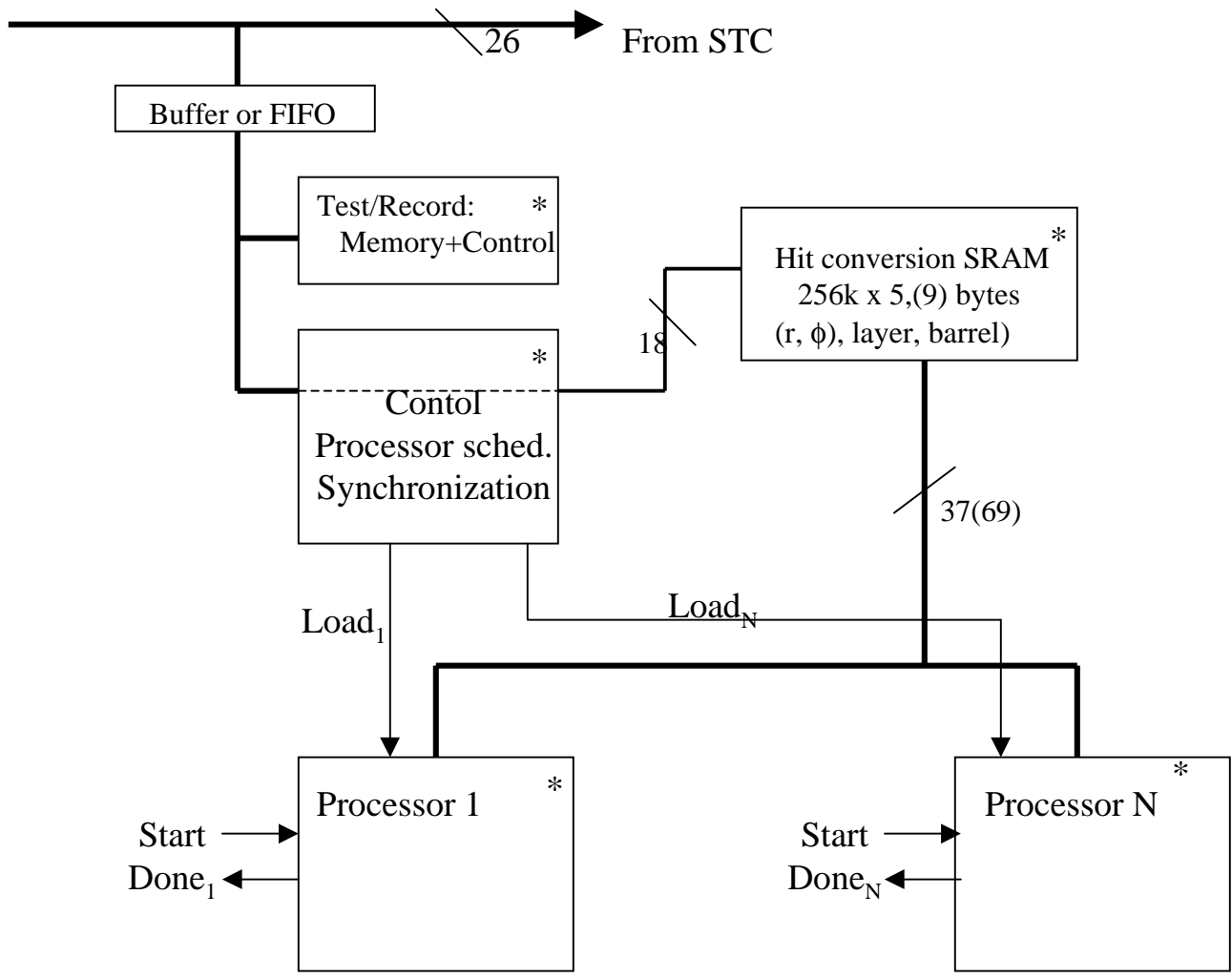
Processor	Time/Road
c6201A	400 μ s
c6201B	137 μ s
c67(FP)	14 μ s

- Space (9Ux400 mm)
 - Assume needs a square of 2x sides of package
 - 16 processors is approx **60% of board area**
- Power
 - TI literature, 2W/processor typical
 - I_{max} at 3.3V and 1.8V \Rightarrow 3W/processor
 \Rightarrow **35-50W/board** in processors
- cost. x67, TI web page, \$110-\$240/processor, 16 processor board fits in estimated \$7.5k + design costs, so \$50k/16 tracks fit + design

Fit Processors: Altera 10k (New information)

- Processing time
 - Floating point “unit” at ≈ 20 Mhz. (16,6) bit (mantissa,exponent), \$4k Integrated Silicon Solutions
 - Going to (24,8) w/Altera reduces frequency by approx. 20%*
 - Algorithm #1 200-250 flops, so $\approx 12.5 \mu/s$ road in floating point
 - * Precision? (certainly 8-bit exponent)
 - * Uses 6-10%(?) of 10k100. Parallel execution?
- Space. as above, 240 pin package, **60% of board space**
- Power. Use Altera’s formula. Assume 100 Mhz and chip 100% used gives 3W/chip. \Rightarrow **50W/board**
- Cost, \$170-\$230/chip

Again, no obvious problem, 1 board/16 tracks fit(power)
Beginning a pseudo-implmentation of algorithm #1



*VME access needed

Output Stage?

Comments on block diagram

- Try to have converted hit → processor on-the-fly
 - If DSP, will need RAM/FIFO at each processor
 - If Altera, arrange to avoid this?
 - Could add FIFO on input stage.
 - SRAM for conversion: 880k Channels/6 sectors x 4(granularity) x 9 bytes ⇒ < 2Mb/sector.
- Control Chip
 - Depends on STC format, especially road ID
 - Assume now, global ID (8 bits)
 - First pass at processor allocation (Altera practice)
 - * 20 Mhz for 64 tracks/sector (input → register set up, no pipelining, pin assignments)
 - * 40% of 10k50 logic
 - * trivial 3-stage pipeline
- Copy existing VME interface? Access to:
Processor internals, SRAM conversion table, Control Chip (diagnostics), test/record memory
- Output stage?

Track-Fitting Plans

- Short term

1. Floating point precision (JDH, by Paris)
2. Floating point performance (Wendy)
3. Control Chip, pipeline? (JDH, Wendy)

- By mid-summer

1. Number of tracks to fit
2. Altera as processor (JDH, Wendy)
3. Filtering algorithm (Under grad)
4. Timing on x67 (Onur Menten)

Global Issues and TFC

- STC transfer mechanism **backplane or ?**
- Road/track ID. **Local vs. Global**
- Power distribution. **Can I get 2.5V trivially?**
- Canned VME interface. **Where?**
- Output design
- Road width and beam spot

*Does the road width come from lifetime or **BEAM POSN**? If roads were given beam position, could they adjust fill-to-fill and be narrowed?*

Personnel

John Hobbs (30%)

Wendy Taylor, Post doc (100%)

Chuck Pancake, Engineer (5% until summer, ramping up after)

Dean Schamberger, consulting

e-shop for prototyping, layout, etc

