

Analysis of Timepix test beam data towards documenting results



Sophie Redford - CLIC workshop - 29.01.15
with thanks to the LCD vertex group

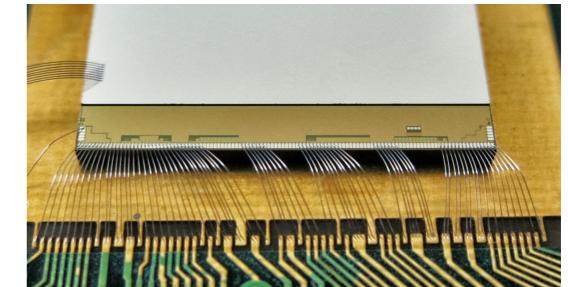
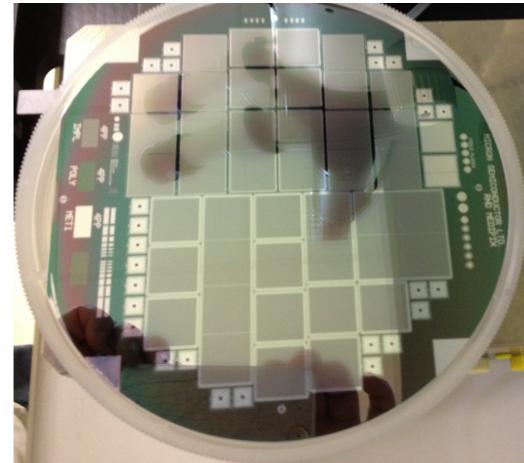
Outline

- 1. Timepix assemblies
- 2. Recorded data
- 3. Test beam data reconstruction
- 4. Hot pixels
- 5. DUT alignment
- 6. Charge sharing as a function of thickness
- 7. Hit resolution as a function of thickness
- 8. Detection efficiency (threshold scan)
- 9. Calibration
- 10. Depletion voltage (bias scan)
- 11. Ideas to increase charge sharing
- 12. Summary & plans

Timepix assemblies

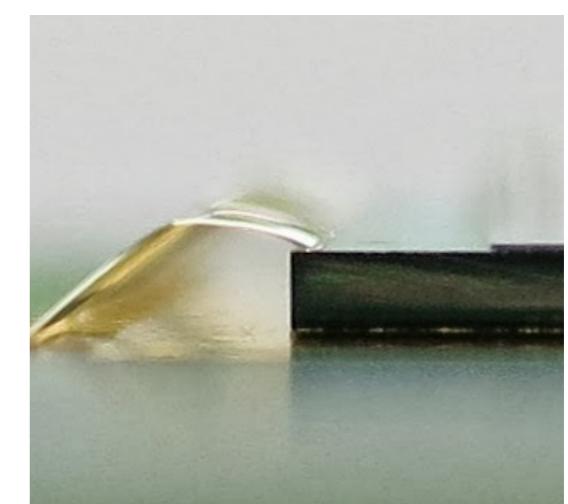
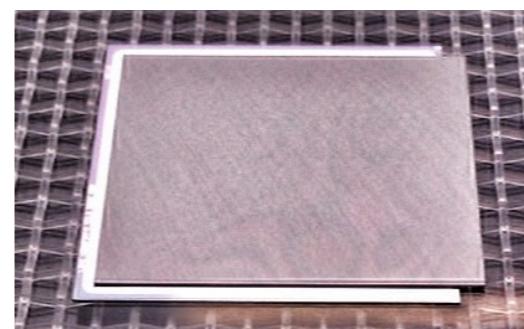
The assemblies:

- Timepix ASIC
- thin silicon sensor
- bump-bonded together
- 256 x 256 pixels
- 55 μm x 55 μm pixel size



Provides (in our case):

- TOT energy measurement per pixel



Variables:

- silicon thickness
- operating conditions of ASIC:
 - bias voltage
 - threshold

Why?

- to characterise thin sensor assemblies
- to validate simulation and extrapolate to smaller pixel size

CLIC vertex detector foresees:

50 μm thick silicon
25 μm x 25 μm pixels

Recorded data

Data recorded with 18 different Timepix assemblies:

- Seven weeks of test beam at DESY (PEP-II): Feb 2013 - Feb 2014
- One week of test beam at CERN (PS): Aug 2014
- ~175M events recorded in test beam
- CERN X-ray sources and fluorescence measurements: Summer 2014
- LNLS measurements: Autumn 2014
- Global and pixel-by-pixel calibration data for 6 assemblies

Assembly	Producer	Sensor			ASIC	
		Thickness (μm)	Type	Edge	Thickness (μm)	
A06-W0110	Advacam	50	p-in-n	20 um active	700	
B04-W0110	Advacam	50	p-in-n	50 um active	700	
C04-W0110	Advacam	50	p-in-n	50 um active	700	
C06-W0110	Advacam	50	p-in-n	20 um active	700	
J09-W0110	Advacam	50	p-in-n	50 um active	700	
C06-W0126	Micron	100	p-in-n	none	100	
D05-W0126	Micron	100	p-in-n	none	100	
D09-W0126	Micron	100	p-in-n	none	100	
L04-W0125	Micron	100	p-in-n	none	700	
L05-W0125	Micron	100	p-in-n	none	700	
D04-W0125	Micron	150	n-in-p	none	700	
D05-W0125	Micron	150	n-in-p	none	700	
D08-W0125	Micron	150	n-in-p	none	700	
B06-W0125	Micron	200	n-in-p	none	700	
B07-W0125	Micron	300	p-in-n	none	700	
C07-W0125	Micron	300	p-in-n	none	700	
I10-W0015	Canberra	300	p-in-n	none	700	
D03-W0170	Canberra	500	p-in-n	none	700	

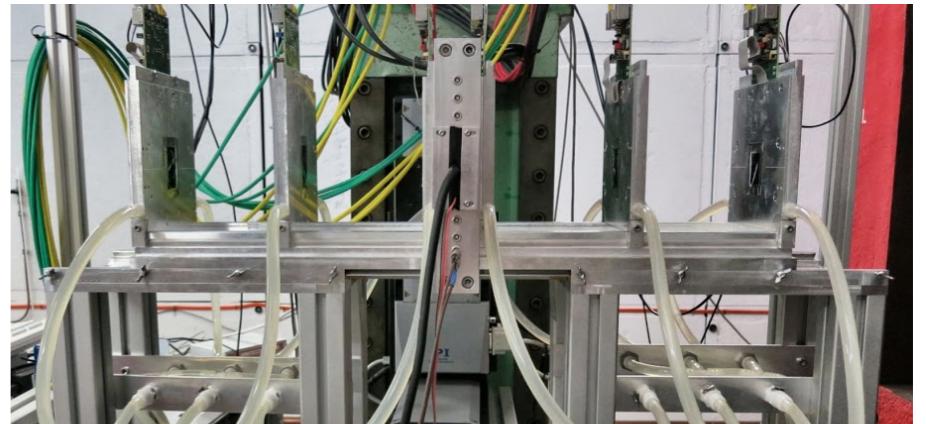
Test beam data reconstruction

The EUDET telescope and framework:

- provides tracks

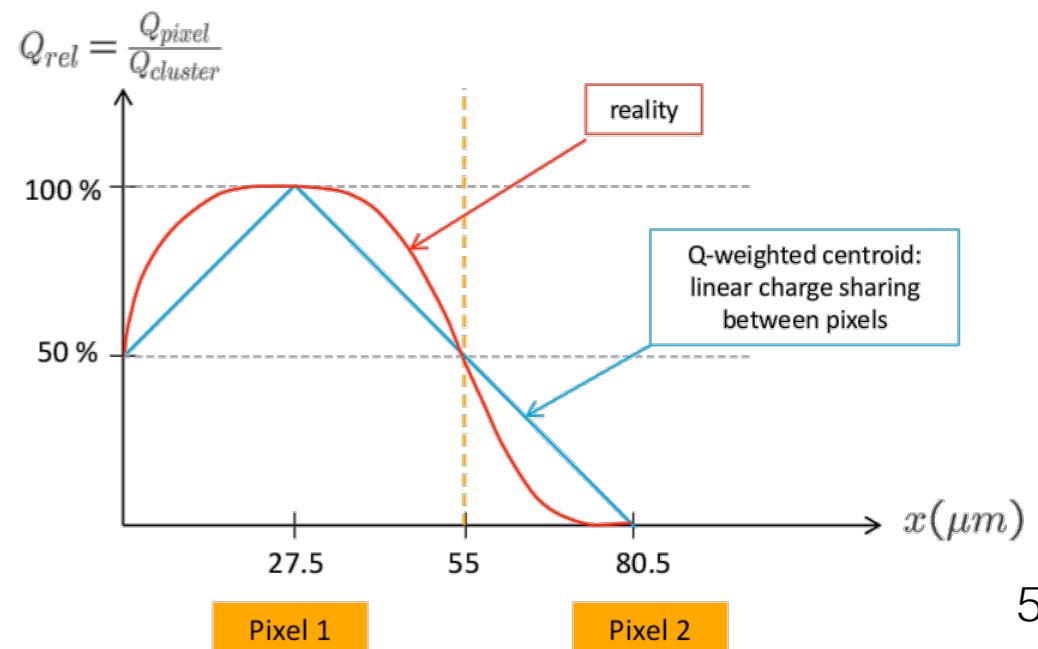
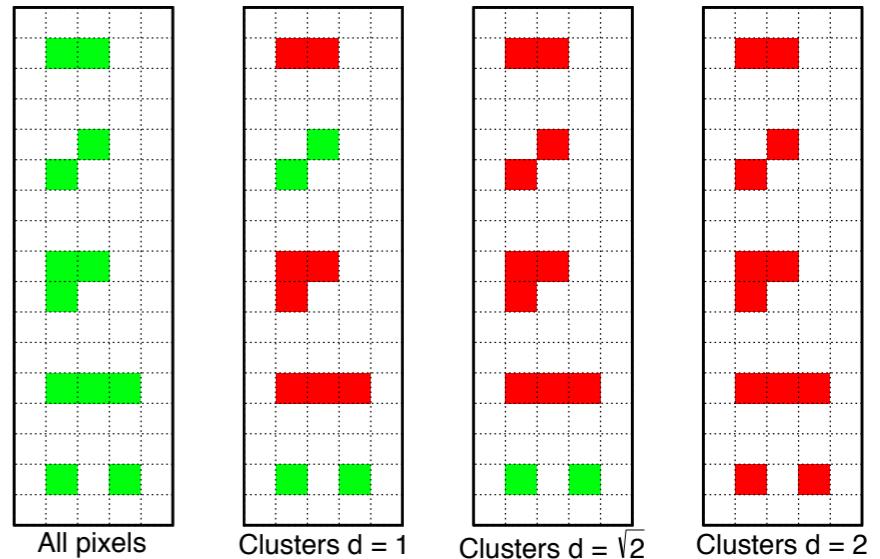
The DUT (device under test - the assembly):

- provides pixel maps



Pixel map treatment:

1. Remove hot pixels (more on this next)
2. Perform clustering
 - using Python function `fclusterdata`
 - using distance criterion $\sqrt{2}$
 - cluster pixels with common edges and corners
3. Reconstruct hits
 - several methods available
 - EtaCorrection is the favourite
 - takes into account non-linear charge diffusion
 - uses a tuneable parameter 'sigma'

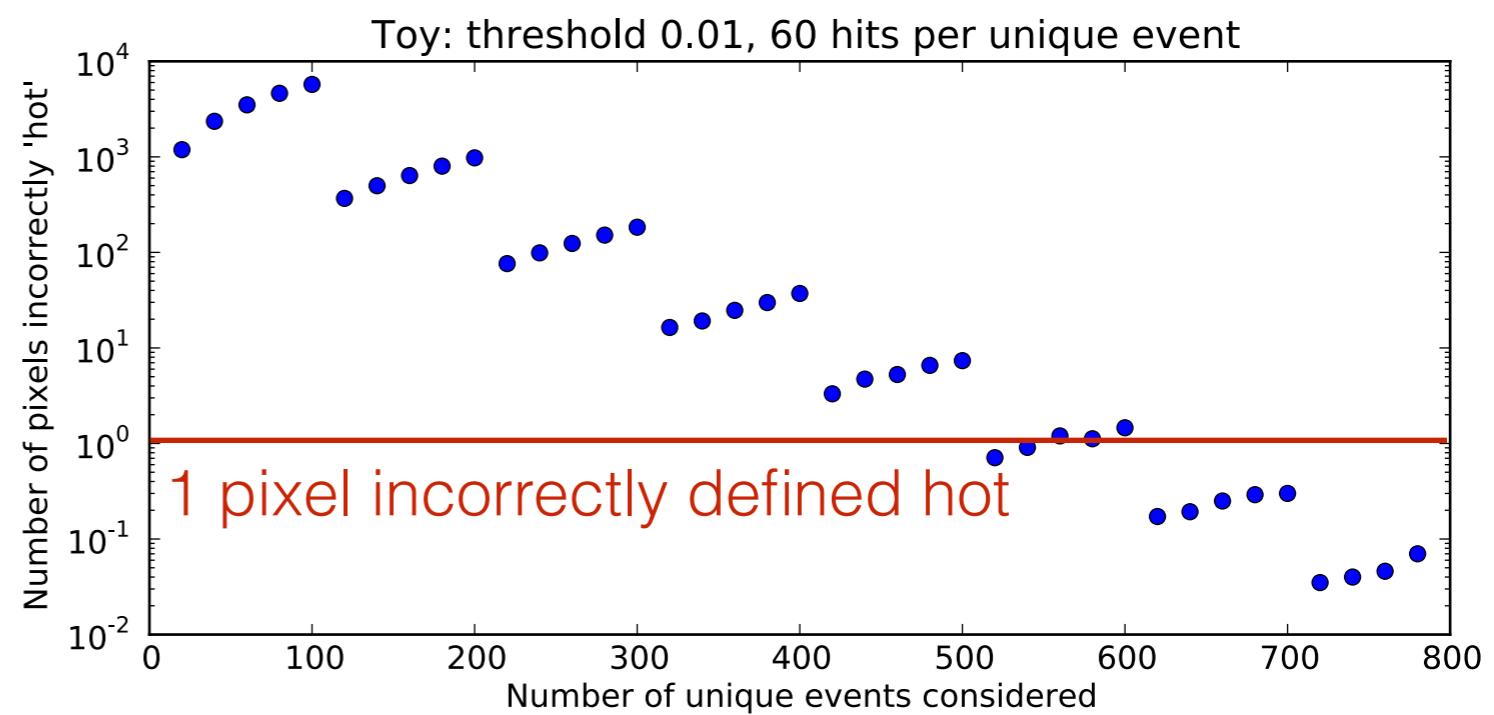
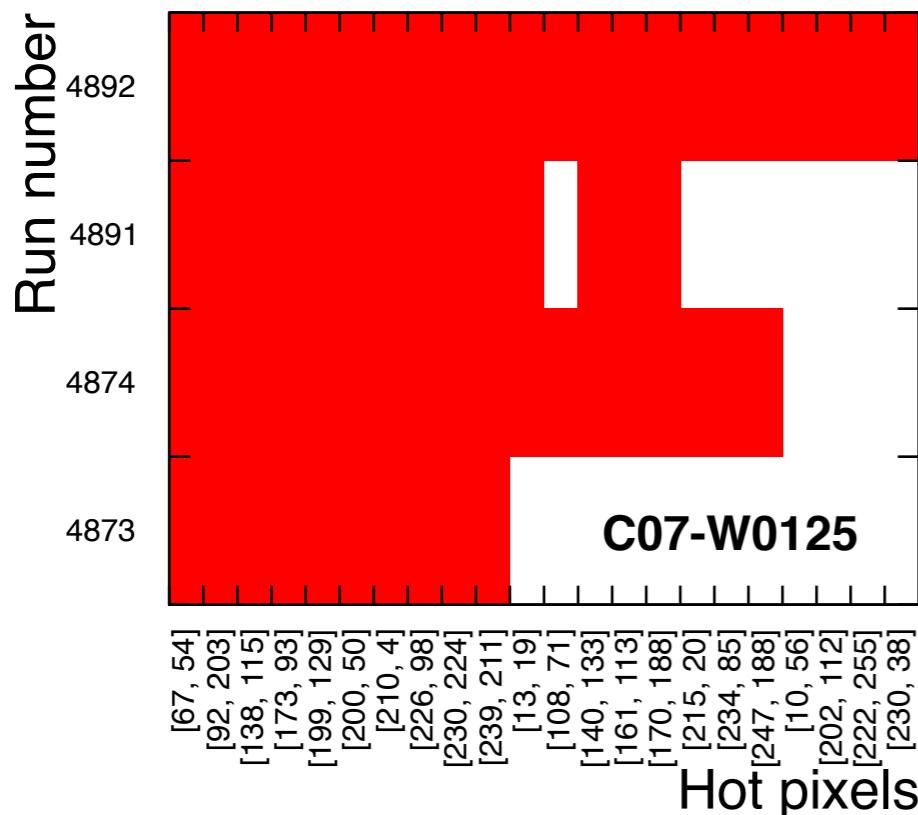


Hot pixels (normal operating conditions)

Hot pixels are defined as pixels which fire more often than once every 100 events
How to implement this in practice?

To be statistically accurate,
need to look at at least 600
unique events (toy study)

→ Use the whole run

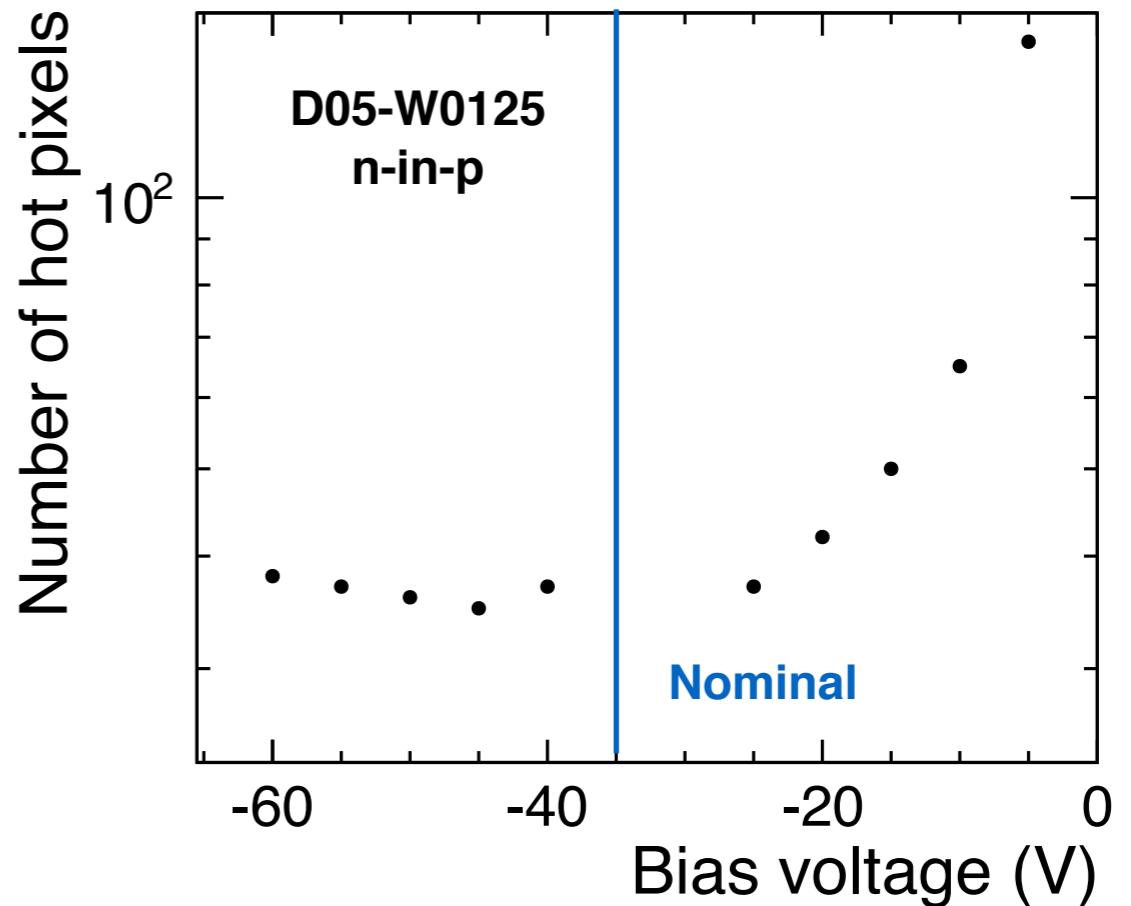
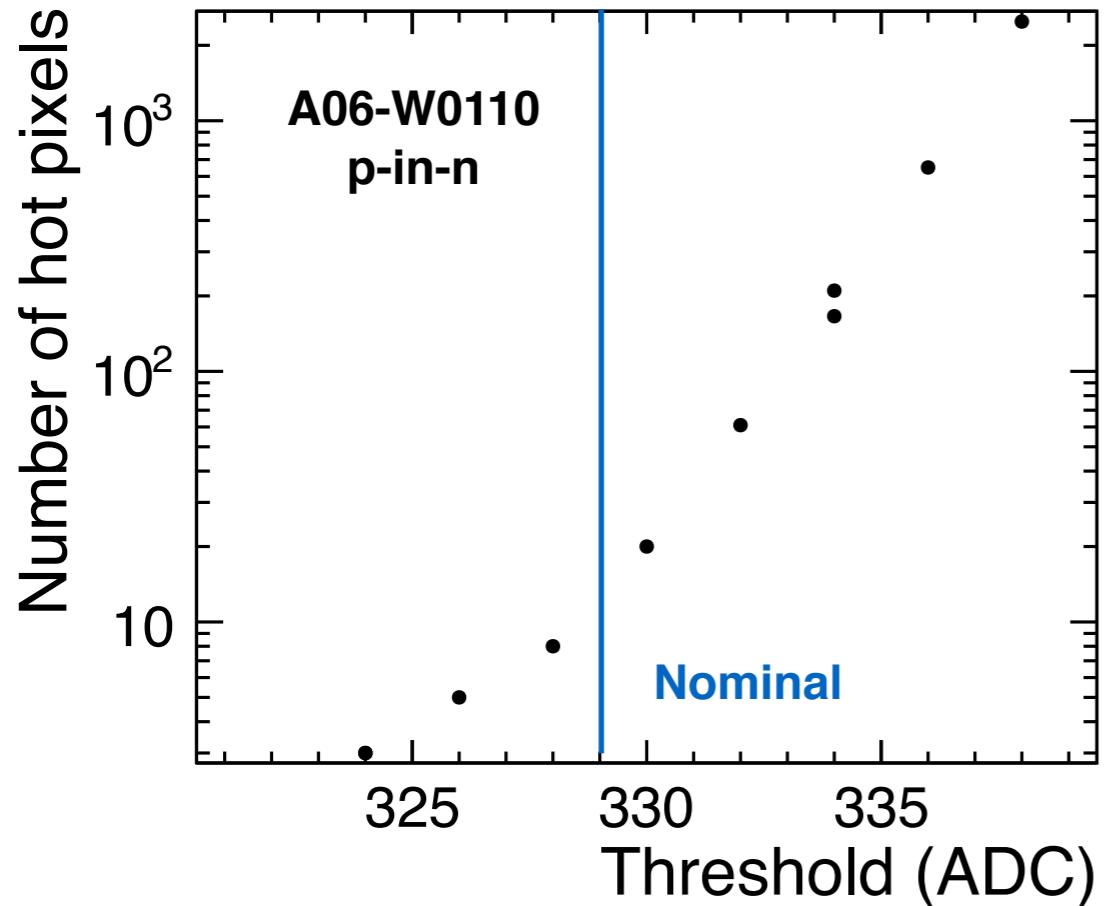


Correlation between hot pixels in TOT runs:

- if a pixel was once hot, likely to remain so
- but not a perfect rule
- and more pixels can become hot over time

Hot pixels (abnormal operating conditions)

Hot pixels also depend on operating conditions (bias voltage, threshold)



'Lower' threshold = more hot pixels

- 10³ pixels = 1.5% of all pixels
- Will affect efficiency (more on this later)

'Lower' bias = more hot pixels

- Need to remove this extra noise

→ Therefore it was decided that each run defines its hot pixels:

Hot pixels are defined as pixels which fire more often than once every 100 events

DUT alignment method

The hits on the DUT must be aligned with the tracks - done per run

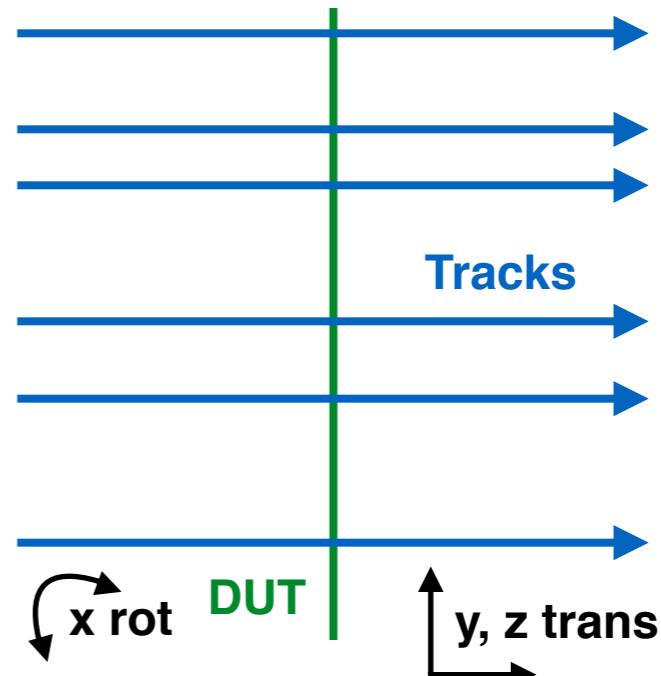
DUT has 6 possible degrees of freedom:

- translations in x, y, z
- rotations about x, y, z

Insensitive to 3 of these:

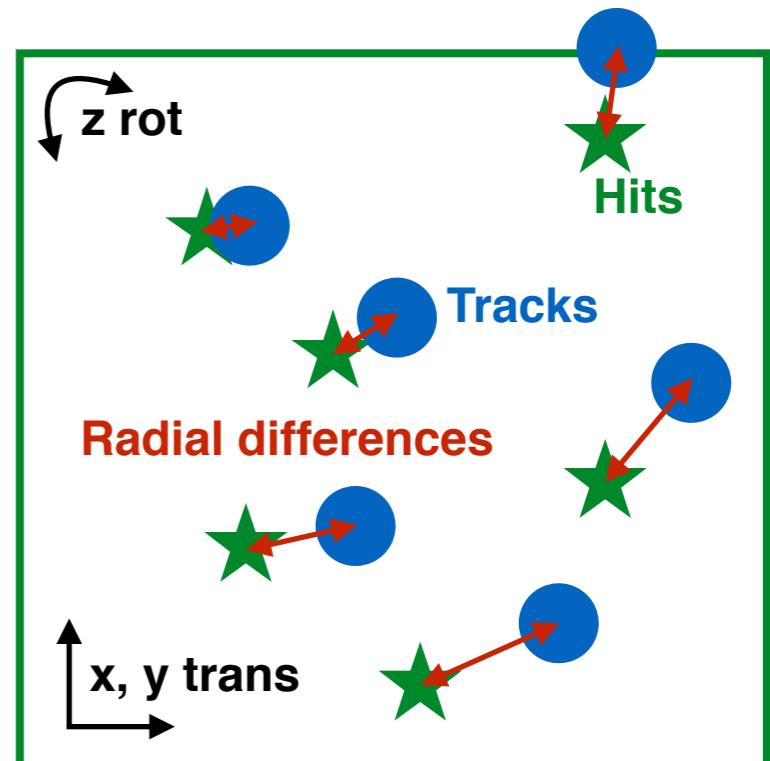
- translation in z (straight tracks)
- rotation about x and y (2D DUT, small angles)

These 3 parameters are fixed, other 3 are calculated



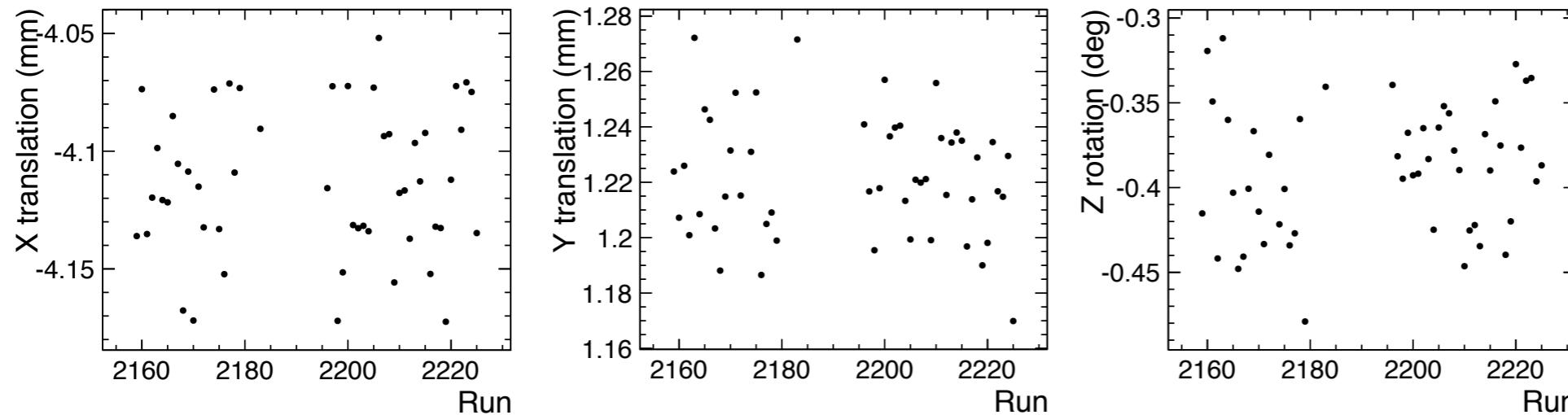
Alignment method:

- pick 3 alignment constants (x_{trans} , y_{trans} , z_{rot})
- calculate the total radial difference between all tracks and their nearest hits
- minimise this total distance by varying alignment constants (python minimize function)



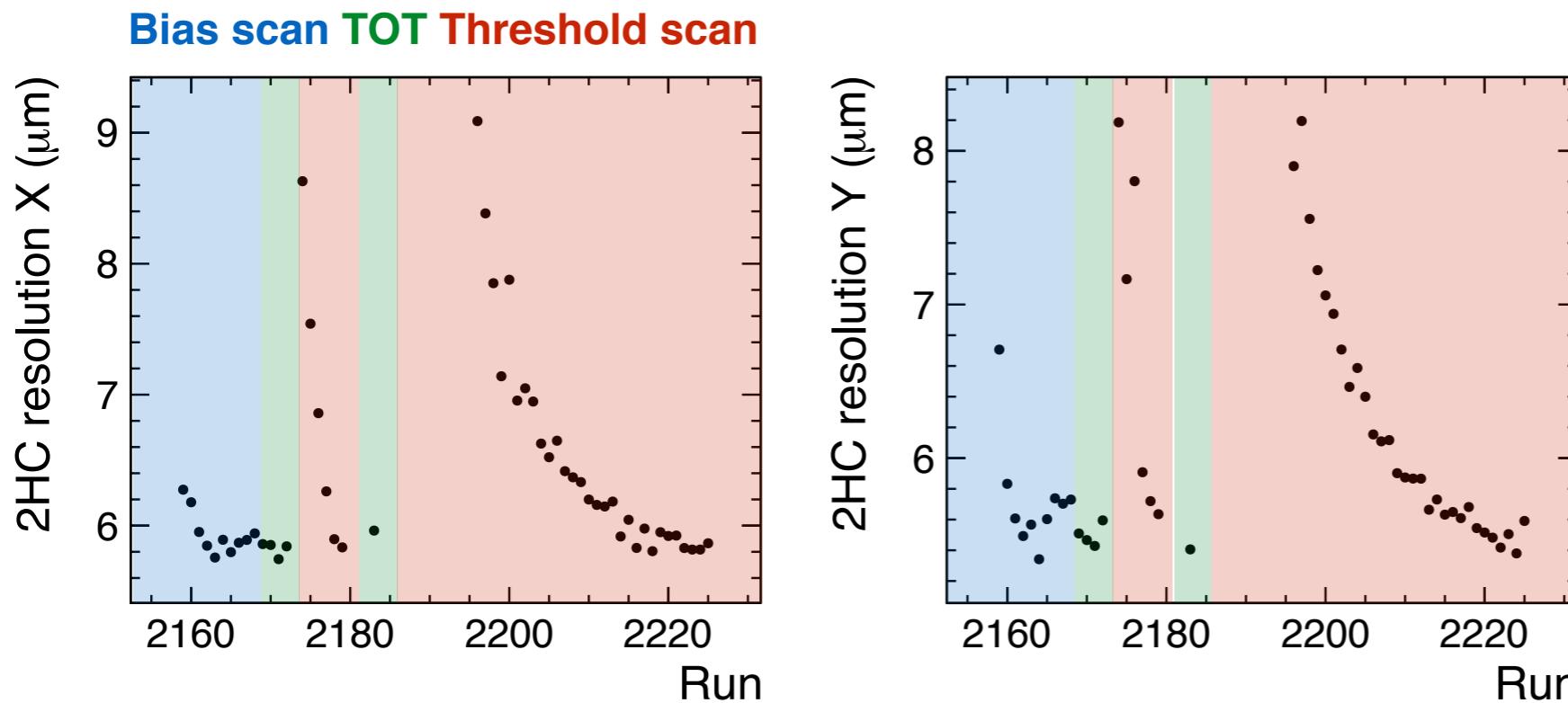
DUT alignment results

Look at calculated alignment parameters over a series of runs: (C06-W0126)



Consistent results
despite independent
calculation per run

Look at (non-optimised) two-hit cluster resolution over a series of runs:



Consistent results,
understandable
variation during bias,
threshold scans
(which will disappear when Eta
Correction sigma is corrected
to take into account different
drift velocity)

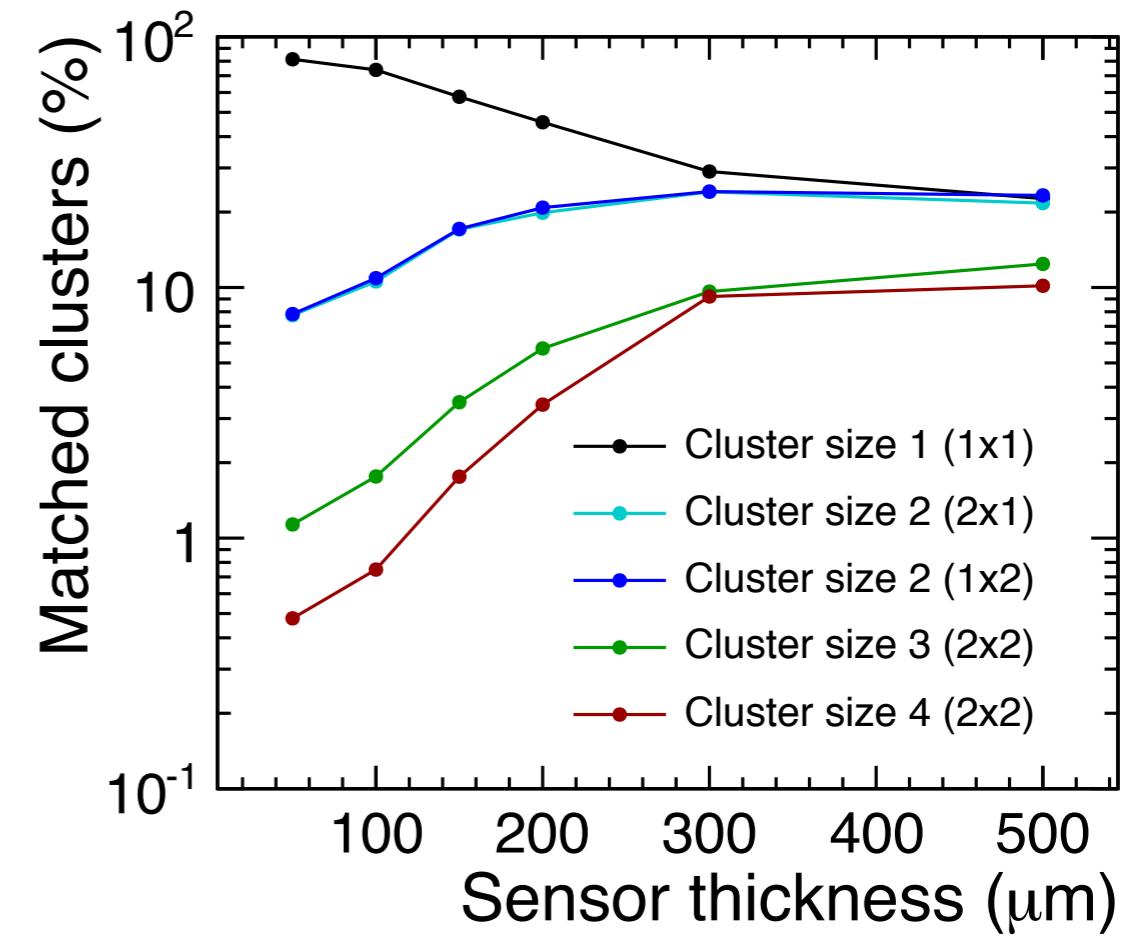
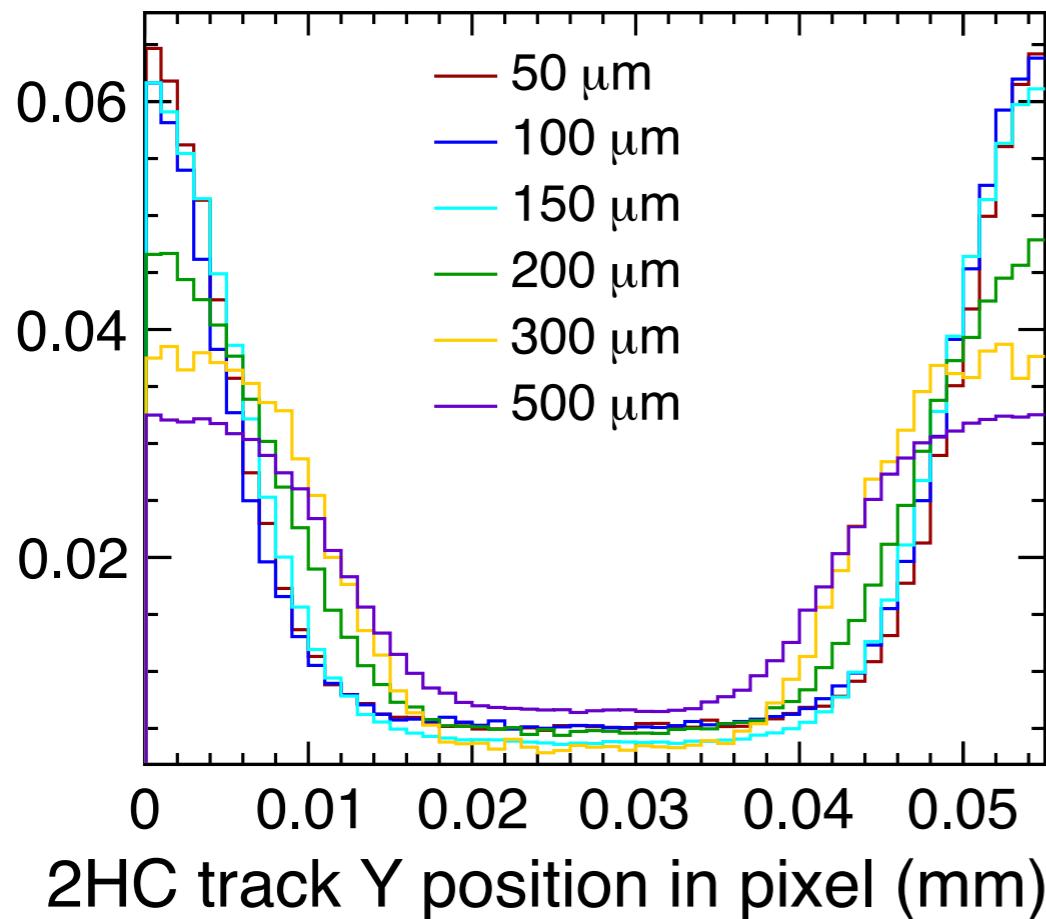
Charge sharing

How many tracks make multi-hit clusters? (So allowing us to benefit from the analogue readout for improved resolution measurements)

Thicker sensors have more charge sharing

Percentage of single hit clusters:

- 81% with 50 μm silicon
- 23% with 500 μm silicon

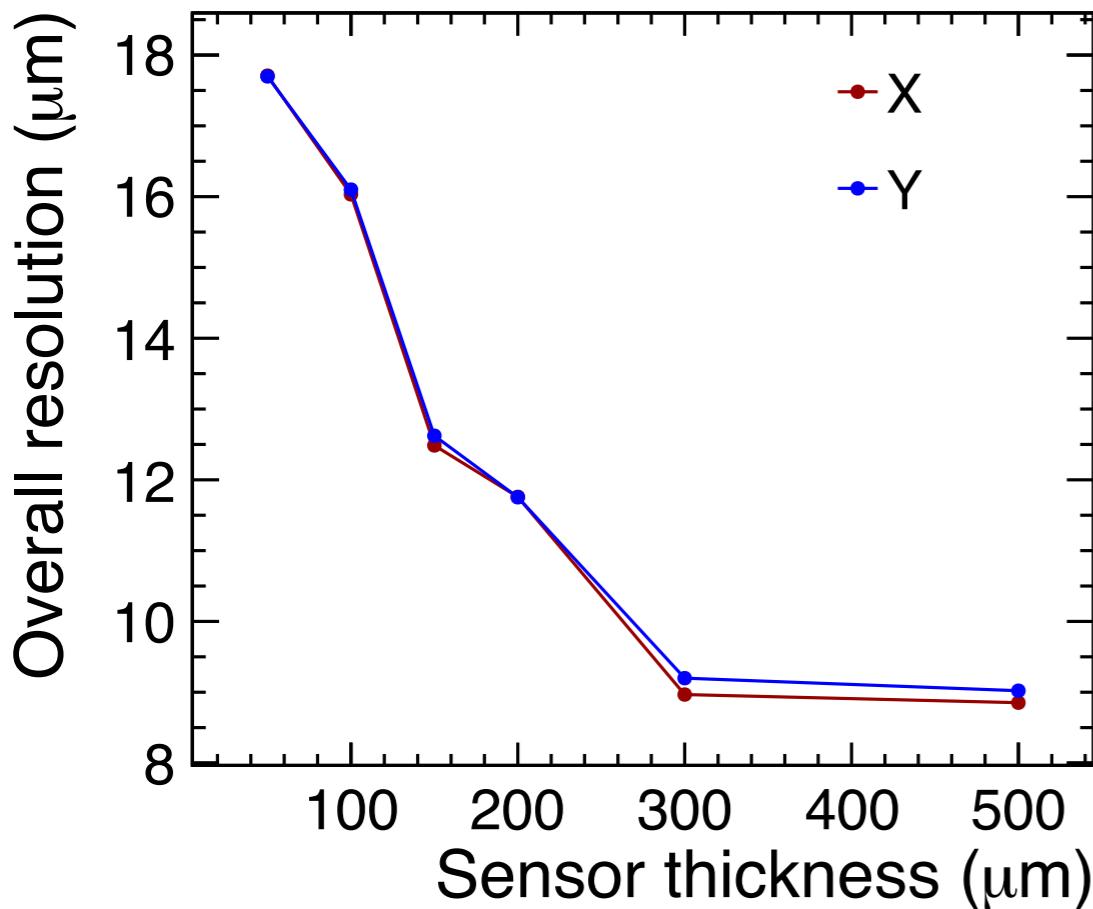


... because in thicker sensors, tracks further from the edges of the pixels make multi-hit clusters due to larger transverse drift distances

Unfortunately material budget constraints limit the silicon thickness to 50 μm .
(Ideas on ways to increase charge sharing later)

Hit resolution (including track resolution)

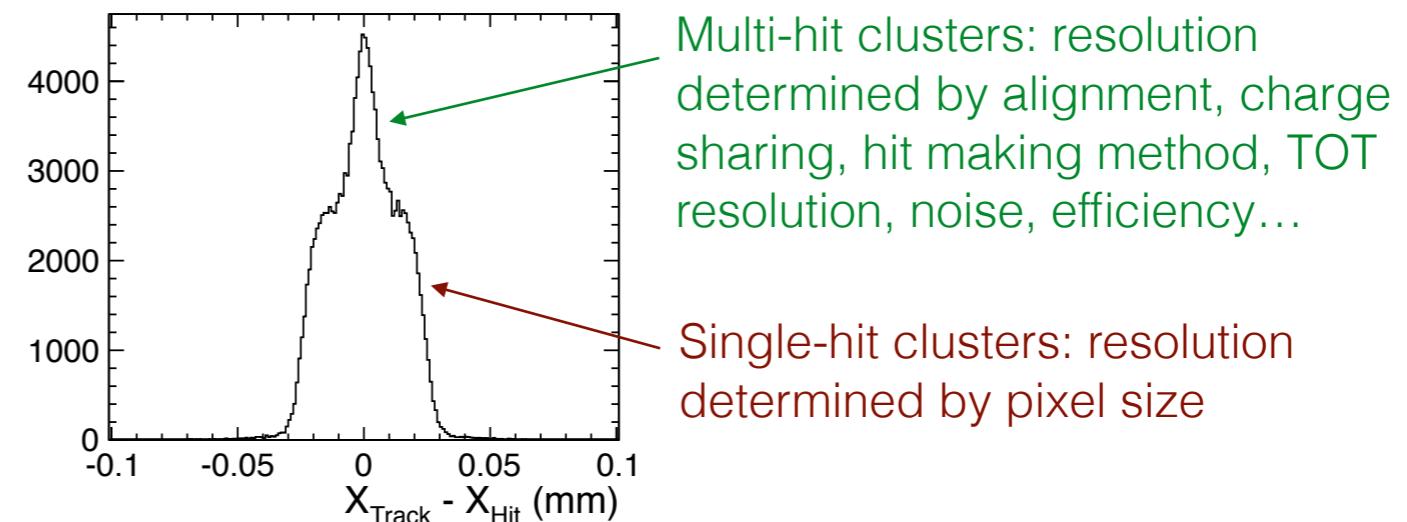
Recall: we use the EtaCorrection hit making method, with a sigma parameter optimised for sensor thickness, type and bias voltage



Two-hit cluster resolution should be (to first order) equal across sensor thickness

Second order effects:

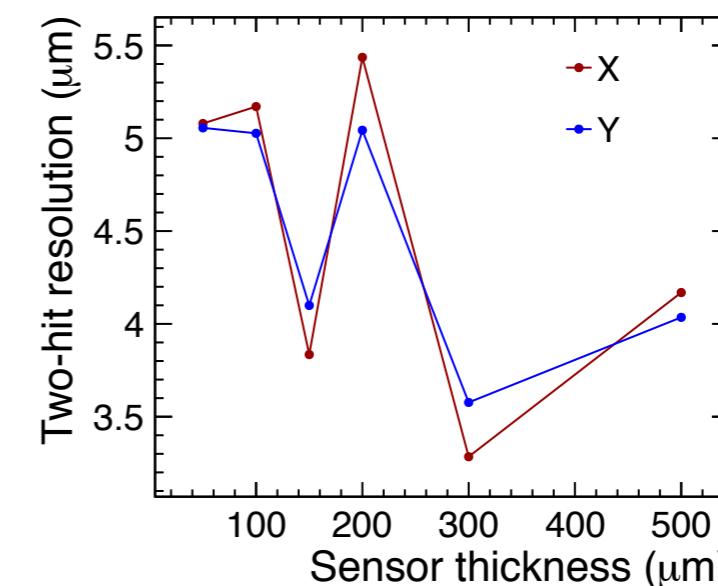
- 300 μm data recorded at CERN PS - higher E so smaller track resolution
- assembly-to-assembly variation (1 μm)



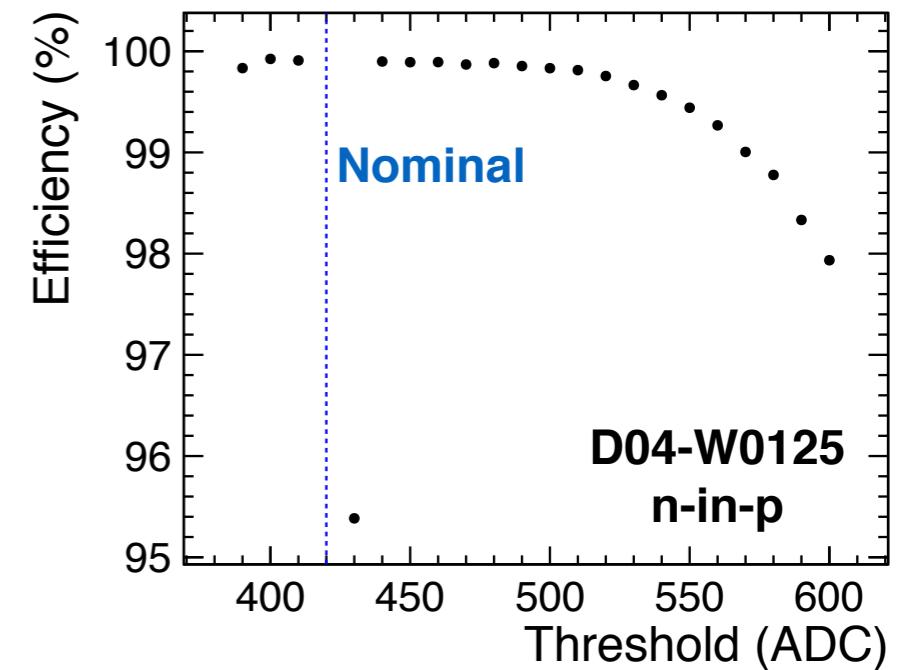
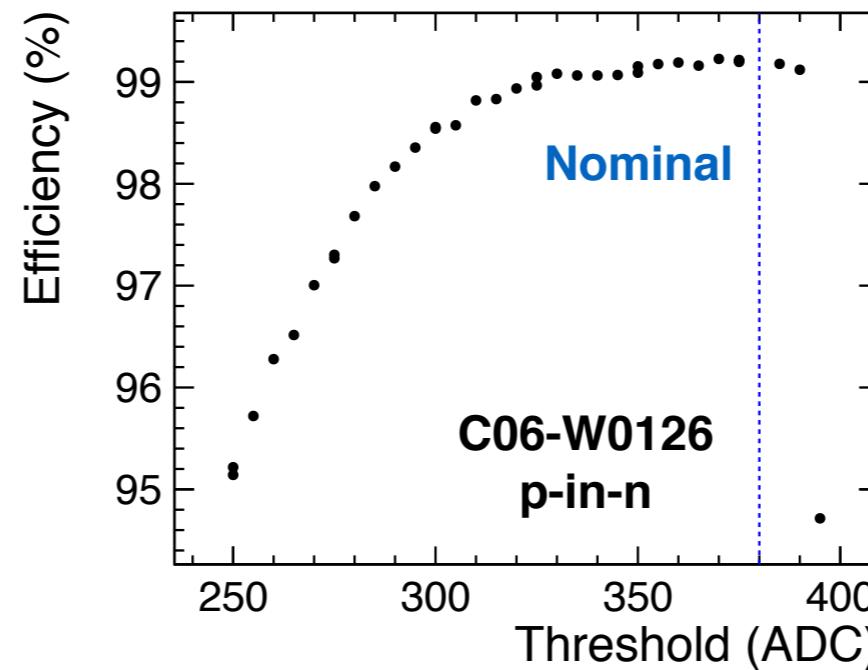
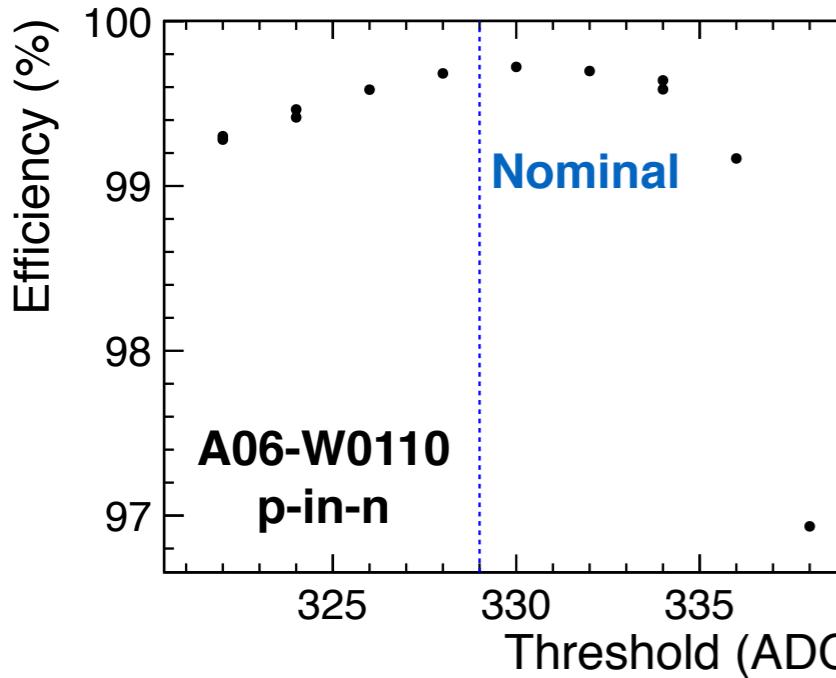
Multi-hit clusters: resolution determined by alignment, charge sharing, hit making method, TOT resolution, noise, efficiency...

Single-hit clusters: resolution determined by pixel size

Overall hit resolution is determined primarily by the amount of charge sharing (and hence the sensor thickness)



Detection efficiency



Detection efficiency calculation method:

- remove hot pixels
- for every track which passes through the DUT,
look for a hit within $r = 0.5$ mm

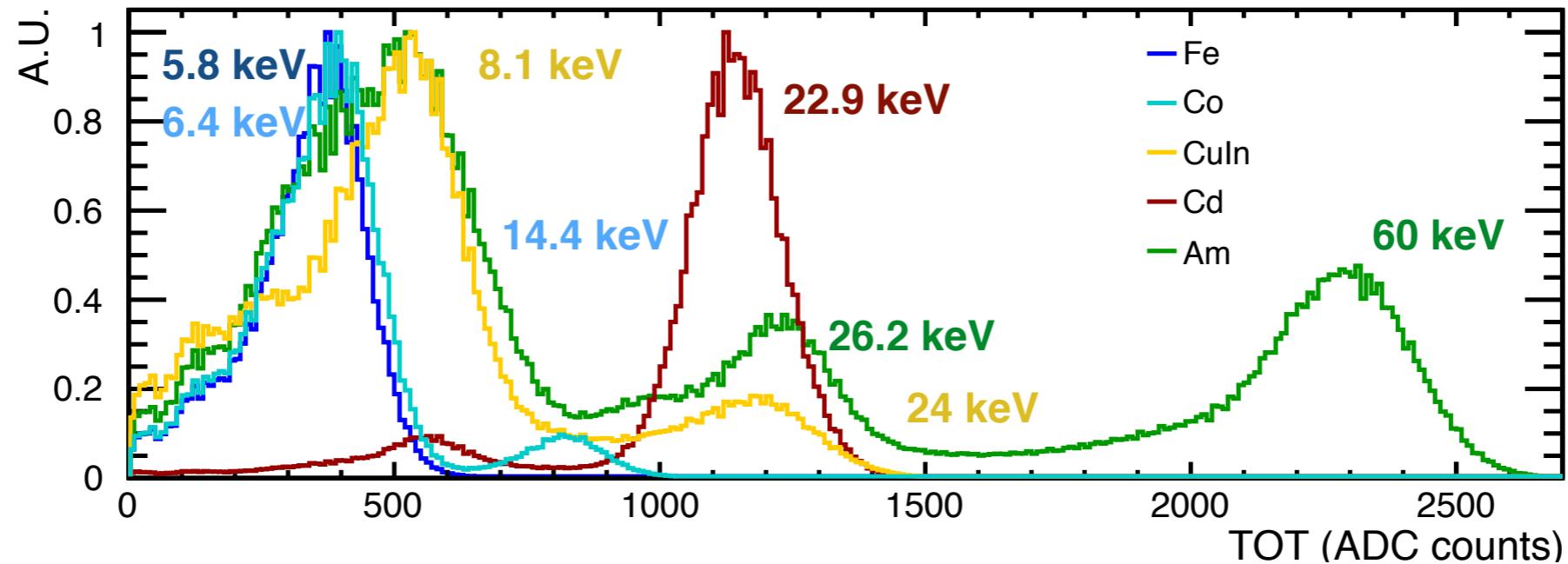
Very ‘honest’ measurement:

- hot pixels removed, not corrected for
- bad pixels masked, not corrected for

Theoretical maximum is not 100%

Calibration

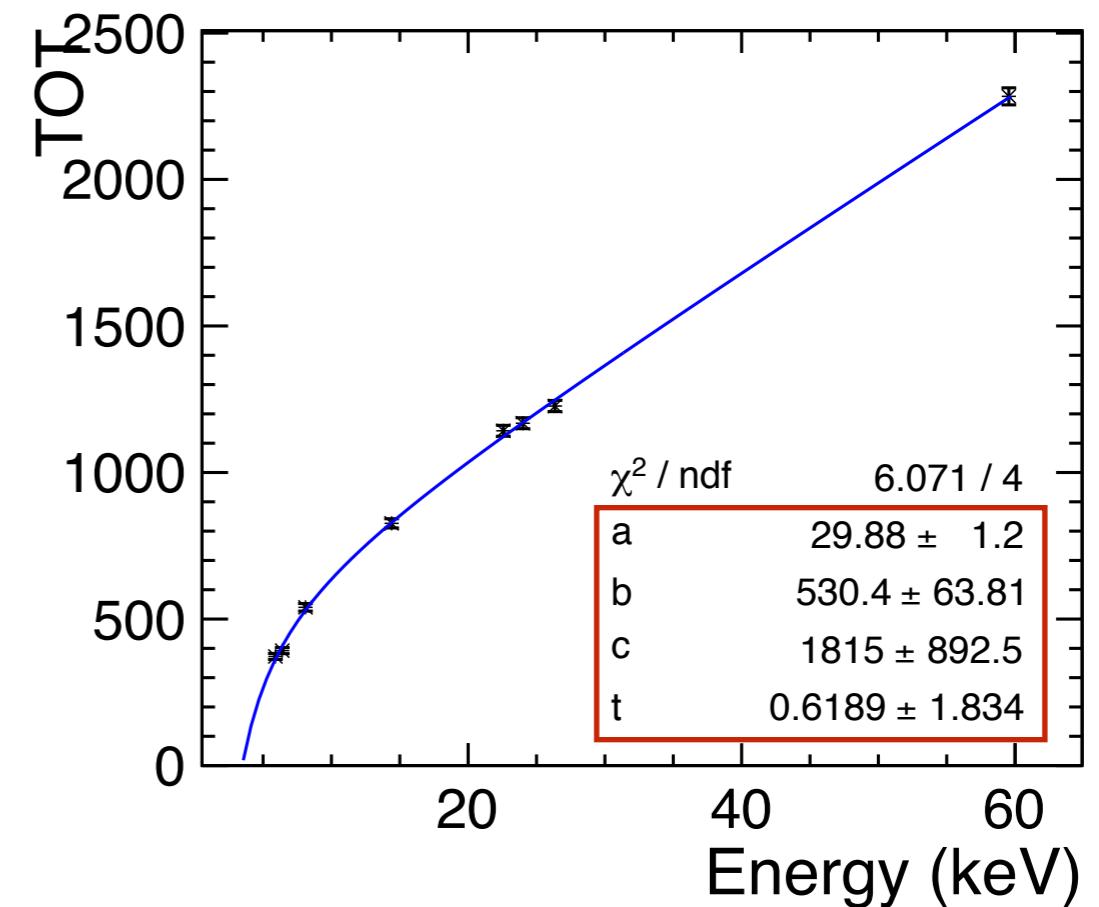
Use photons of known energy to calibrate TOT measurements



Calibration method:

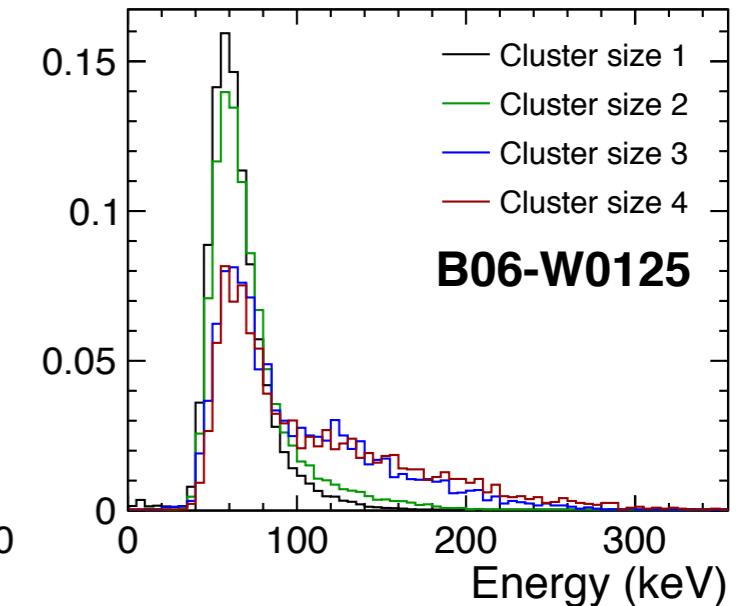
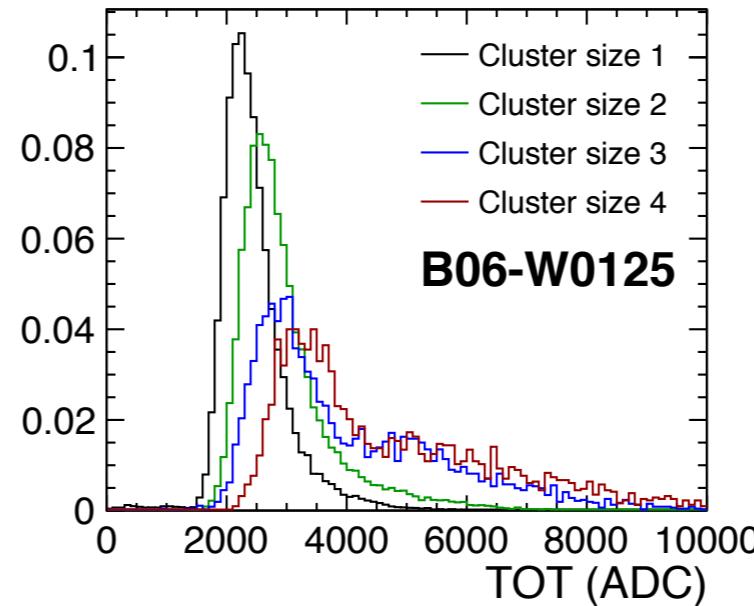
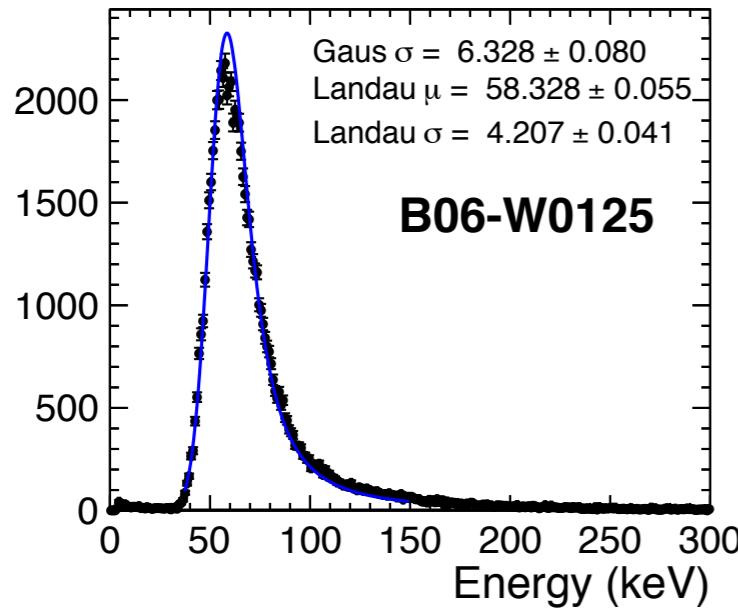
- fit peaks to find mean TOT values
- plot TOT vs energy, fit with surrogate function to find **calibration constants**
- invert surrogate function, use constants to turn measured TOT into energy

The response is non-linear due to the pre-amplifier in the Timepix ASIC



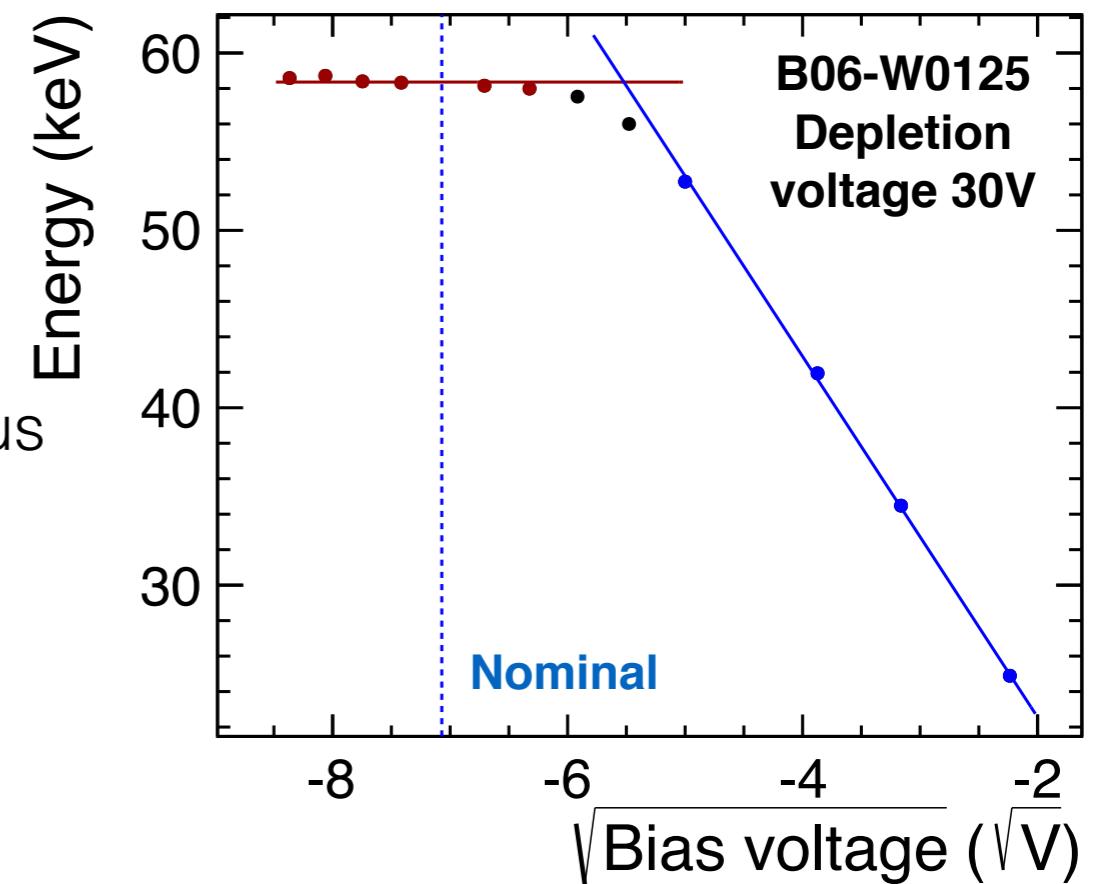
Depletion voltage

Applying the calibration to test beam data turns TOT measurements into energy



Using bias scan data we can determine the depletion voltage of each sensor:

- Fit the energy distribution with Landau x Gaus
- Find the mean energy
- Plot mean energy as a function of $\sqrt{\text{bias}}$
- Intersect of linear and sloped section gives depletion voltage



Ways to increase charge sharing

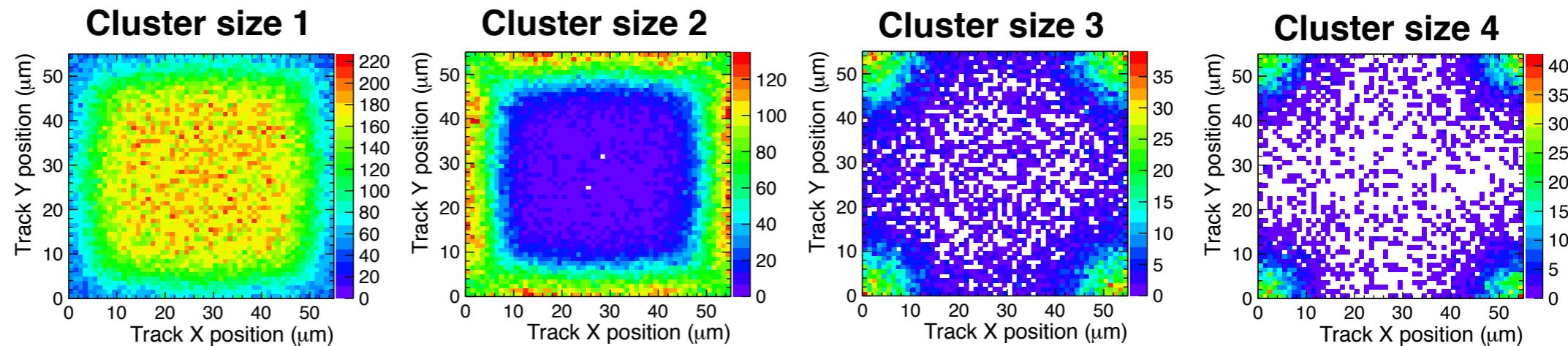
- To achieve 3 μm single point resolution with $25 \mu\text{m} \times 25 \mu\text{m}$ pixels, the majority of hits need to make a multi-hit cluster
- So we need to maximise charge sharing. How?
 1. Smaller pixels for the same sensor thickness
 - only beneficial
 2. ‘Lower’ the threshold
 - collecting smaller energy deposits vs collecting noise
 3. ‘Lower’ the bias
 - collecting enough charge to get over threshold vs pulling it too strongly

Disclaimer: First thoughts, not an expert, just to start discussions

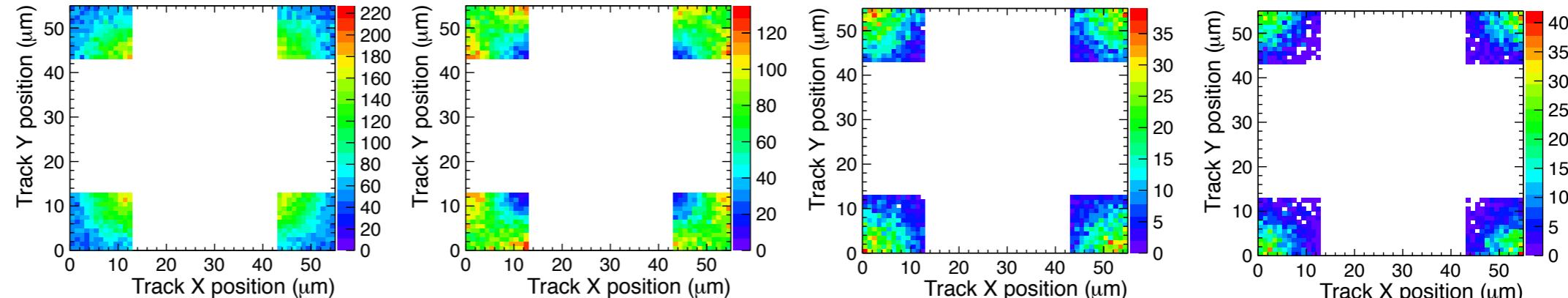
Extrapolating to smaller pixels

Using data from $55\text{ }\mu\text{m} \times 55\text{ }\mu\text{m}$, $50\text{ }\mu\text{m}$ thick sensor

All the data



Just the corners
(removes 80% of
the area)



Corners together
simulating $50\text{ }\mu\text{m}$ thick
 $25\text{ }\mu\text{m} \times 25\text{ }\mu\text{m}$ pixels

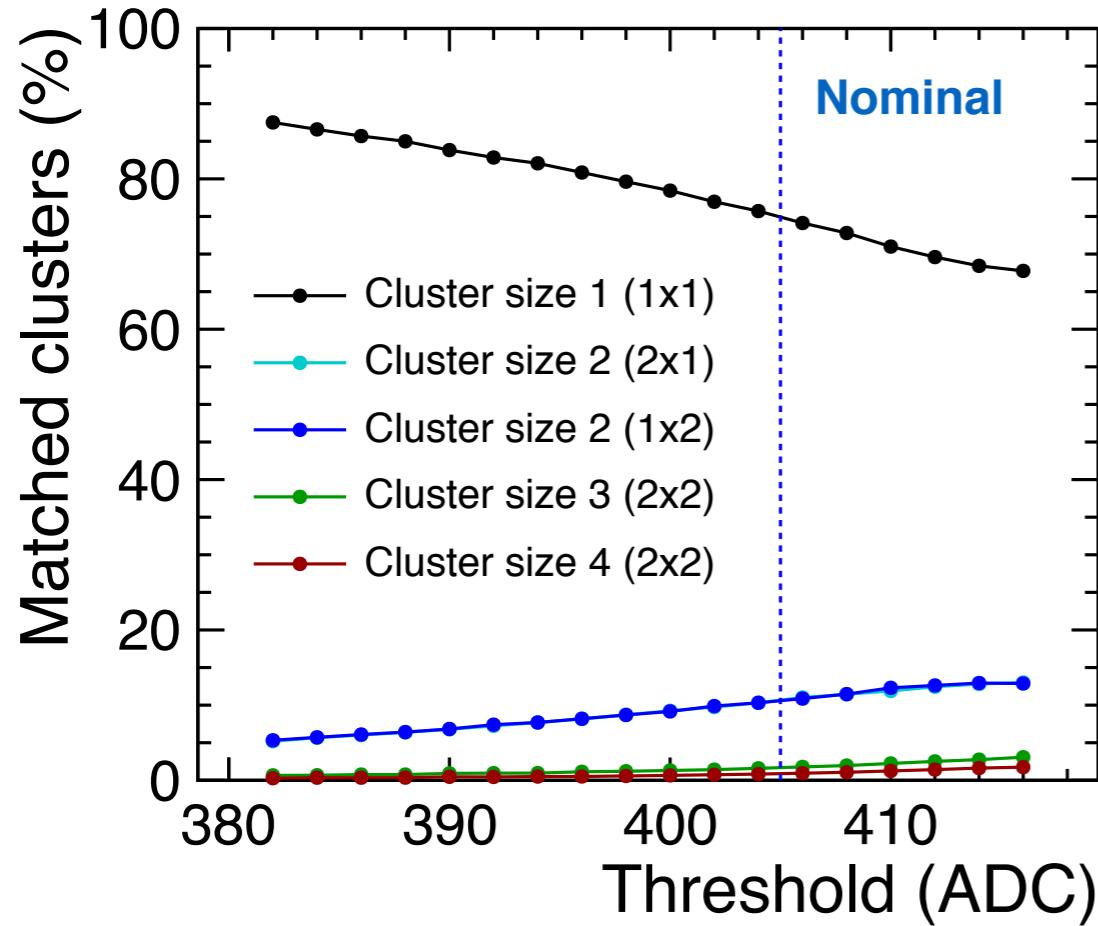


Resolution in $25\text{ }\mu\text{m} \times 25\text{ }\mu\text{m}$ pixels will improve due to:

- more multi-hit clusters (difficult to quantify here due to effects of tracking resolution)
- smaller pixels also improve single hit resolution:

$$\frac{55\mu\text{m}}{\sqrt{12}} = 15.9\mu\text{m} \quad \frac{25\mu\text{m}}{\sqrt{12}} = 7.2\mu\text{m}$$

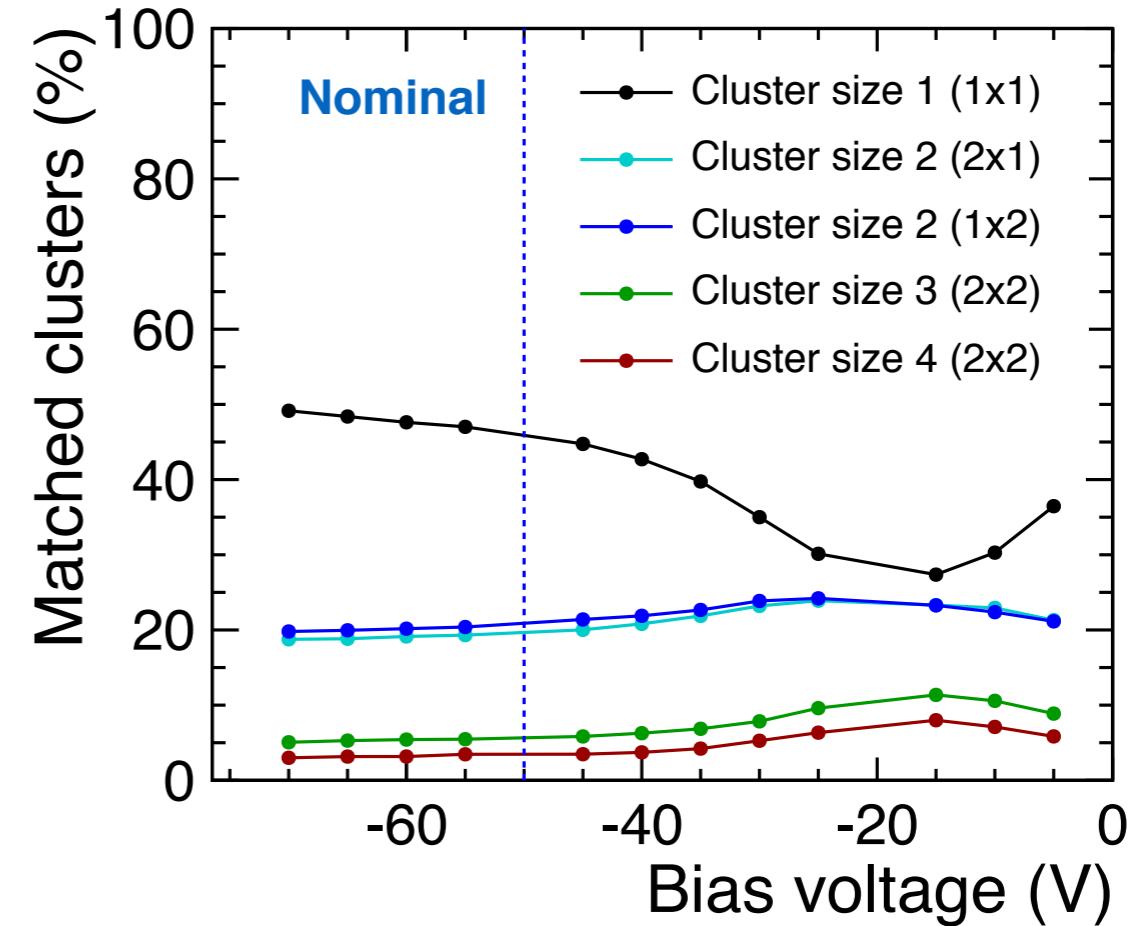
Varying operating conditions



C04-W0110: 50 µm silicon, p-in-n
'Lower' threshold = more charge sharing

Need to balance:

- collecting smaller energy deposits
- collecting noise



B06-W0125: 200 µm silicon, n-in-p
'Lower' bias = more charge sharing

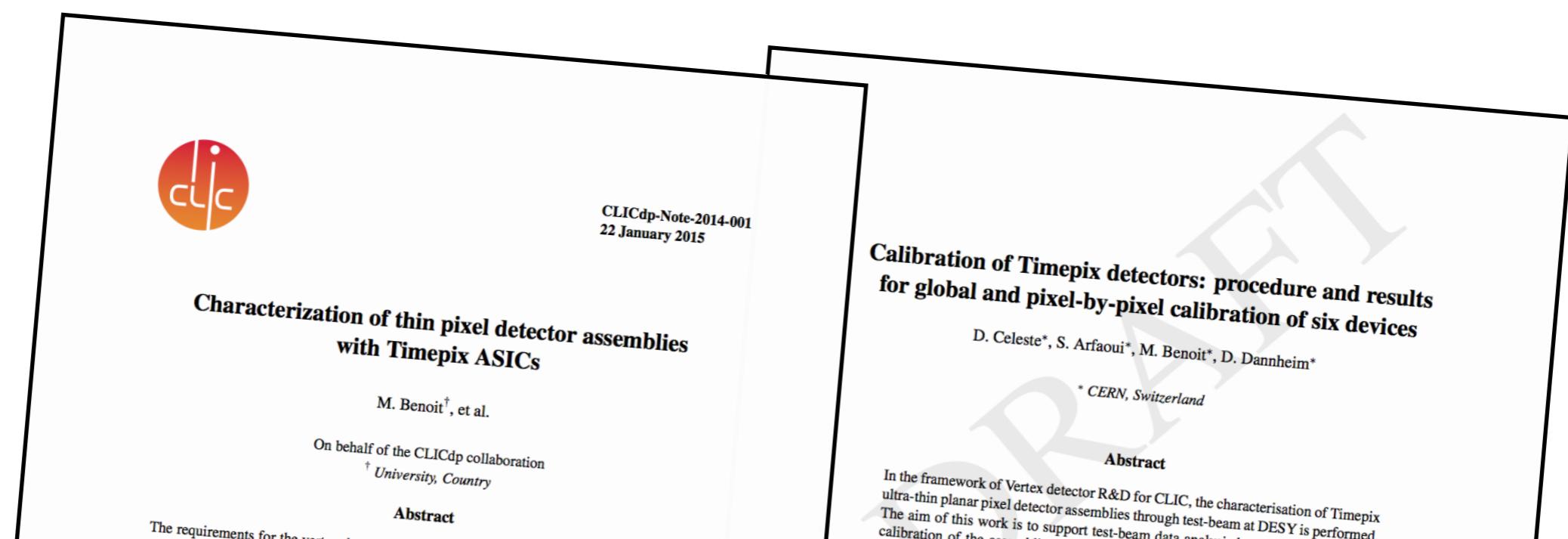
Sweet spot between:

- collecting enough energy to get over thr.
- E field too strong, reducing drift radius

Note: under-depleted - silicon thickness not fully used

Summary

- Timepix assemblies → data recorded → reconstructed **Thanks team!**
- Hot pixels **well defined**
- DUT alignment **simple procedure, good results**
- Charge sharing **need more! Possibilities exist**
- Hit resolution **good thanks to Eta Correction method**
- Detection efficiency **excellent**
- Depletion voltage **uses calibration**
- **What's left/plans:** pixel-by-pixel calibration, lab tests, edge efficiencies, DUT drifts, large clusters, track isolation, 3-hit cluster resolution...
documentation: started



Thanks for your attention