Progress on the online tracking algorithm

Yutie Liang, Hua Ye, Martin Galuska, Jifeng Hu, Wolfgang Kühn,
Jens Sören Lange, David Münchow, Björn Spruck

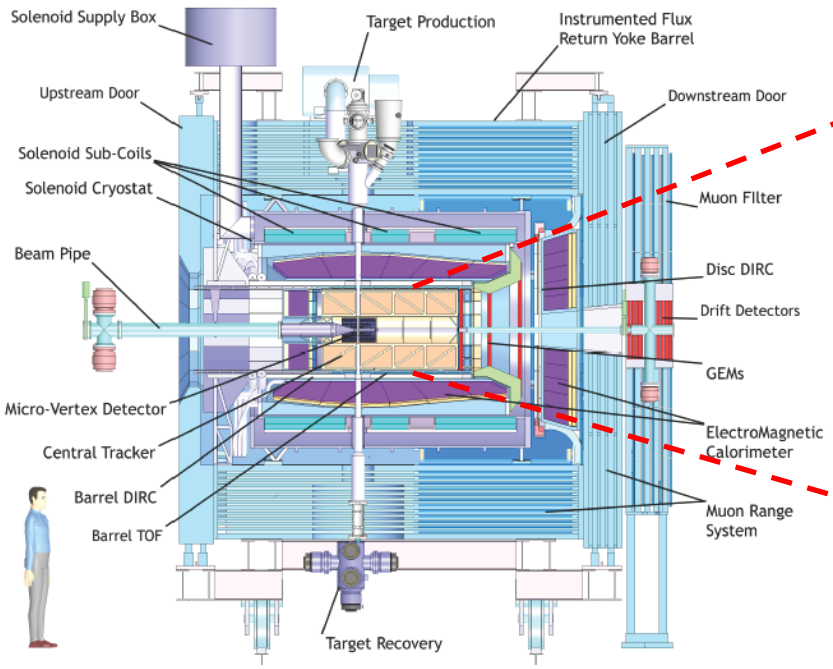
II. Physikalisches Institut, JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

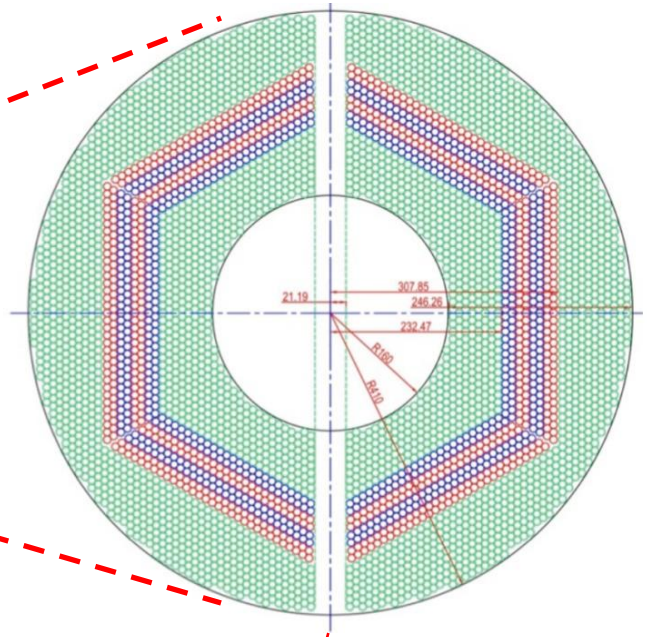
Jun. 25 2013

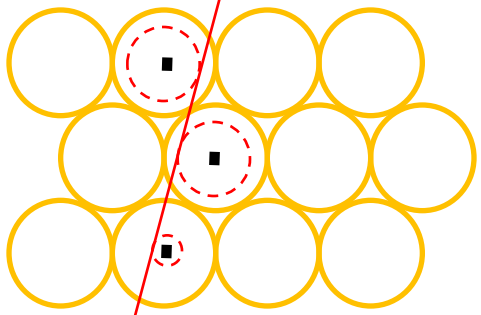
Outline

1. Introduction
   Straw Tube Tracker
   Tracking using conformal & hough transformation
   Adaptive design
2. Recent progress on the algorithm
3. VHDL implementation
4. Test on FPGA
5. Summary and outlook

Straw Tube Tracker(STT)



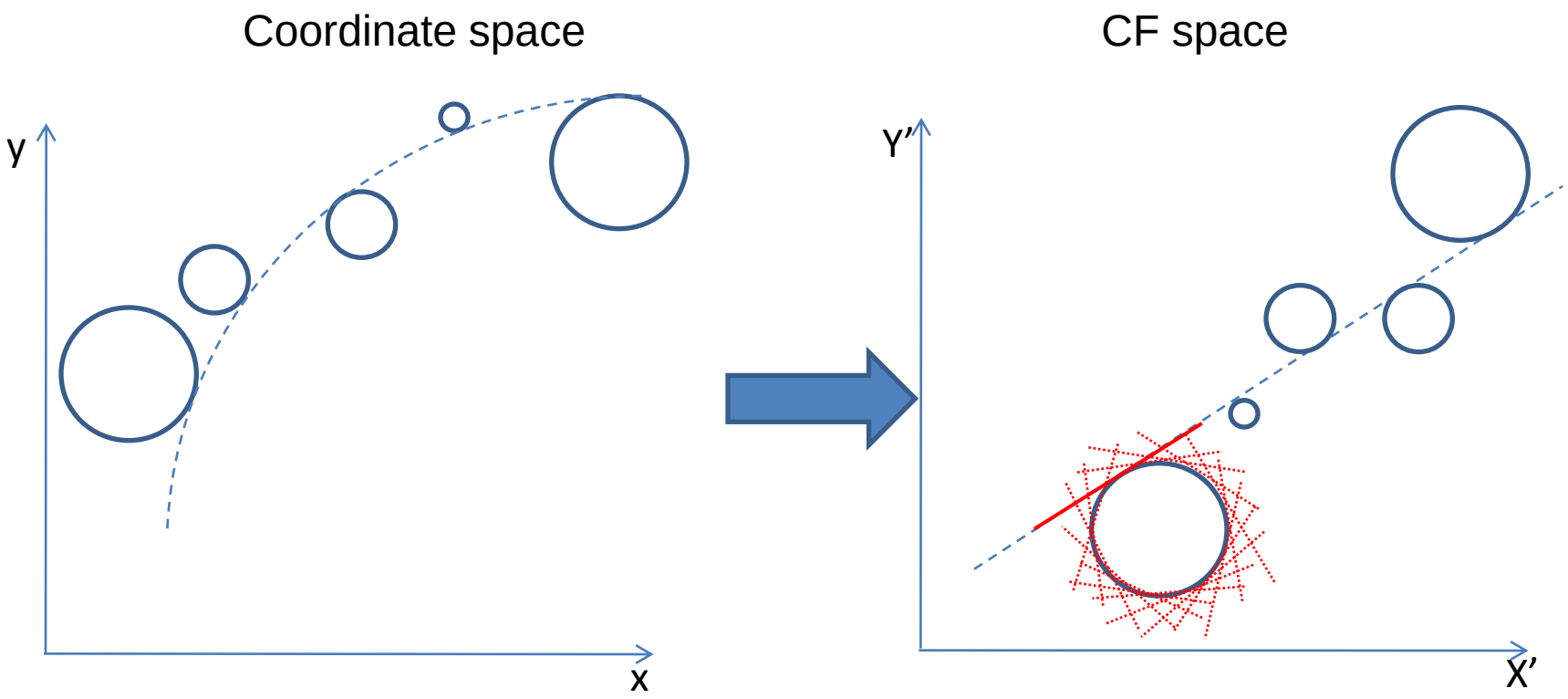




- ➢4636 Straw tubes
- ➢23-27 planar layers
- 15-19 axial layers(green) in beam direction
- 4 stereo double-layers for 3D reconstruction, with ±2.89 skew angle(blue/red)

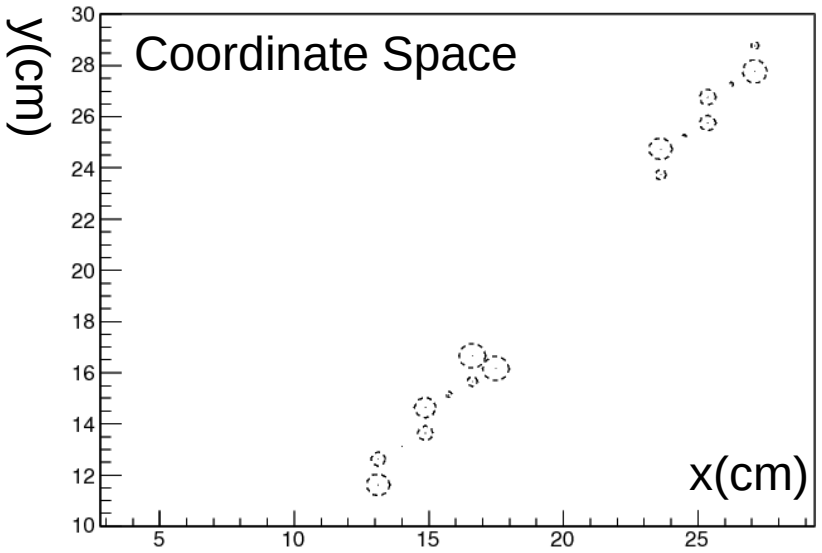
From STT : Wire position + drift time

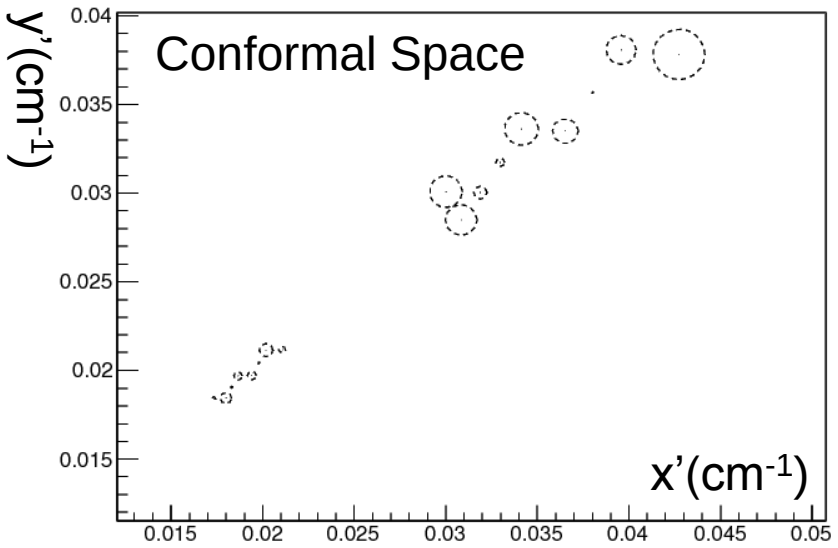
Method of Pt reconstruction with STT

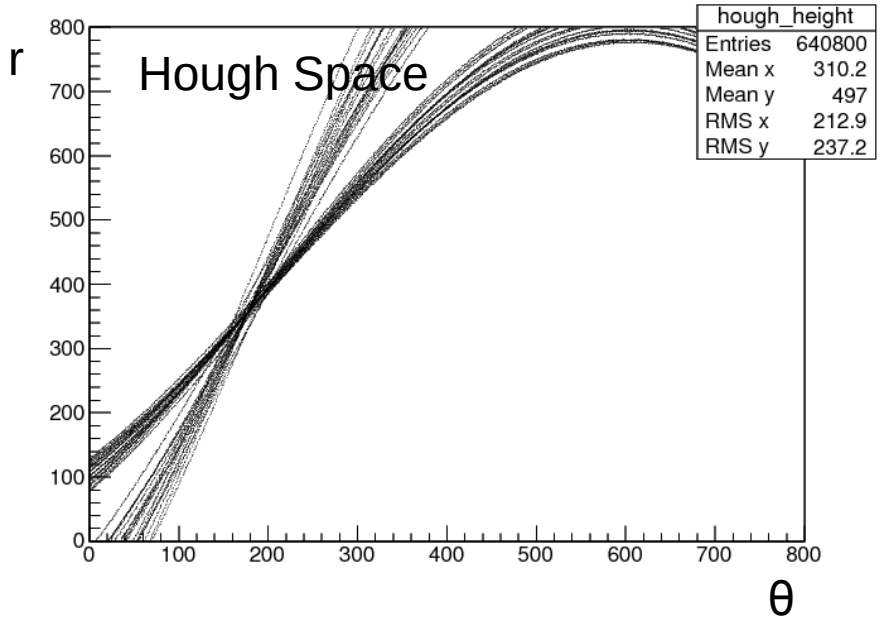


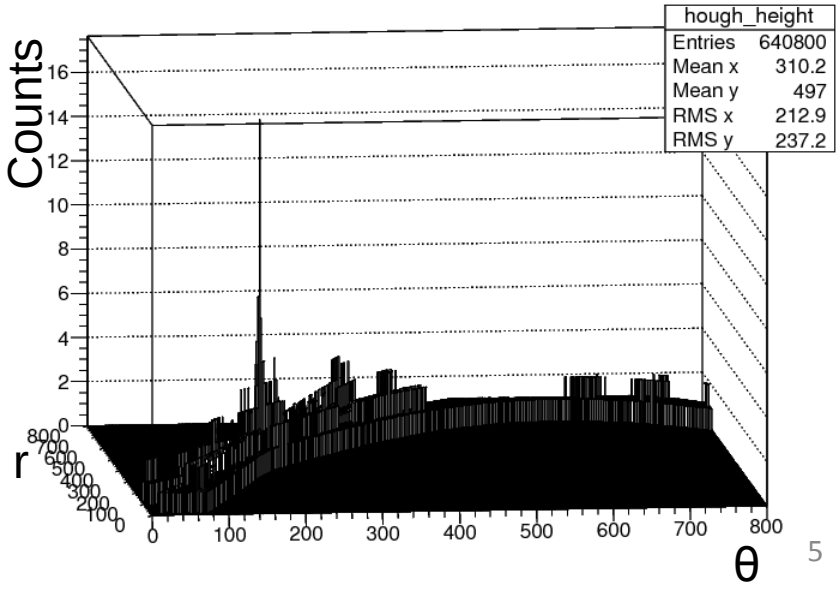
- Transform the drift circle to CF space.
- Draw lines around the “circle” in CF space.
- Fill the line parameter into the Hough space.

A look at one example event

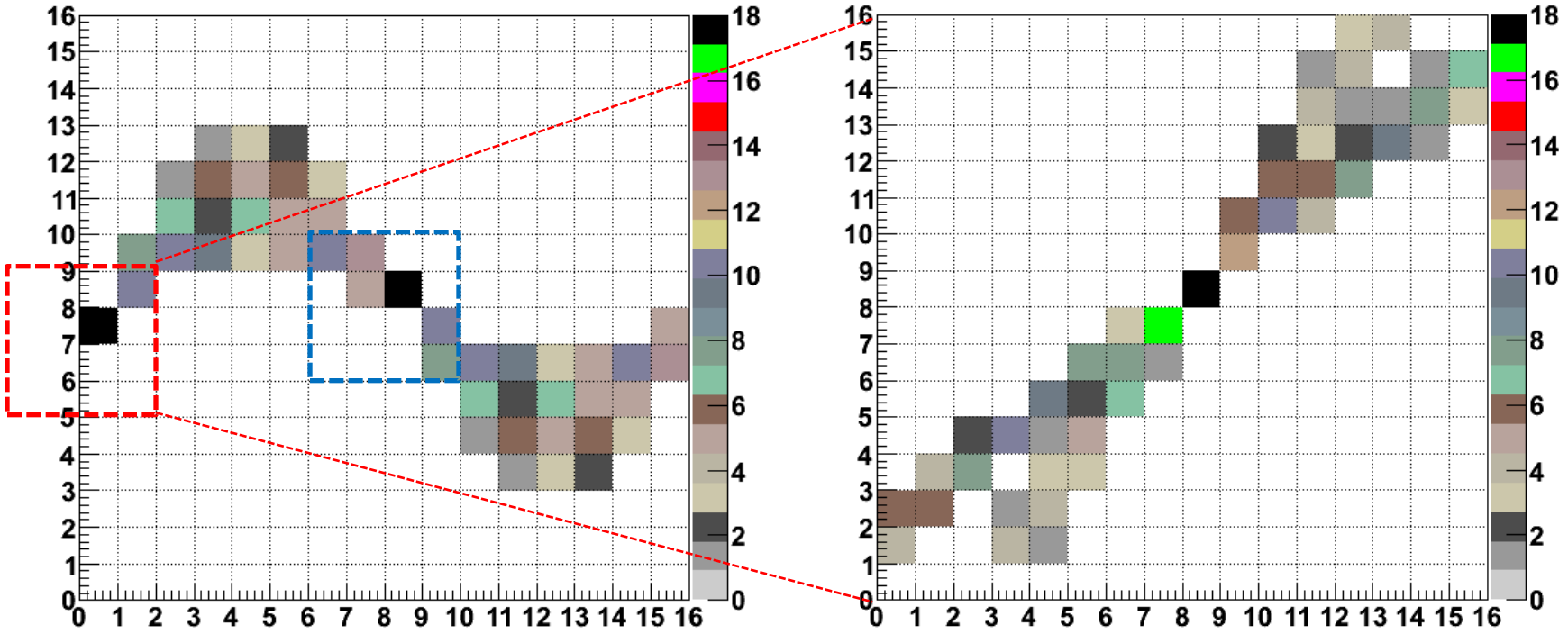








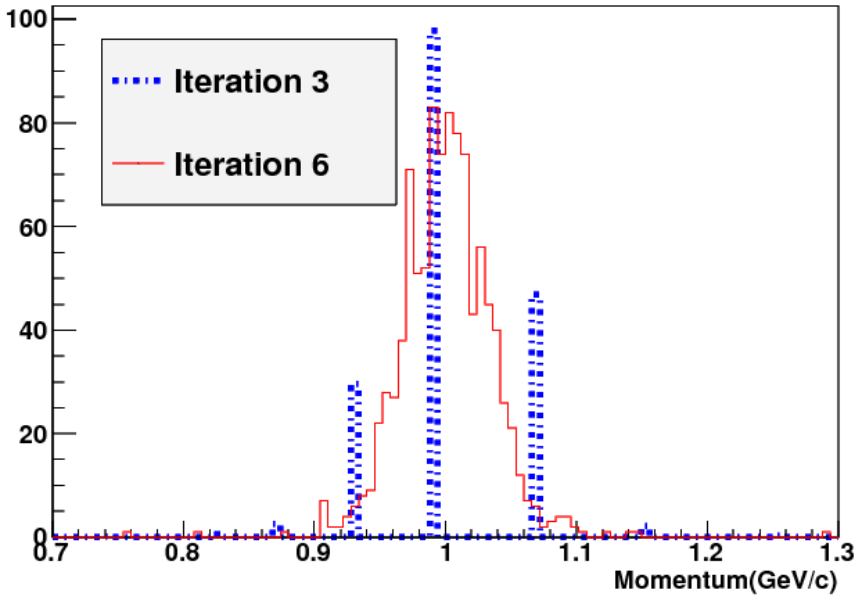
Adaptive Hough Transformation

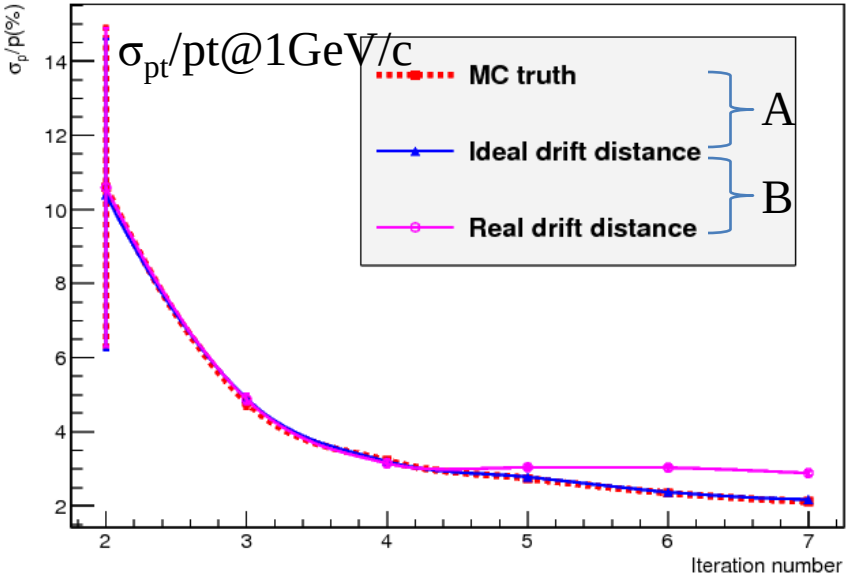


Iteration	0	1	2	3	4	5
Hough space	16X16	64X64	256X256	1024X1024	4096X4096	16384X16384

Performance Study

1000 events of single μ track with pt @ 1GeV/c





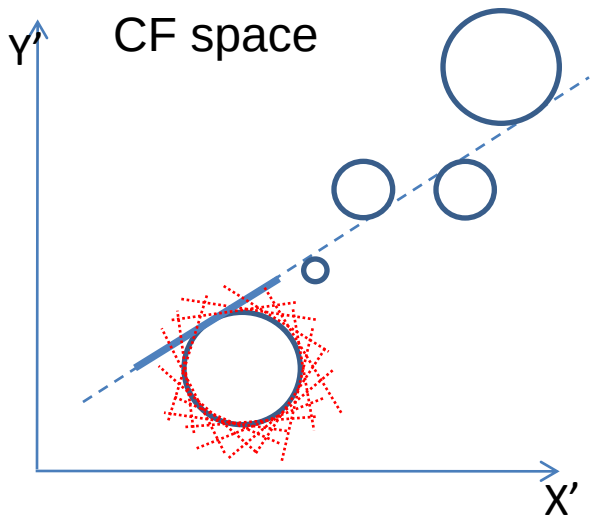
A: Using drift circle is as good as using MC truth point.

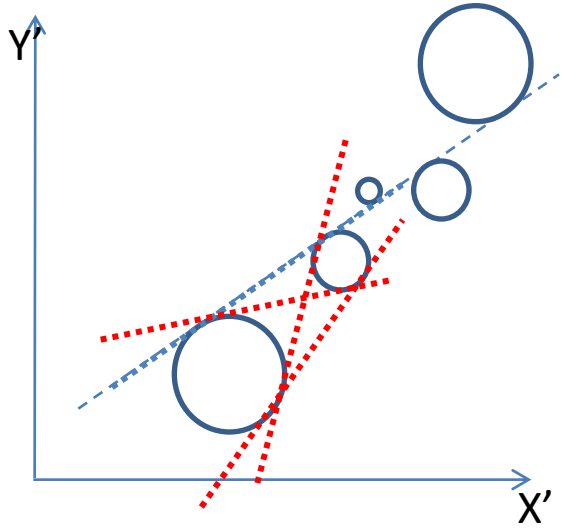
B: No improvement after iteration 4, using realistic drift circle.

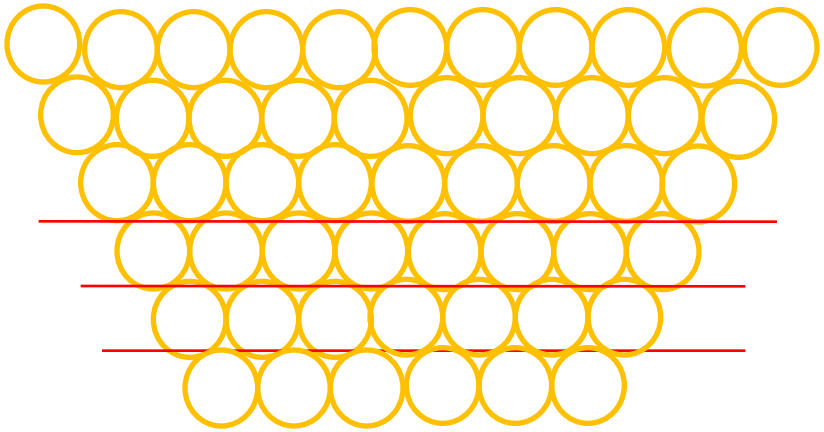
Tracking efficiency: ε ~ 99%

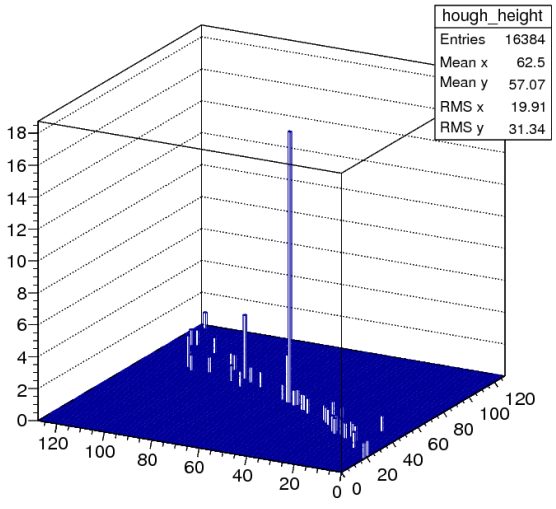
* Realistic drift distance: fast approximation from the true distance to the reconstructed one, using Juelich exp curve COSY-TOF

Improvement on the algorithm







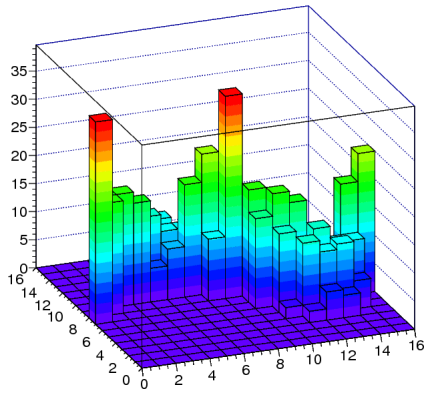


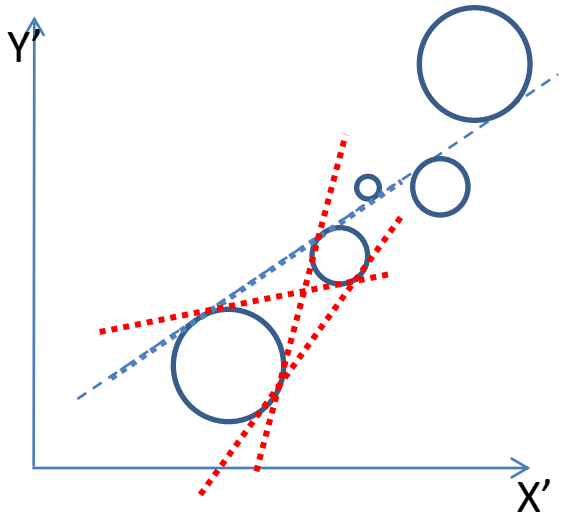
Comparison of new & previous algorithm

1: New method has less lines to be dealt with.
If 100 hits, assume we divide the detector into 10 layers: 10 hits/per layer
10 hits in layer N need to combine with the other 10 hits in layer N+1:
(10 * 10)combinations * 4 lines/ combination * 10 layers = 4000 lines

Previous method:
Iteration 1: 100 hit * 16 lines/hit = 1600
Iteration 2: 100 hit * 16 lines/hit * 16 (peaks from iteration 1) = 25600
Iteration 3: same as iteration 2…

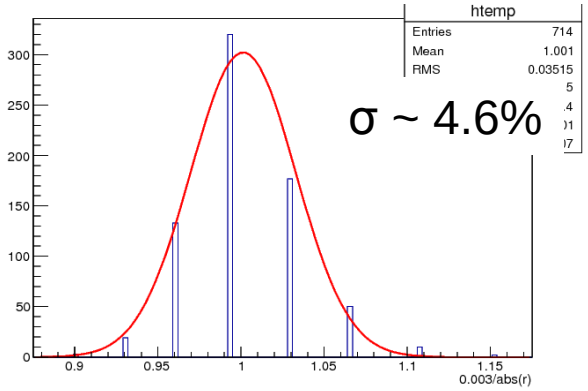
2: No adaptive design is required.



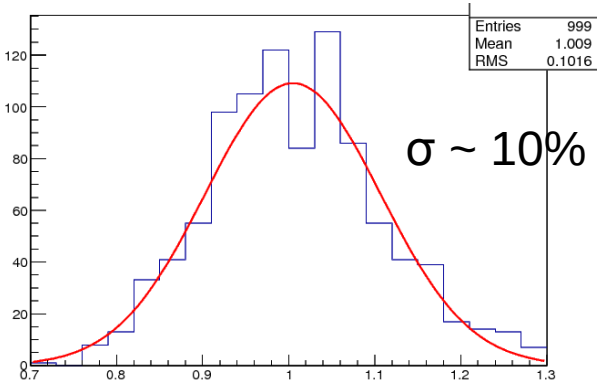


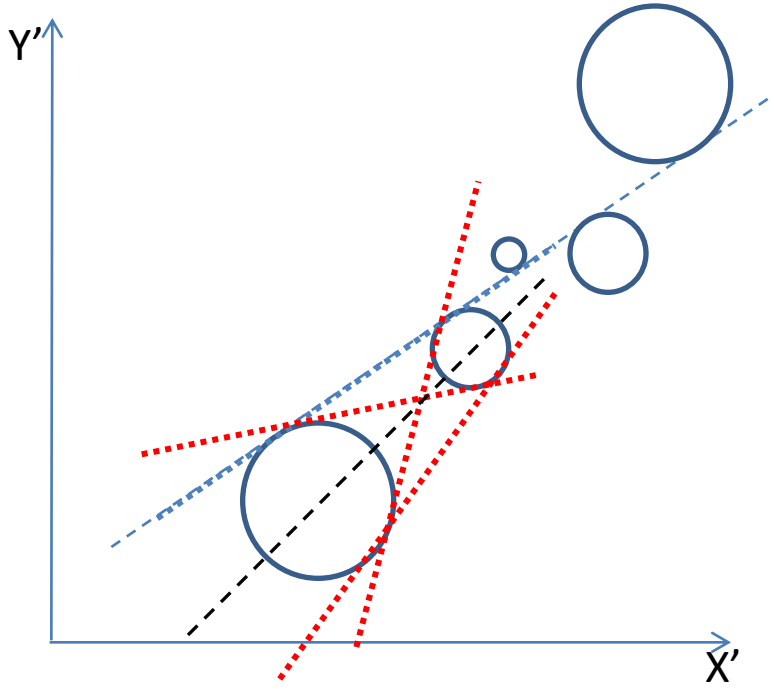
Two schemes

✓ Scheme I (Full): Four lines tangent to two circles.



✓ Scheme II (Simply): One line connects centers of two circles.

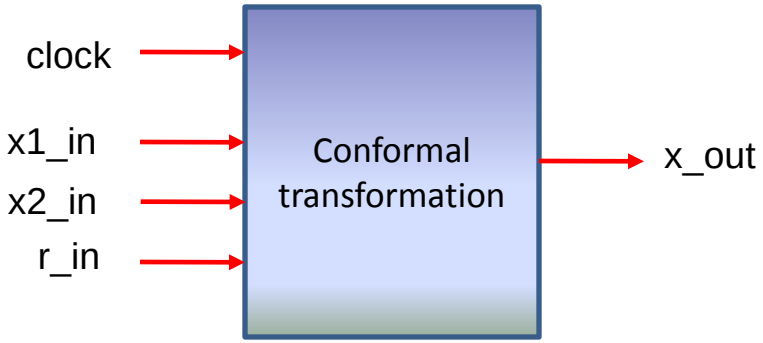




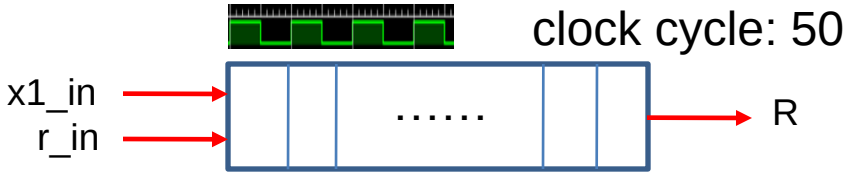
For the detailed performance study of the new algorithm, please see the next talk, given by Hua Ye.

VHDL coding, different from C++

VHDL: Very-high-speed integrated circuits Hardware Description Language.
It's a dataflow language, unlike procedural computing languages such as C++ which runs sequentially, one instruction at a time.




```
R       = x1_in/r_in ;
x_out   = x2_in + R  ;
```



Divider: A 32 bits divided by 16 bits has latency ~ 50 clock cycles


```
R       <= x1_in/r_in ;
x_out   <= x2_delay + R  ;
```

Tracking algorithm in VHDL

24 bits fixed point number: 000000011010100101110010

first 8 bits for the integer part (1 sign bit), last 16 bits for the fractional part

A line of 96 bits represents one hit: pos_x, pos_y, pos_z, drift distance

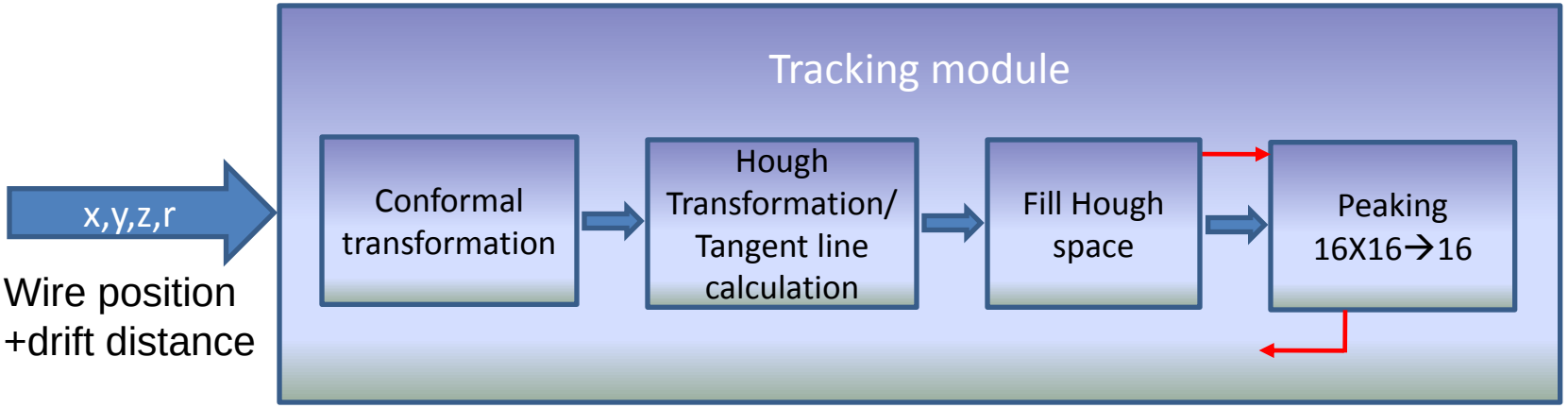
……

000000011010100101110010000000000000110011101101000000100000000000000000000000000000011001101011

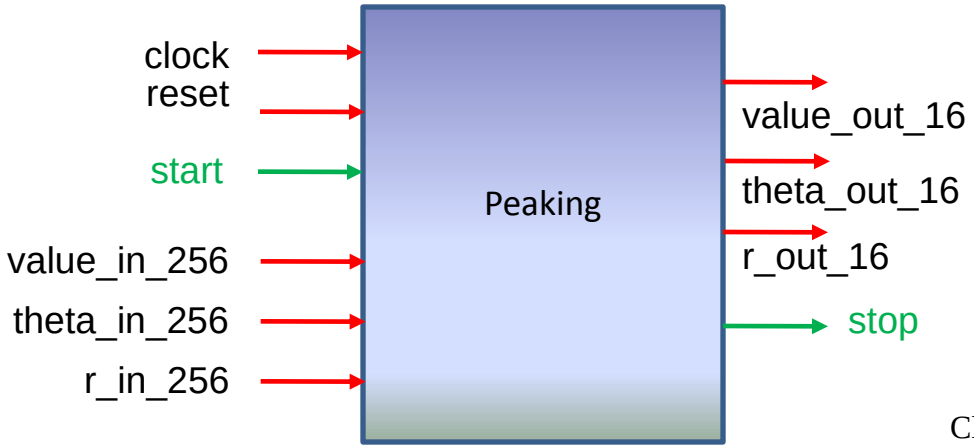
(1.6619, 0.0505, 2, 0.0250782) Unit (0.1m)

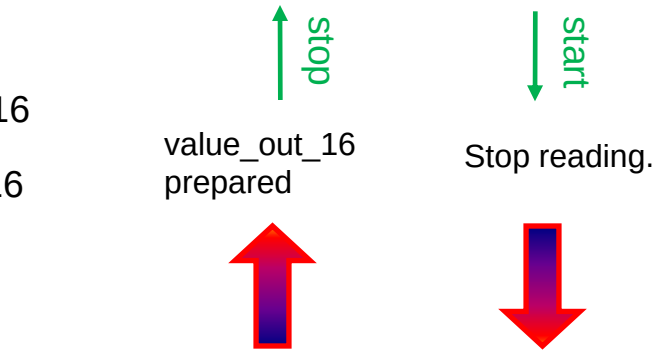
……

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000

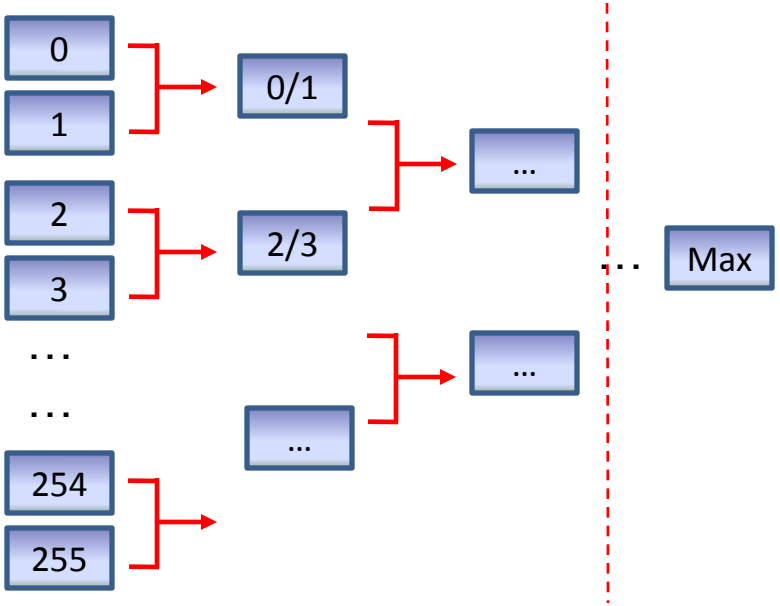


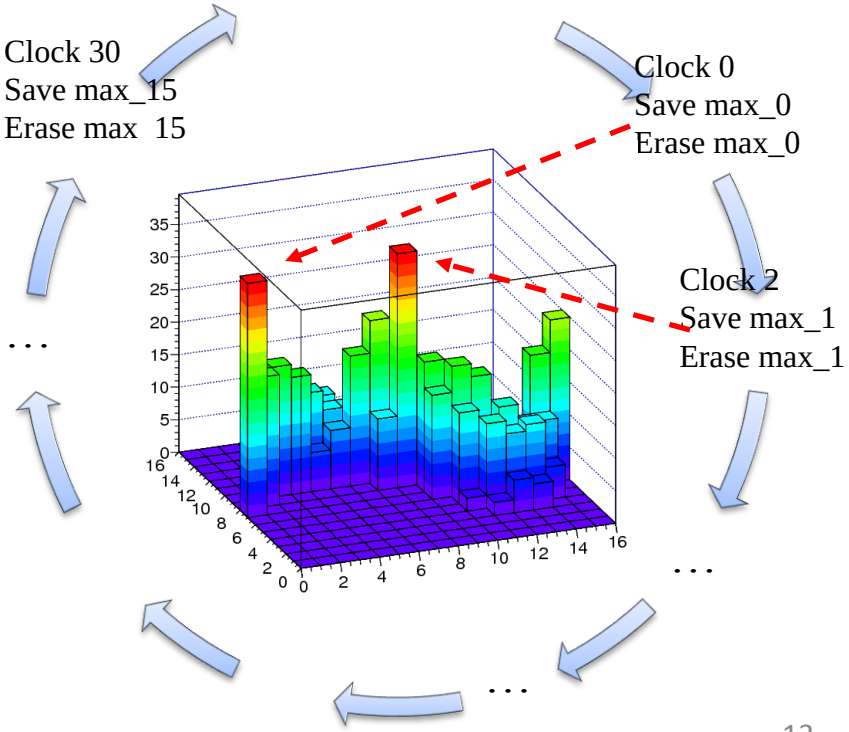
Peaking module





2 clock cycles to locate one maximum

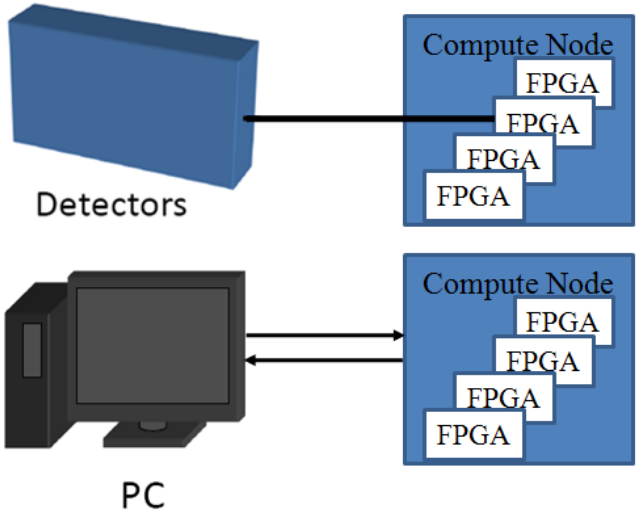


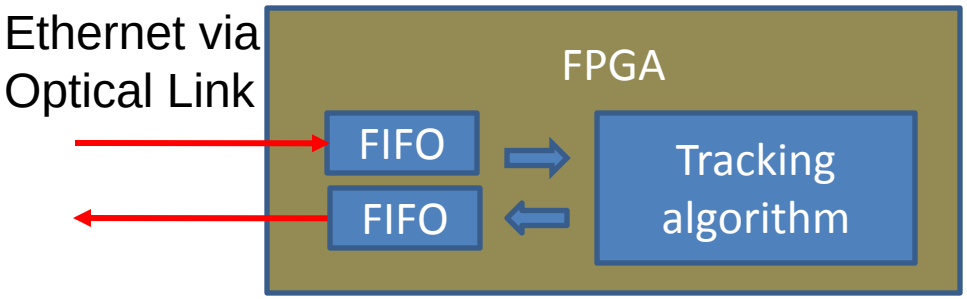


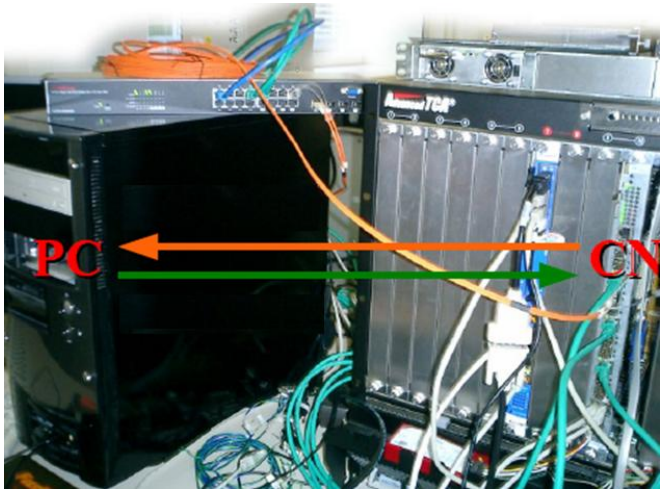
Setup and test

PC as data source and receiver.

- Ethernet.
- Optical link (UDP by Grzegorz Korcyl) (not integrated yet)







Data Check (Scheme II)

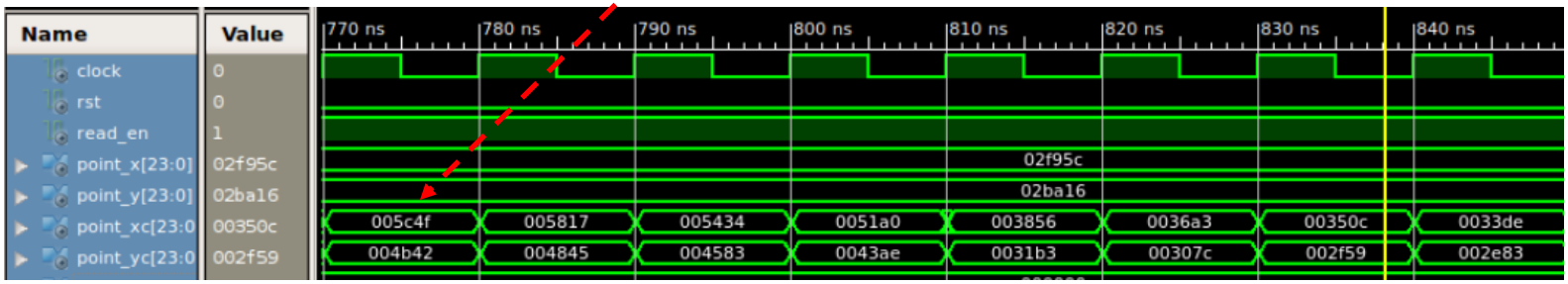
- Input package: 8-bit binary data
  00000000
  Flag bits: Bit 0,1
  01: head 00: body
  10: tail 11: invalid

- Output package: 8-bit binary data

- Results:
  Pt input : 1GeV/c
  Pt output: 0.9788 GeV/c

```
[stt@pchad1 ~]$ hexdump -C binary_1.bin
00000000  40 16 19 05 00 11 31 2a  00 20 00 00 00 00 03 2d
00000010  00 17 32 2a 00 12 25 18  00 20 00 00 00 00 18 2d
00000020  00 19 0c 0d 00 13 19 05  00 20 00 00 00 00 2e 24
00000030  00 19 0c 0d 00 15 00 20  00 20 00 00 00 00 1d 34
00000040  00 1a 25 32 00 15 34 0f  00 20 00 00 00 00 0a 3e
00000050  00 1b 3f 16 00 16 27 3b  00 20 00 00 00 00 0f 07
00000060  00 1d 18 3b 00 17 1b 29  00 20 00 00 00 00 29 13
00000070  00 1d 18 3b 00 19 03 04  00 20 00 00 00 00 1f 3b
00000080  00 28 25 1d 00 24 17 02  00 20 00 00 00 00 19 1a
00000090  00 29 3f 02 00 25 0a 30  00 20 00 00 00 00 05 1b
000000a0  00 2b 18 26 00 25 3e 1d  00 20 00 00 00 00 24 17
000000b0  00 2b 18 26 00 27 25 38  00 20 00 00 00 00 20 00
000000c0  00 2c 32 0a 00 28 19 26  00 20 00 00 00 00 01 20
000000d0  00 2e 0b 2f 00 29 0d 13  00 20 00 00 00 00 1e 32
000000e0  00 2e 0b 2f 00 2a 34 2f  00 20 00 00 00 00 23 00
000000f0  00 2f 25 13 00 2b 28 1c  00 20 00 00 00 00 02 82
00000100
```

```
00000000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000010  5c 58 54 51 29 36 35 33  32 43 30 2e 2e 2e 2e 2e
00000020  4f 17 34 a0 e7 a3 0c de  6d 03 03 c5 c5 c5 c5 c5
00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
*
00000050  0d fd 8b 8b 4c 8b 8b 8b  7b a7 61 7b 7b 7b 7b 1e
00000060  00 00 e5 00 20 00 00 00  00 00 00 00 00 00 00 00
00000070  00 02 4f 08 08 08 08 05  07 07 07 07 07 07 07 07
00000080  33 71 a5 23 0e 0b 09 f2  da d9 d9 d9 d9 d9 d9 d9
00000090  aa 05 7b 00 00 00 00 00  27 3b 00 00 00 00 00 00
```



Device utilization Summary

Scheme II

Latency: ~ 100 clock cycle

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	14,976	50,560	29%	
DCM autocalibration logic	14	14,976	1%	
Number of 4 input LUTs	13,610	50,560	26%	
DCM autocalibration logic	8	13,610	1%	
Number of occupied Slices	13,450	25,280	53%	
Number of Slices containing only related logic	13,450	13,450	100%	
Number of Slices containing unrelated logic	0	13,450	0%	
Total Number of 4 input LUTs	16,365	50,560	32%	

PANDA@20MHz. $2*10^7$ events/second

1 event: ~3 tracks/event * ~16 hits/track * ~ 2 (overlap factor) → ~100 hits/event

When dividing STT into 16 layers, ~6 hits/layers. 6*6*15 = 540 combinations

~1~2 clock cycles/combination. → 500~1000 clock cycles/event

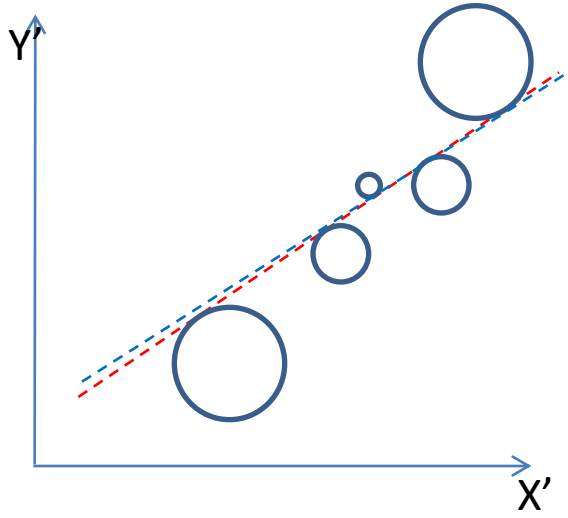
If FPGA running at 100MHz, (500~1000)*10ns/ 50ns → 100~200 FPGA → 25~50 CN

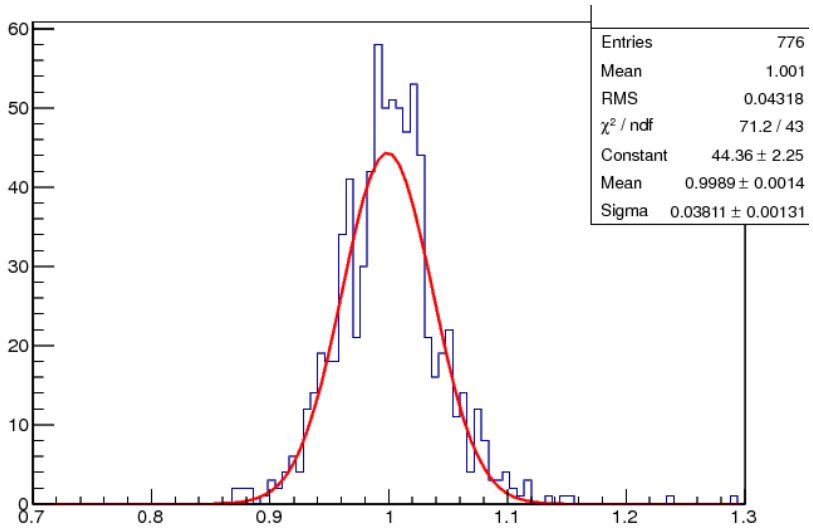
Summary

- Algorithm upgraded: four lines tangent to a pair of drift circles.
- Very simplified VHDL code implemented and tested in FPGA.
- Many designs tried.

Next to do:

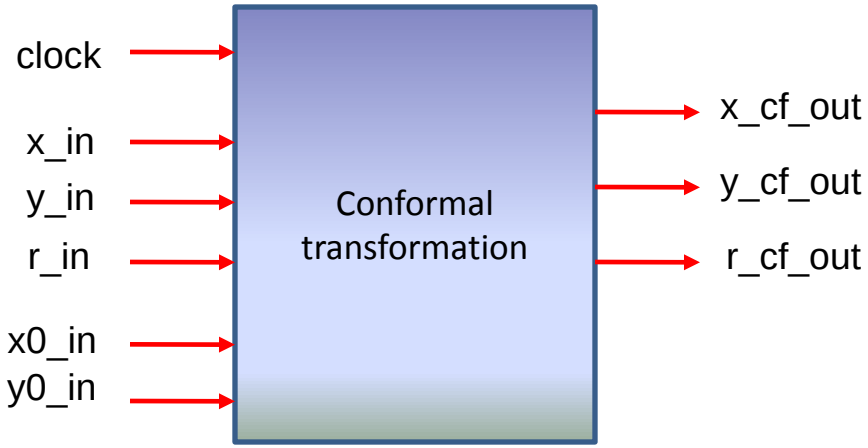
- A LOT to do.





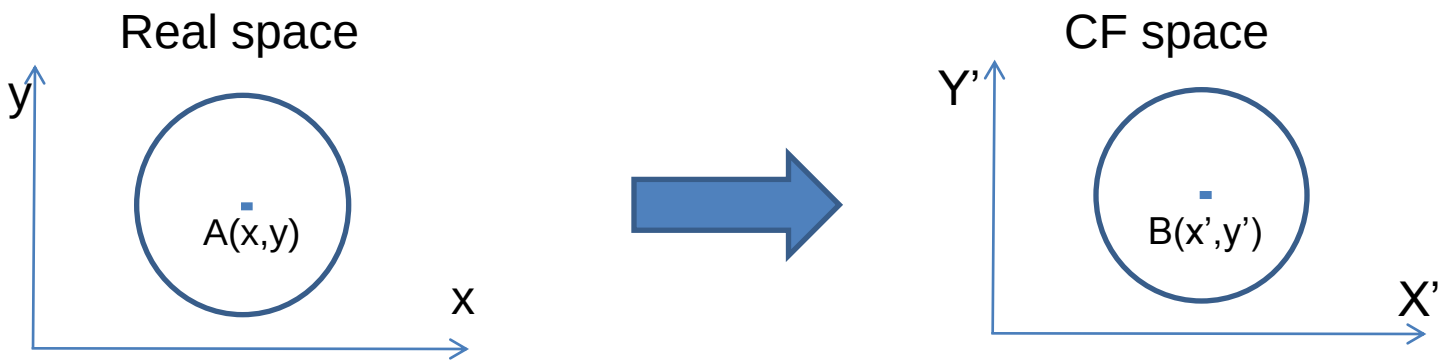
Thank you

Conformal transformation module

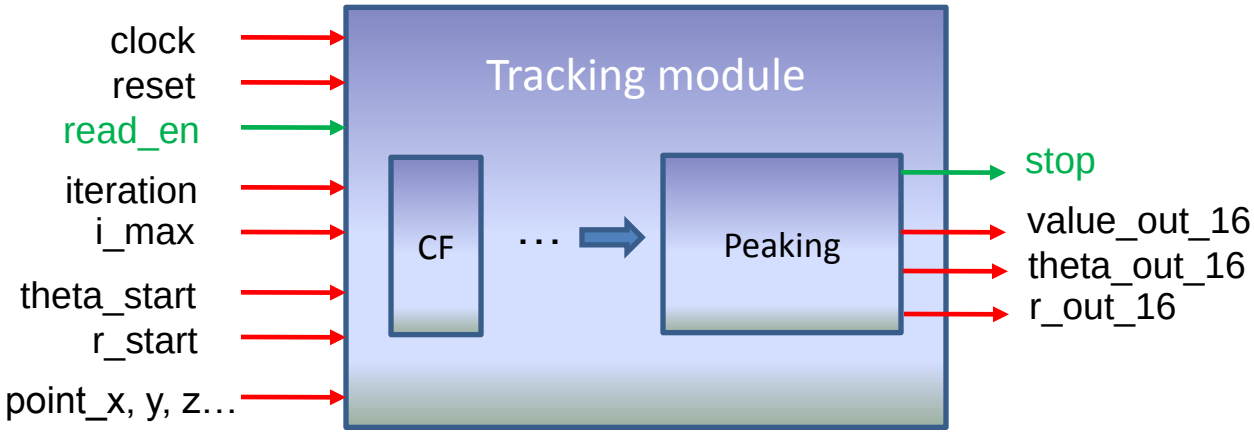


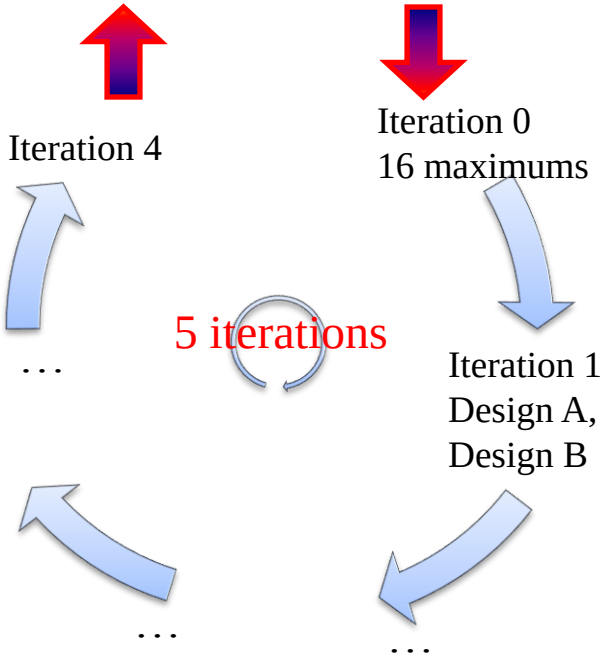
$R2 = (x\_in-x0\_in)^2 + (y\_in-y0\_in)^2-r\_in^2$

$x\_cf\_out = (x\_in-x0\_in)/R2 + x0\_in$
$y\_cf\_out = (y\_in-y0\_in)/R2 + y0\_in$
$r\_cf\_out = r\_in /R2$

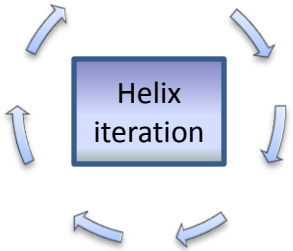


Adaptive design of tracking module

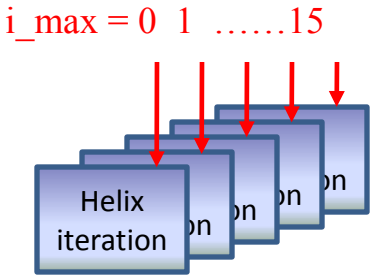




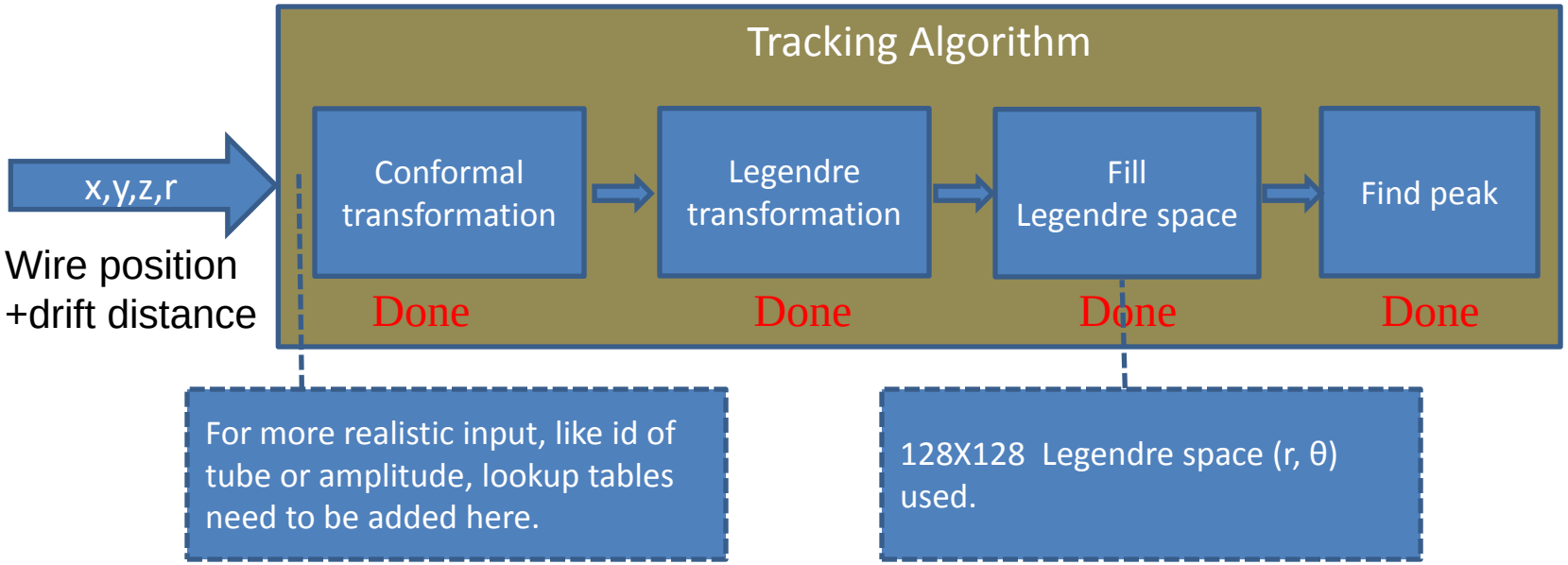
Design A:
Only one module, change i_max every loop



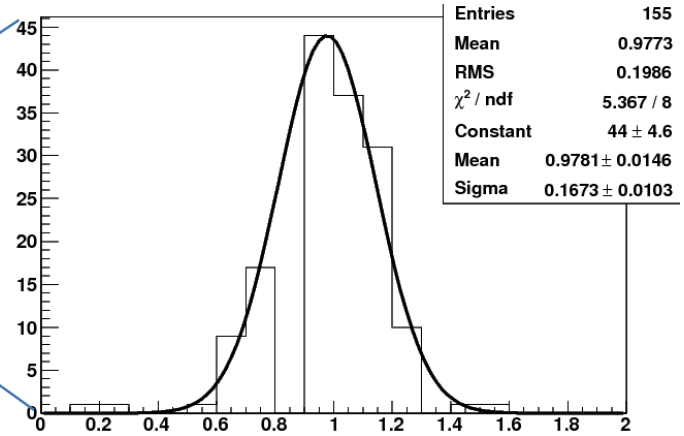
Design B:
16 modules, paralized



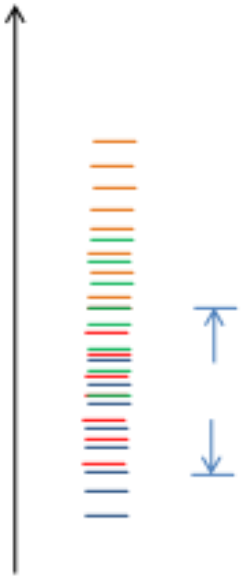
Status of the tracking algorithm in VHDL



Simulation with ISim

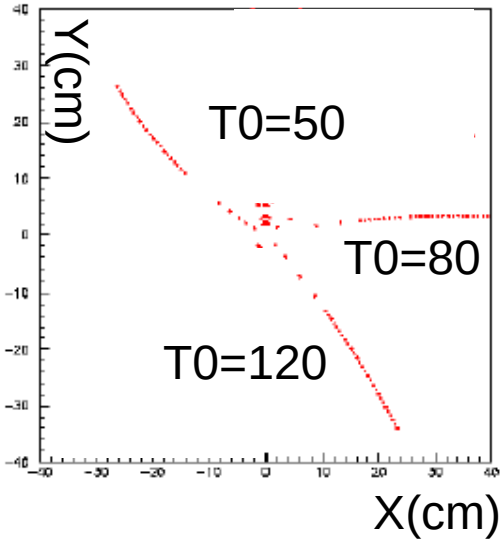


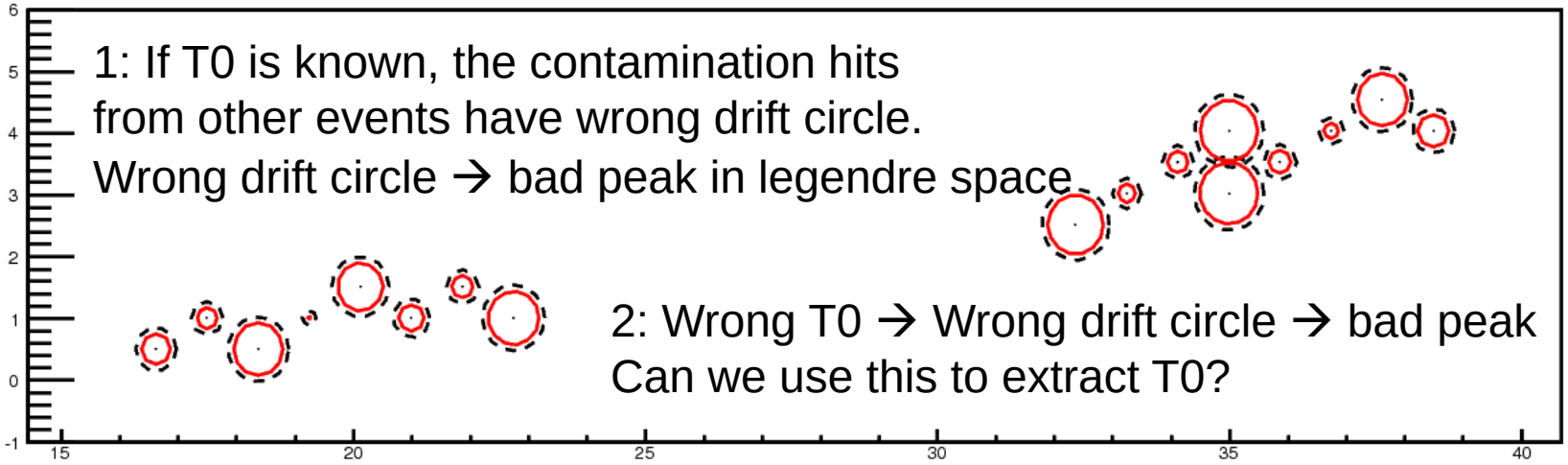
Can we extract the event start time?



- Need fast detector to provide T0.
- With T0, are the overlapped hits problem?

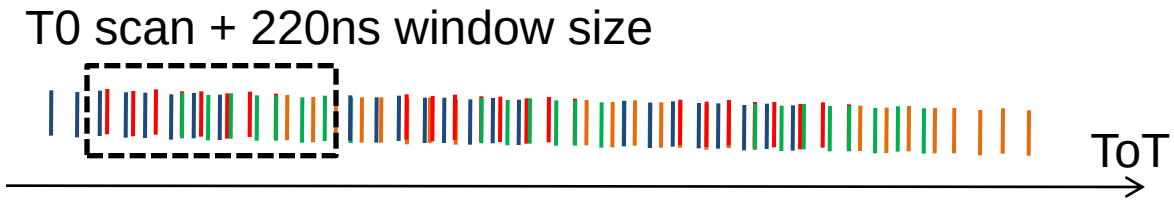
T0 + 220ns window size



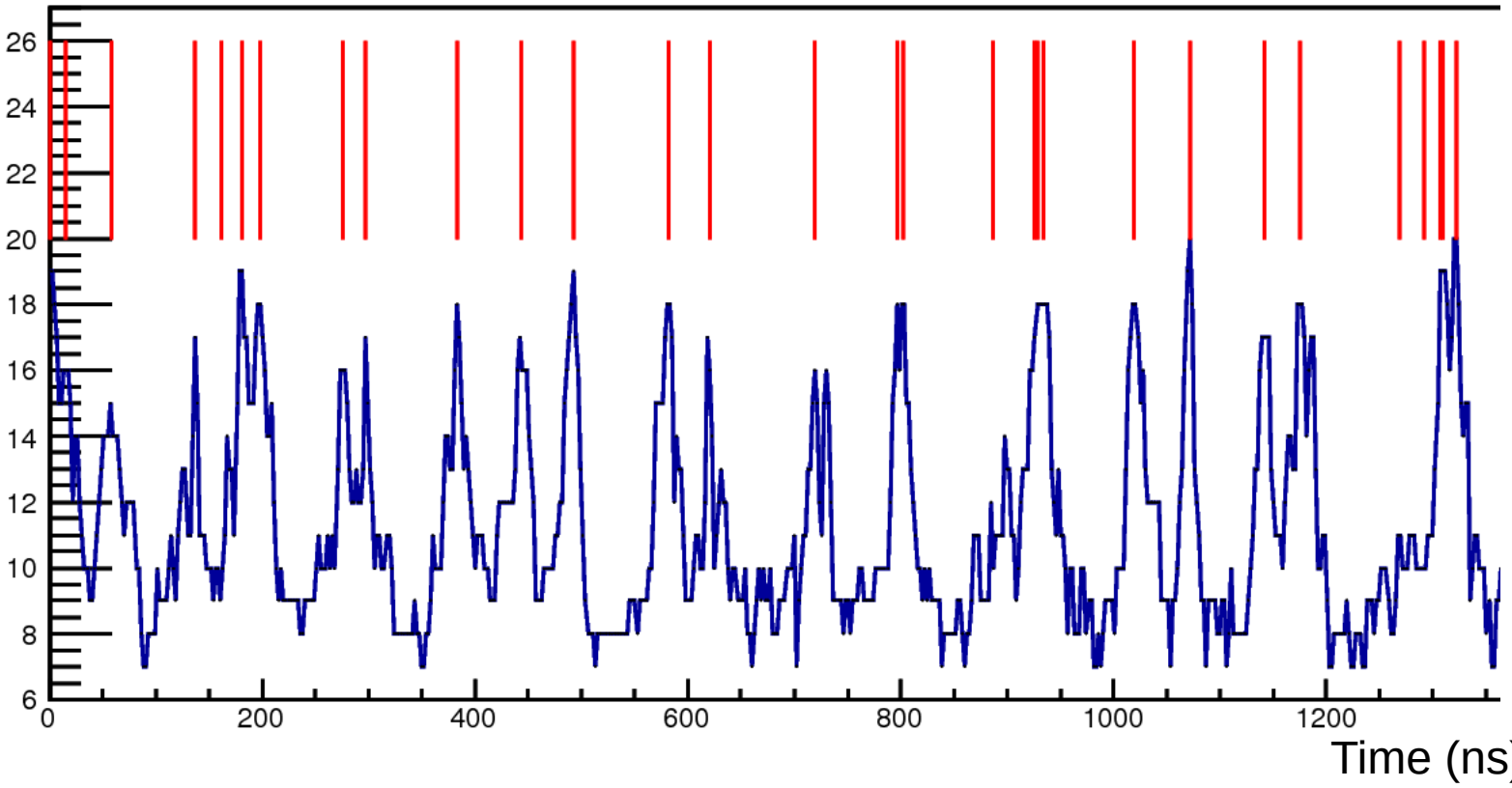


T0 extraction — for large burst

If large burst:



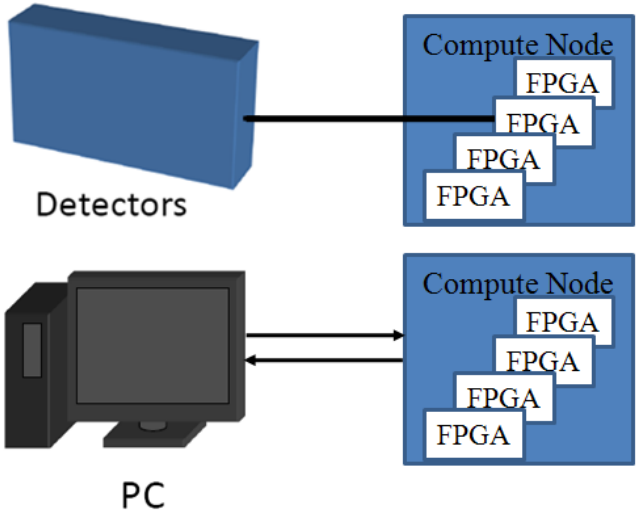
Time based simulation of one burst with 50 events

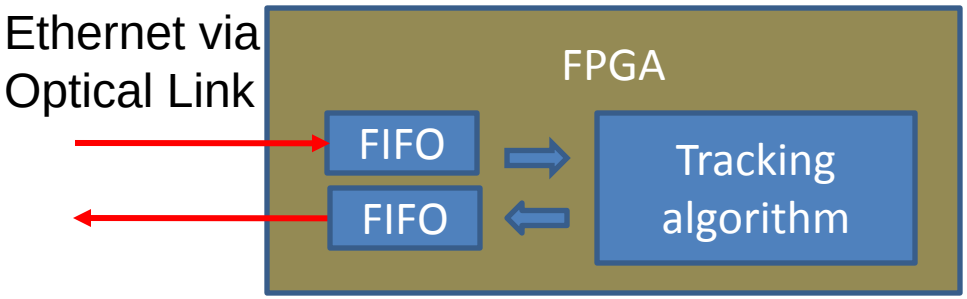


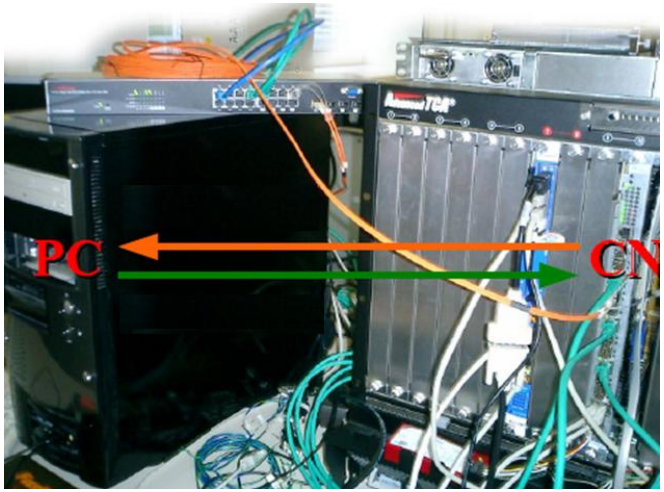
Setup and test

PC as data source and receiver.

- Ethernet.
- Optical link (UDP by Grzegorz Korcyl) (not integrated yet)







Conformal transformation and Hough transformation

Transform circles to straight lines

$$x' = \frac{x - x_0}{r^2}$$

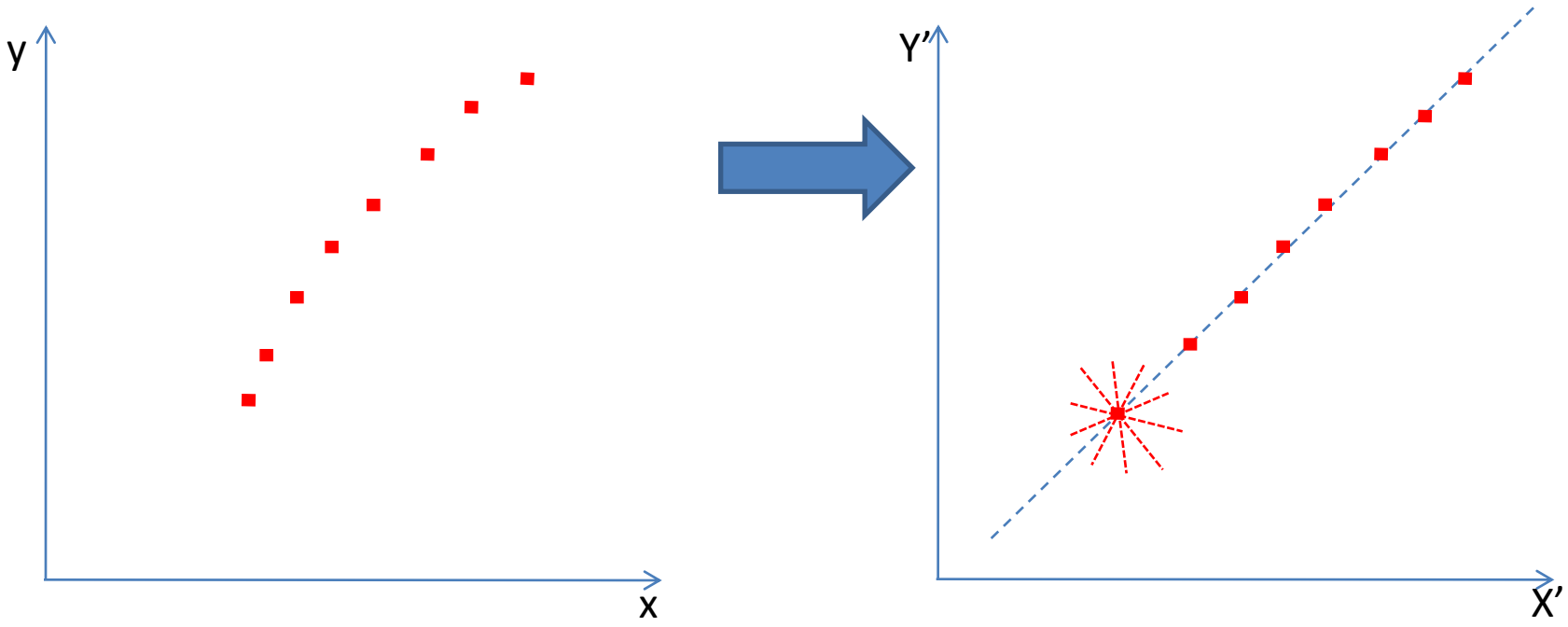
$$y' = \frac{y - y_0}{r^2}$$

$$r^2 = (x - x_0)^2 + (y - y_0)^2$$

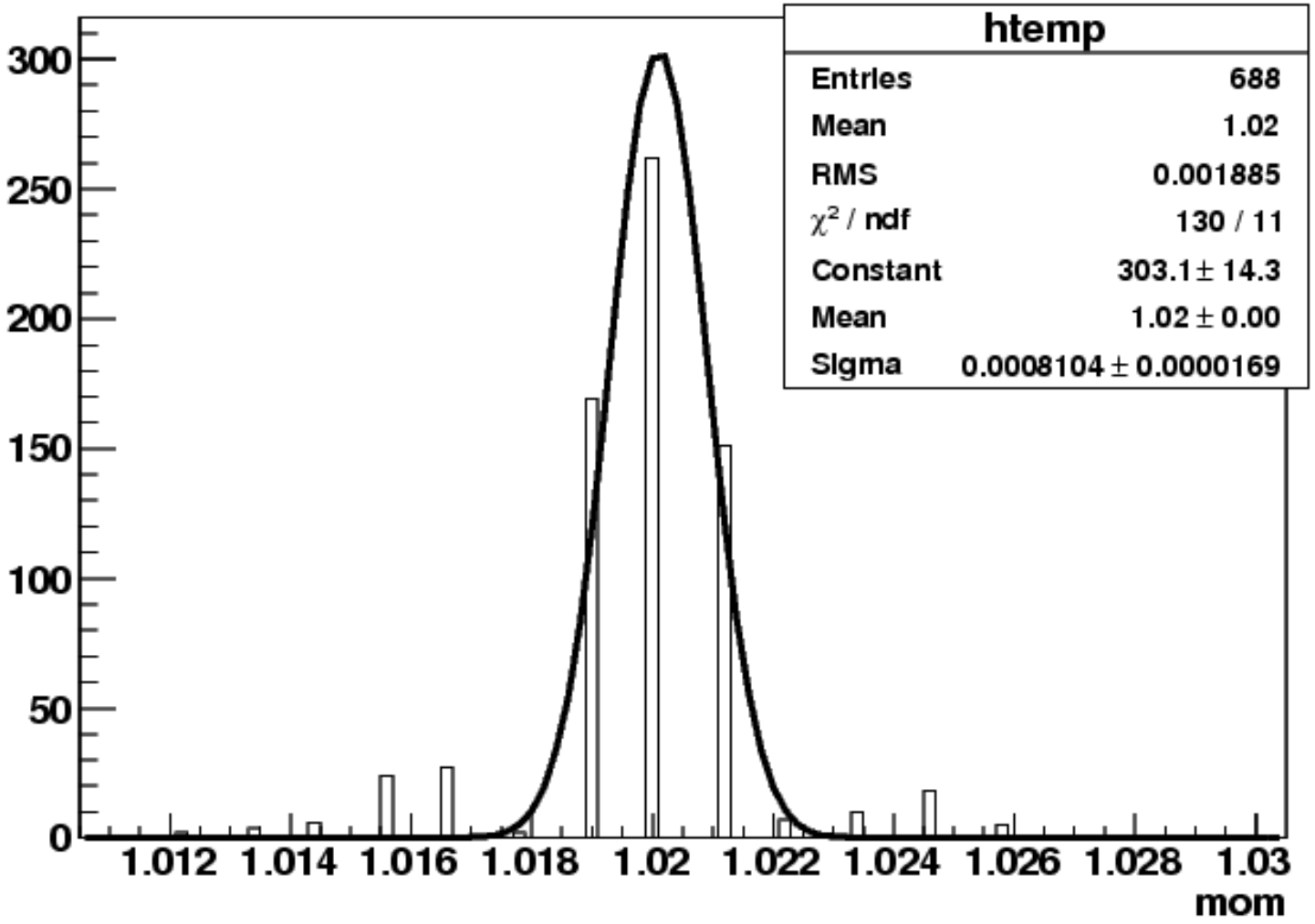
Describing lines by parameters

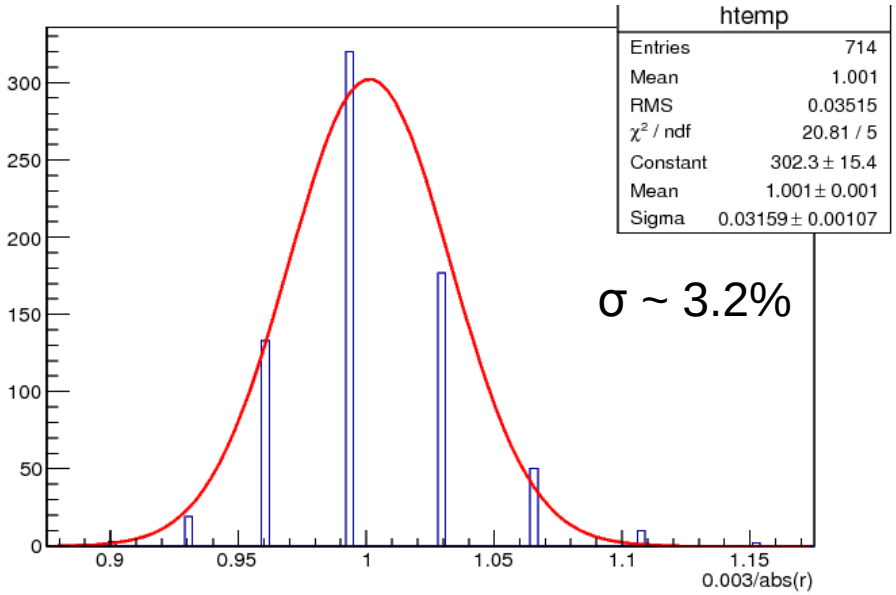
y = mx + b → (m,b) or (r, θ)

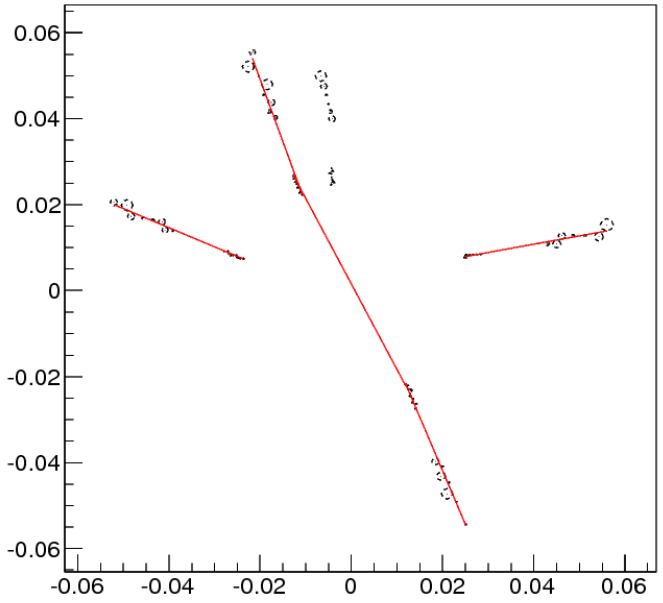
- Use all possible angles
- Save data in histogram
- Peaks in histogram represent possible lines in point set



Momentum resolution can reach this good, if no energy lost in the detector.




htemp
Entries 714
Mean 1.001
RMS 0.03515
χ² / ndf 20.81 / 5
Constant 302.3 ± 15.4
Mean 1.001 ± 0.001
Sigma 0.03159 ± 0.00107
σ ~ 3.2%
0.003/abs(r)



Scheme II

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	25,853	50,560	51%	
DCM autocalibration logic	14	25,853	1%	
Number of 4 input LUTs	22,481	50,560	44%	
DCM autocalibration logic	8	22,481	1%	
Number of occupied Slices	23,706	25,280	93%	
Number of Slices containing only related logic	23,706	23,706	100%	
Number of Slices containing unrelated logic	0	23,706	0%	
Total Number of 4 input LUTs	25,337	50,560	50%	