Off-Detector Processing for Phase II Track Trigger

Ulrich Heintz (*Brown University*)
for
U.H., M. Narain (*Brown U*)
M. Johnson, R. Lipton (*Fermilab*)
E. Hazen, S.X. Wu, (*Boston U*)

basic idea

- for each sector
  - represent every possible hit combination by a logic “equation”:
    $S_{1i} \cap S_{1o} \cap S_{2i} \cap S_{2o} \cap S_{3i} \cap S_{3o}$
  - create a table of all possible equations in FPGA
  - load all hits for an event into registers in FPGA
  - evaluate all equations simultaneously in one clock cycle
  - equations which are satisfied correspond to reconstructed tracks
  - timing dominated by time needed to load hits into FPGAs
- problem
  - if we tried to do all six layers at one time
  - too many equations
  - too many inputs
  - need to factor problem

sector size

- what is the minimum p_T we need to trigger on?
- tracks must be contained in 3 adjacent sectors
- sector size ←→ minimum p_T

min p_T	sector opening angle	
2 GeV	18°	←requires more/wider modules
2.5 GeV	15°	←lots of inputs but geometry easier
5 GeV	7.5°	←less inputs – makes trigger easier

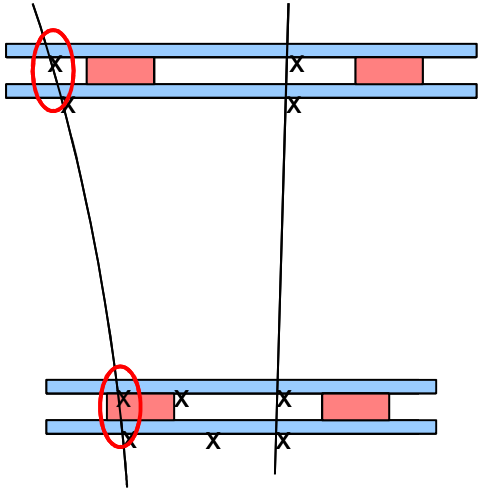
- built into hardware design
- need to decide “soon”

- → here assume 15° sectors

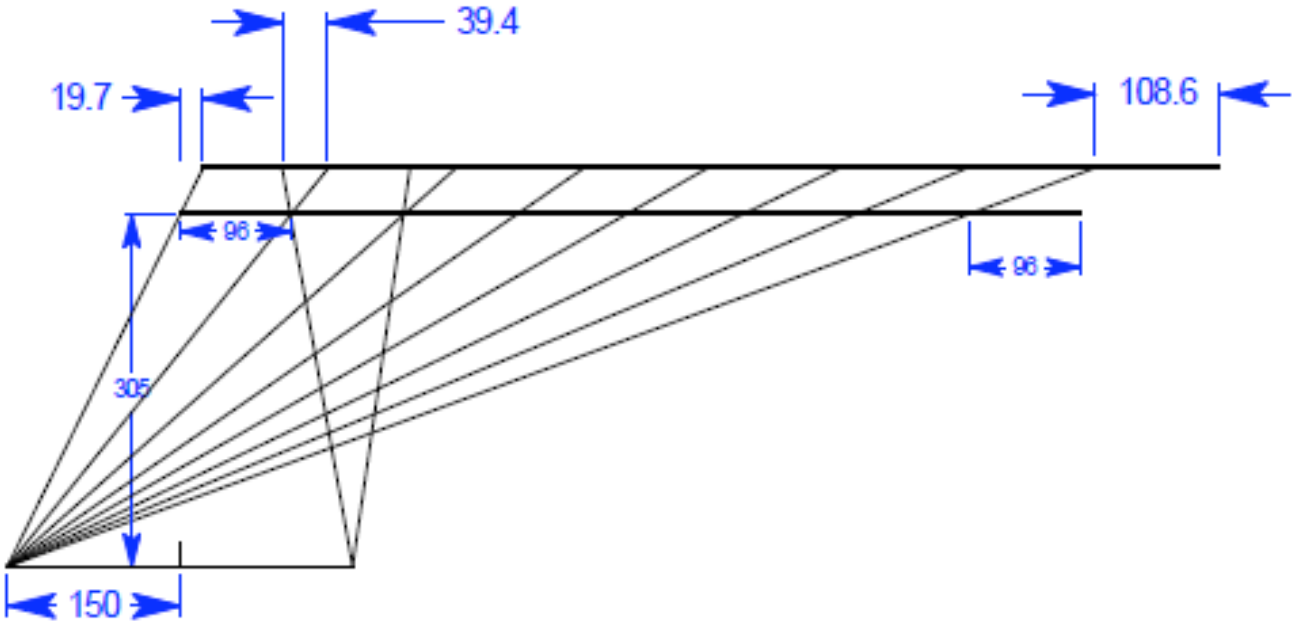
tracklets

- double stacks allow local track finding on each rod
- combine stubs from the two stacks in each station to form tracklets
  - drop in rate by about factor 4 (but need ≈30 bits/tracklet)
  - for each stack in inner layer need to process data from two adjacent stacks in z in outer layer

stub (2-layer coincidence)



Tracklet
(4-layer coincidence
with P_T validation)



tracklet inputs

max stub rate in simulation	0.3 MHz/cm^2
safety fudge factor (MC imperfect, stat fluctuations)	10
ave rate over z/max rate	50%
module area (inner station)	4200 cm^2
stub size	20 bits
data rate from one layer	125 Gb/s
link bandwidth	8 Gb/s
number of links/station	30

tracklet processing

- processing done off detector
  - input 30 fibers per station
  - fits into current FPGA (44 inputs at 10 Gb/s)
  - must do all processing in 25 ns
- local to each module
  - 42 modules in inner layer
  - need information from top and bottom stacks plus neighboring stack for z overlap
  - sensor overlap in ϕ so no rod to rod communication

tracklet processing

- data volume
  - 8 stubs/event/module @ 20 bits/stub ➔160 bits
  - fits into FPGA registers
  - compare all hits between stacks simultaneously
  - compare in both ϕ and z
    - z range restricted by length of IP
    - ϕ range restricted by min. p_T

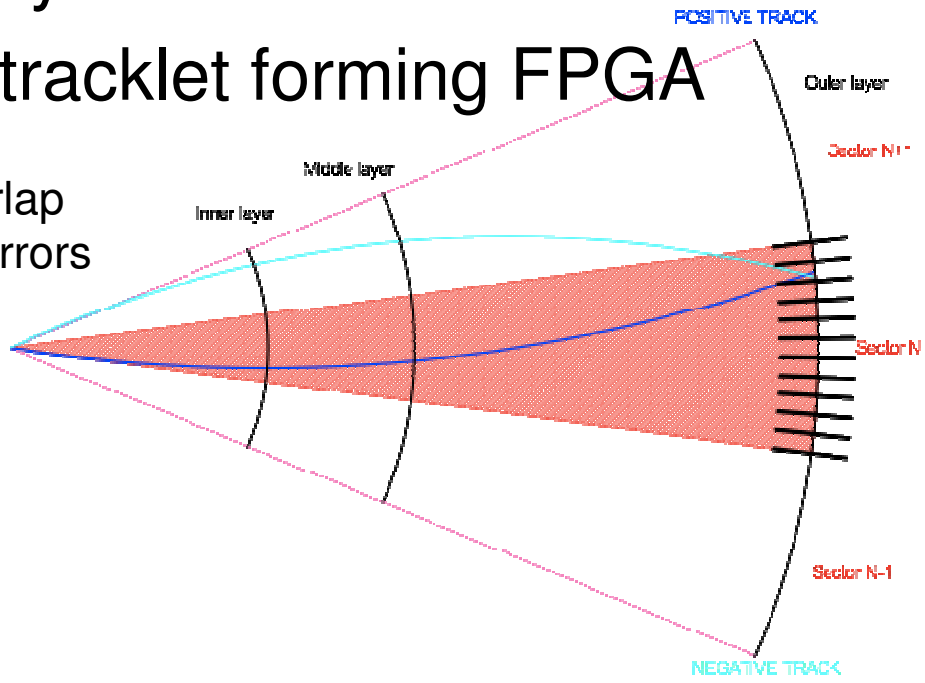
tracklet bandwidth

- simulation shows tracks are half the stub rate
- safety factor
  - 10 included fluctuations
  - should be less over rod ➔ use safety factor of 6
  - track density drops faster than stub density with z
  - use same density as for stubs (conservative)
- 40 tracklets/rod @ 30 bits/tracklet = 48 Gb/s

tracklet output

- sort tracklets by destination segment
  - send over dedicated fiber line
  - 12 segments/sector times 3 sectors = 36 fibers
    - 48 Gb/s divided among 36 fibers so 8 Gb/s fiber OK
  - project to 2 different layers
  - 72 output fibers from tracklet forming FPGA

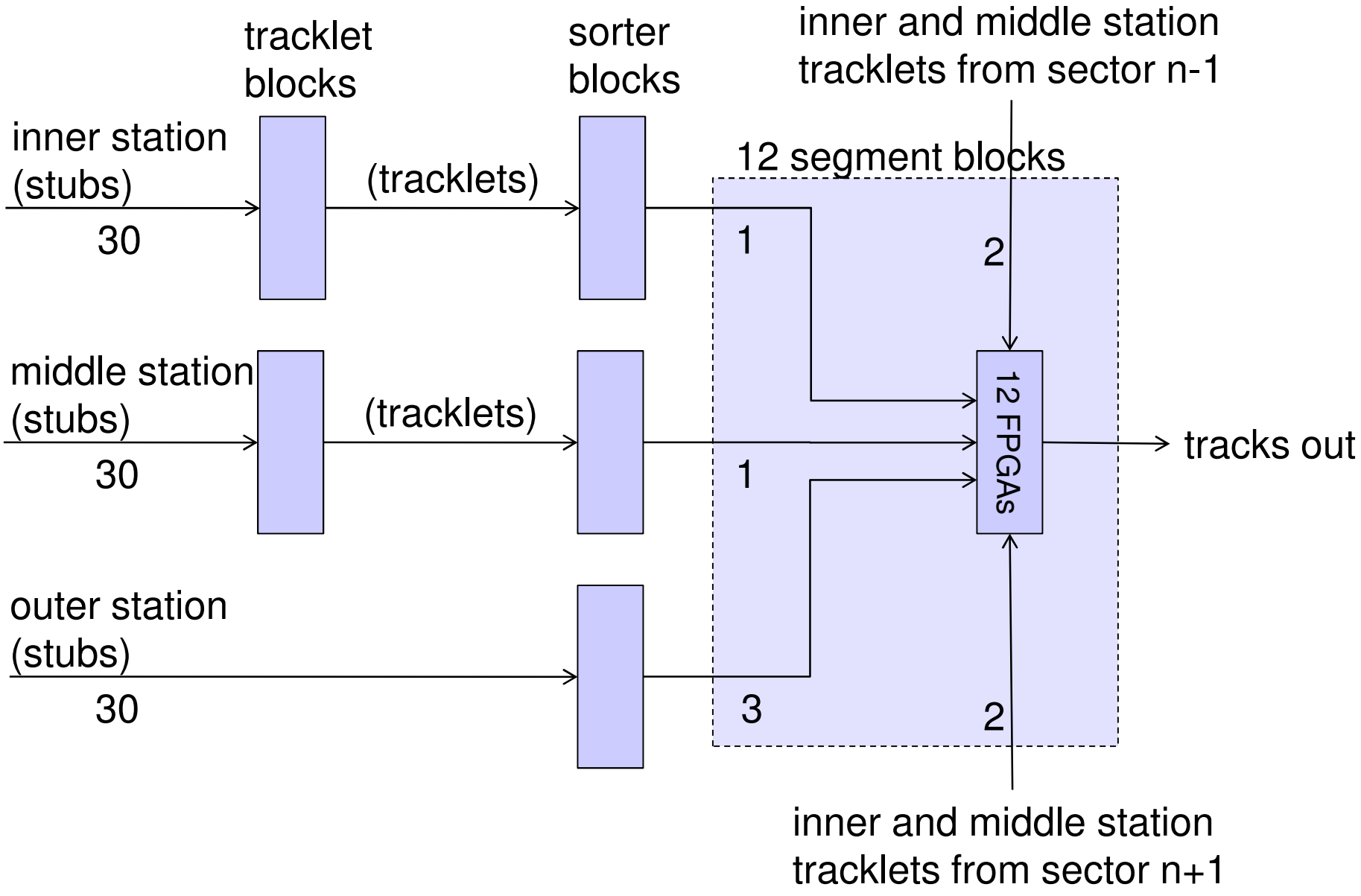
Segments must have overlap
to account for projection errors
(about 10% of segment)



segment processor

- receive tracklets from 2 stations in 3 sectors plus stubs from anchor layer in home sector
  - 1 fiber/rod/layer/sector for inner and middle layers
  - 3 rods times 3 sectors plus 3 for anchor station = 12 input fibers
  - plus 1 fiber for trigger output
- need to compare all possible combinations of input tracklets so need all tracklets in registers in FPGA
  - 3 times 40 tracklets/station = 120 tracklets in 12 segments
  - 10 tracklets/segment times 30 bits = 300 bits so OK
- output tracks ordered in p_T - highest first

sector processor



duplicate eliminator

- find tracks in all 3 layers simultaneously
  - ➔remove duplicate tracks
- receive data from 12 segments times 3 anchor stations or 36 fibers
- tracks ordered in p_T ➔ simplifies search
- output tracks on perhaps 4 fibers

pipeline stages in tracklet block

- (1) load stub data from sensors
- (2) form tracklets from stubs
- (3) sort tracklets by destination segment
- (4) transfer to segment processor

pipeline stages in segment processor

- (5) receive tracklet data from 3 stations
- (6) find tracks
- (7) check z consistency
- (8) transfer tracks to duplicate eliminator

pipeline stages duplicate eliminator

- (9) receive track data from all 3 layers
- (10) compare inputs, eliminate duplicates
- (11) send track data to L1 trigger

- $\approx$11 total pipe line stages for system
- $< 1\ \mu s$

FPGA capacity

- doubling FPGA capacity: 12 segments ➔ 6
- doubling I/O capacity: 44 lines ➔ 88
- tracklet processor
  - 6 segments requires only 36 input lines
  - tracklet input requires 30 lines and output 36 so it would fit
- segment processor and duplicate eliminator are not changed

trigger and download (unidirectional GBT)

- download is infrequent and slow
  - single twisted pair cable should work.
- trigger and clock are common to all sensors
  - optical link to tracker end
  - twisted pair on rod
  - could put additional link at mid point of rod if needed

tracker readout

- readout event rate is 0.25% of trigger rate at 100 kHz trigger rate
- factor of 10 larger event size would take 2.5% of the fiber band width
- put data on the fibers from the sensors and separate it out in the tracklet block

optical link power

- 30 fibers in inner station/sector
- about same number of tracks in all stations ➔ 90 fibers/sector
- 24 sectors ➔ 2160 fibers
- power ≈1 watt/driver ➔ 2 kW

system advantages

- robust
  - can lose a stub in any layer
  - still form track with middle layer tracklet and stubs in inner and outer layer.
- easy scaling with bigger gate arrays
- one sensor or chip failure does not disable the trigger in that part of a layer

system drawbacks

- very large number of high speed links
- many single fiber connections
  - reliability
  - mass
- problem is significantly more difficult if rate estimates are too low
  - get better estimates after LHC start up

Summary

- getting all required data into one place is an important constraint
- very large number of high speed links
  - reliability, power, mass
- not constrained by FPGA size
- 2 layer design is simpler but it still has many of the same issues, potentially not as robust
- verify rate estimates with LHC data
- need MC simulations implementing specific algorithms