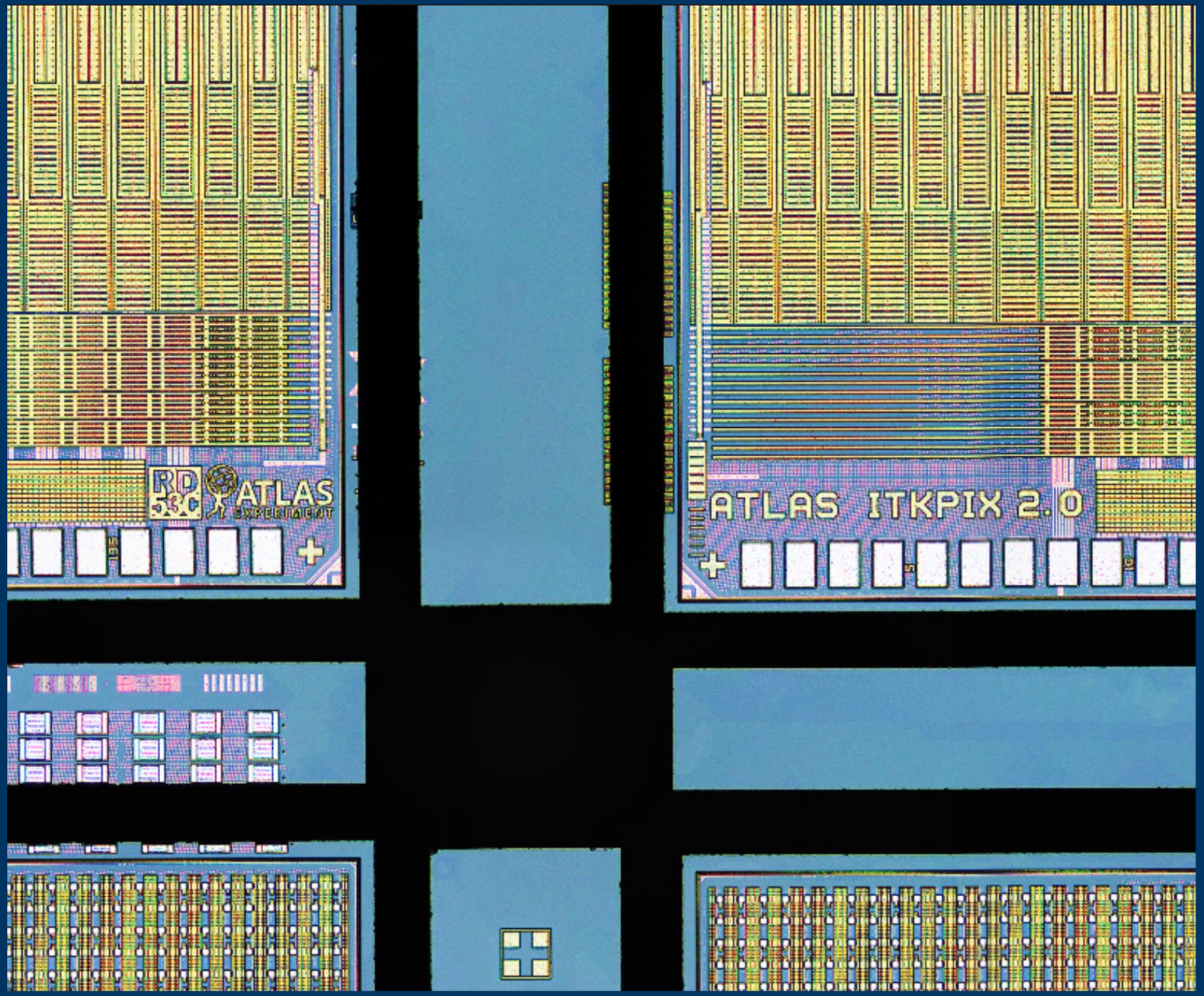


Diced ITkPixV2 Readout Chips (Maurice Garcia-Sciveres)

Testing the Limits of ITkPixV2

The ATLAS Inner Tracker Pixel Detector Readout Chip

Luc Le Pottier, Timon Heim, Maurice Garcia-Sciveres, Maria Mironova, and Liam Foster

Lawrence Berkeley National Lab

November 17th, 2024

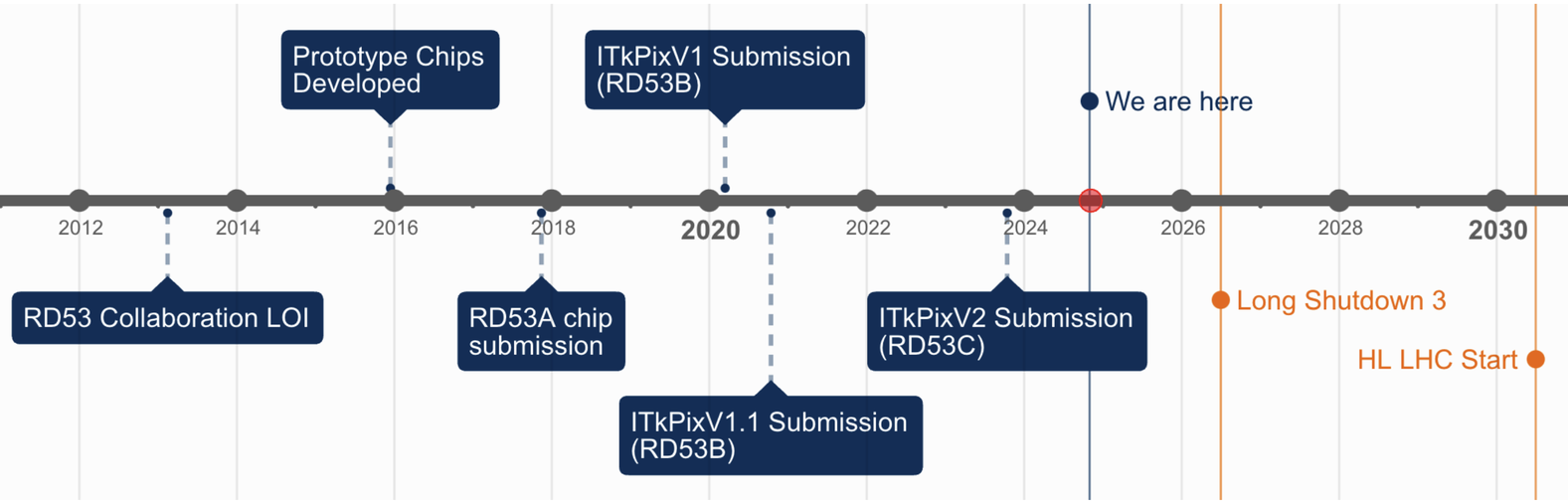
The ITkPixV2 Chip

	Curent Pixel	ITKPixV2	Difference
Chip Size	2x2 cm^2	2x2 cm^2	0 %
Pixel Size	50 x 250 μm^2	50 x 50 μm^2	5x less
Hit Rate	0.4 GHz / cm^2	3 GHz / cm^2	7.5x higher
Trigger Rate	0.1 MHz	1 MHz	10x higher
Trigger Latency	6.4 µs	12.8 µs	2x longer
Radiation Tolerance	300 MRad	500 MRad	1.6x higher
Current Draw	20 µA/pixel	<8 µA/pixel	2.5x lower
Min. Threshold	1500e	600e	2.6x lower

Summarized ITkPixV2 **Design Requirements**

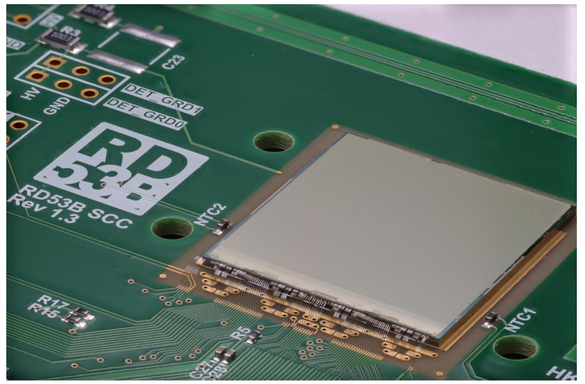
Development

- Final version of chip with a 10-year cycle by **RD53 Collaboration**, first proposed in 2013
- Building block of ATLAS ITk Pixel Detector (see talk from Simone Monzani, 10:30am Monday)



Requirements

- Major improvement over current pixel detector
- Needs to accommodate a wide range of conditions: **readout, hit rate,** and **trigger rate (+10x!)**

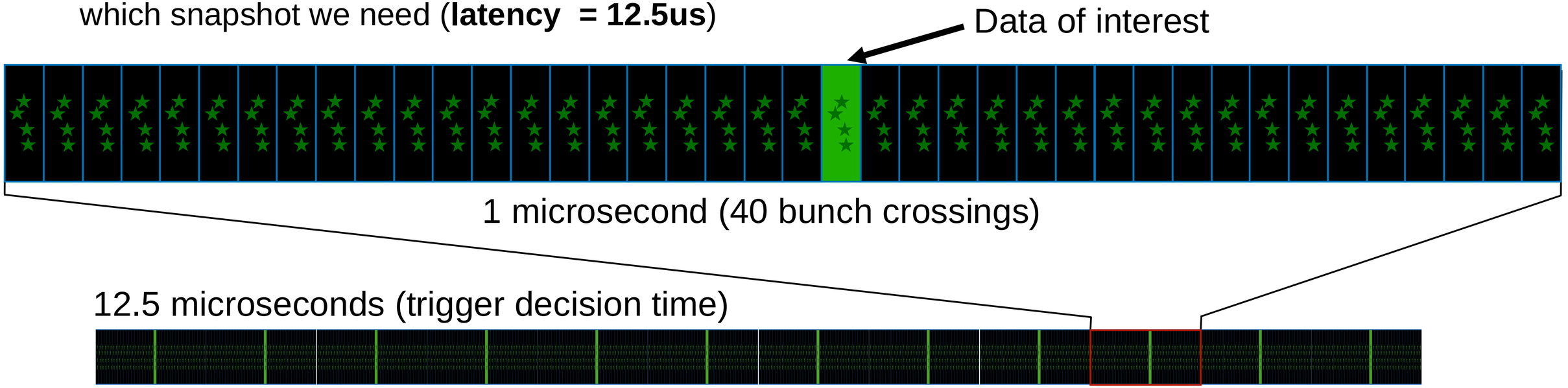


Triggering the Chip

ITkPixV2 hits are grouped into **25 nanosecond time snapshots** (LHC bunch crossing collision rate, 40MHz)

To trigger at 1MHz on average, we only want **1/40th** of all 25ns snapshots…

- But, we don't know which one it will be
- The **level 0 trigger** needs 12.5 microseconds to decide which snapshot we need (**latency = 12.5us**)



Upshot: chip memory needs to store 12.5 microseconds of data at full rate! (500 slices of 25 nanoseconds)

- To accommodate this: ITkPixV2 is **>50% memory** by transistor count

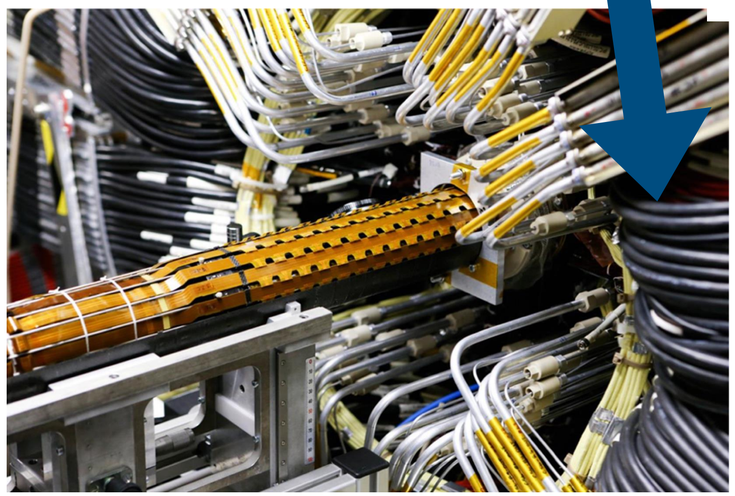
Next challenge: how do we **read out** all this data in time?

Data Readout

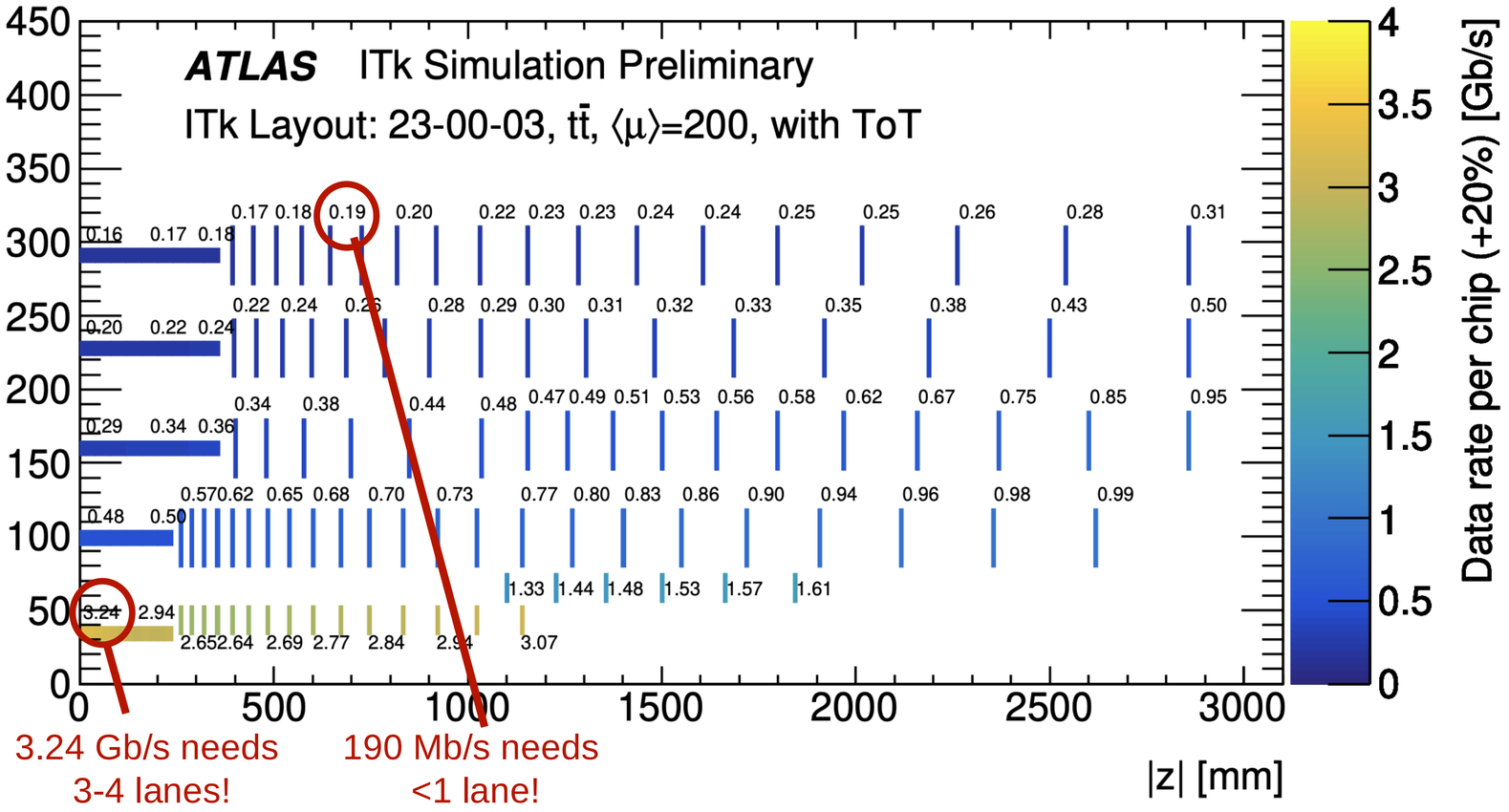
20x difference in readout rate between chips in inner/outer ITK

How **do we handle this all with one** chip?

- 4 readout channels per chip, called **lanes: 1.28 Gbps each**
- Combine multiple chips onto one lane to **minimize material**



Current ATLAS IBL during insertion



Inner pixel system simulated data rates per chip (ATL-ITK-PUB-2022-001)

Ultimately, **chip bandwidth** is limited by output bandwidth: need to transmit over **low-mass cables**, with **rad-hard drivers**

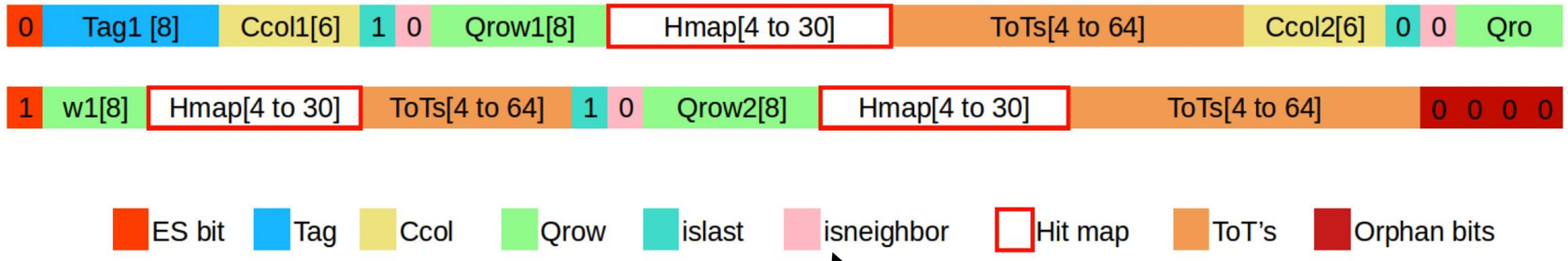
- Higher output bandwidth = more data = **more physics**

How to reduce need for output bandwidth?

- Need to design a **data compression scheme** which works equally for both **high** and **low** hit rates

Encoding Data

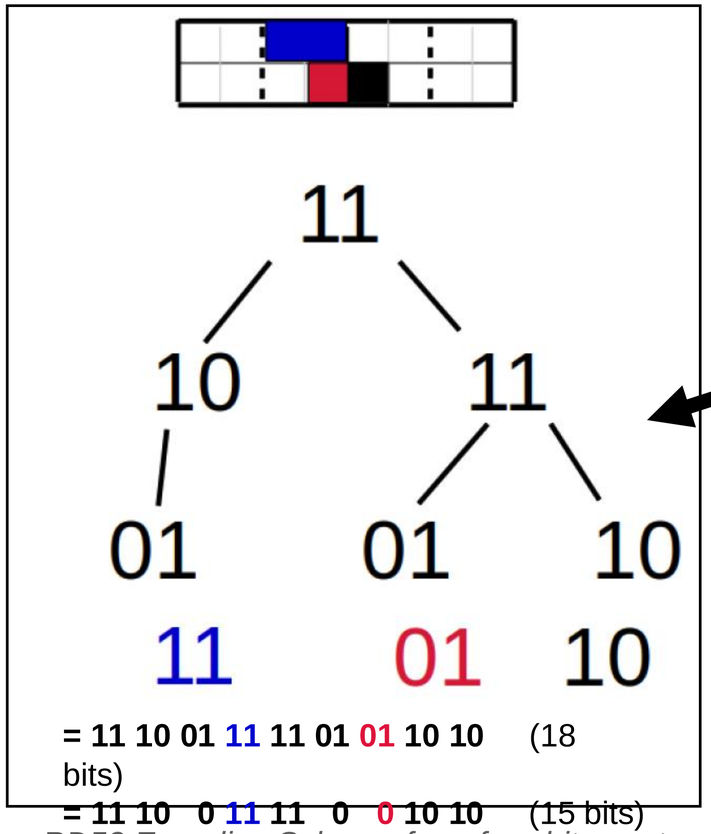
What is the best way to compress such variable data?



Encoded output for three hits, two of them being in the same core column, spanning two 64 bit data blocks

Minimum ionizing particles typically form **clusters…**

We can **drop addresses** by identifying neighboring pixel hits (“islast” and “isneighbor”)



RD53 Encoding Scheme for a four-hit event within a single quarter core

Binary tree encoding of hits in a quarter core helps for < 5 hits

01 -> 0 substitution everywhere helps as well

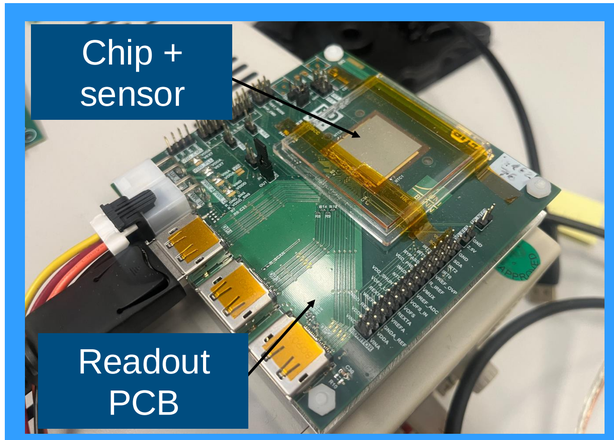
Great! But.. this data format is **very complicated**

- An event with 50 hits could range in encoded size by 2-3x
- Not **byte aligned,** so can not easily decode in FPGA → **Making a data decoder which can run in real time is an extremely challenging task**

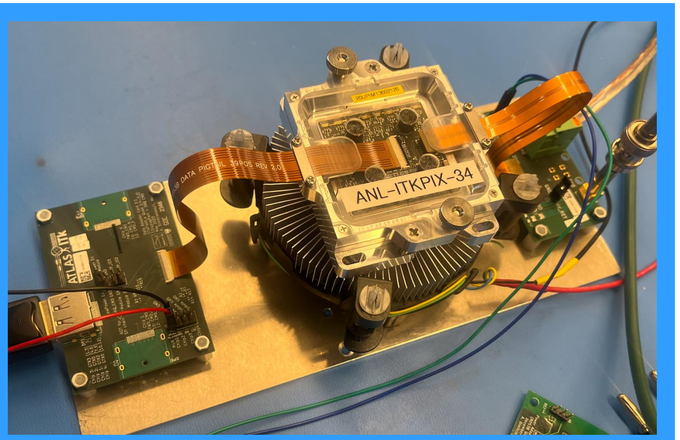
Benchtop Testing

We can already decode at rates needed for **bench testing**, using YARR readout firmware/SW

- **Single chip cards (SCCs)** and **triplet/quad modules** (3x1 and 2x2 chip arrays), at all module testing sites worldwide (~13 sq. m!)
- **Firmware:** handles triggering, fast chip communication, and data routing to DAQ PC
- **Software:** does all real-time data processing of encoded data, and provides high level control for detectors (tuning, scans)



ITkPixV2 Single chip card (SCC) being tested



ITkPixV2 Quad Module (center) with cooling, data readout card (left), and power (right)

Detector Frontends

FPGA + firmware

YARR firmware

YARR software

Readout PC + DAQ software

Even in simple environments, things can go wrong: we must handle

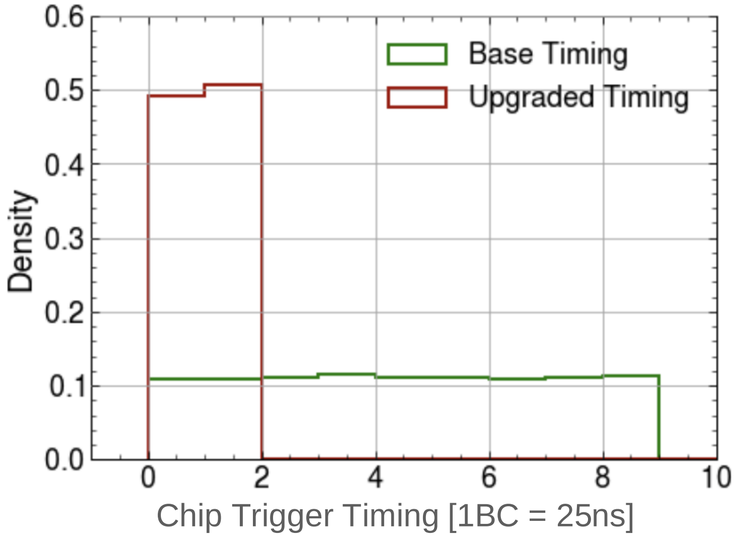
- Data transmission hiccups
- Irregular chip output data from **manufacturing** and **electrical** issues (see: core columns)
- Dead, noisy, or disconnected chip sections

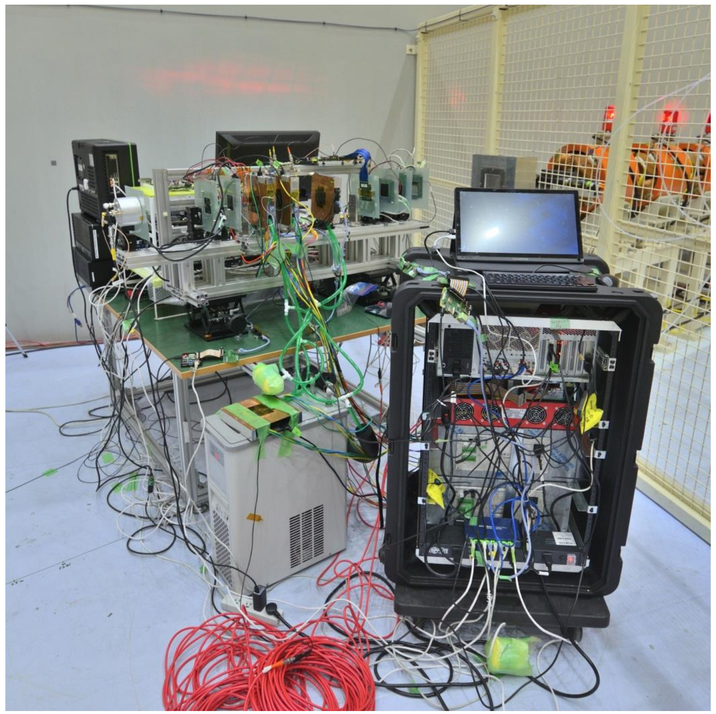
Test Beams

See results from collaborators!
- **poster** from Md Arif Abdulla Samy on Thursday
- KEK test beam with Koji Nakamura et al

A step up in output bandwidth, trigger rate, and hit rate!

- 10-100 KHz trigger rate (**10-100 times lower** than max)
- ~5 particles per chip for every trigger (**60x lower** than operation max)

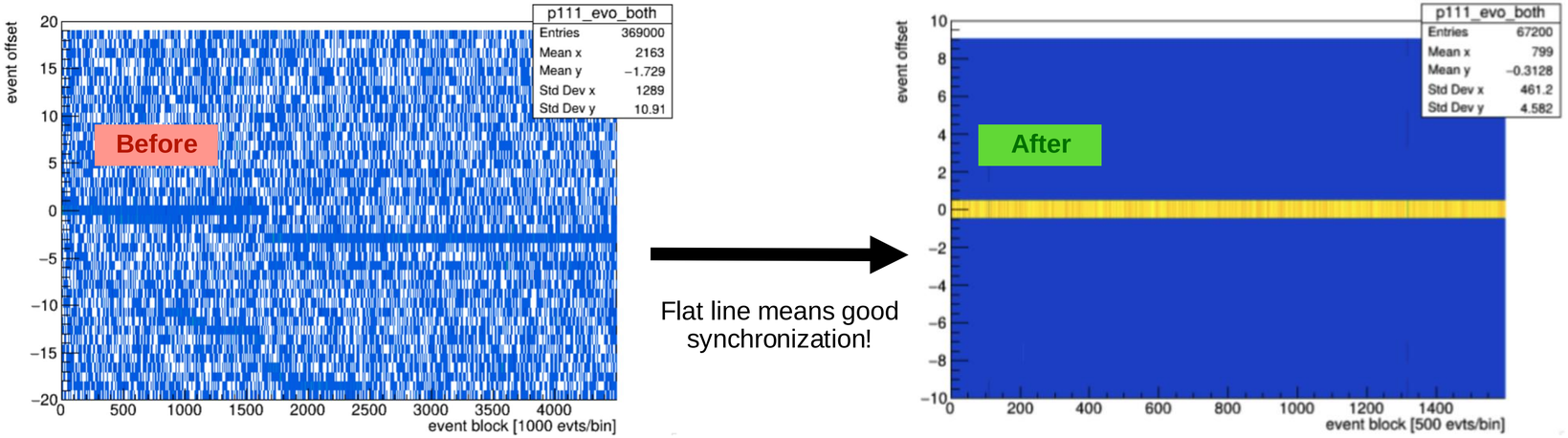




Test beam at KEK, as part of the US-Japan program to construct a test beam made up of ITkPixV2 modules only

New **readout challenges** in these environments!

- Firmware upgrades required to tighten chip **timing distribution**
- **Dropped events** in telescope planes, caused by dropped events in chip, DAQ, and transmission lines: must be fixed



Synchronization Plot in time showing desync at around 100k collected events

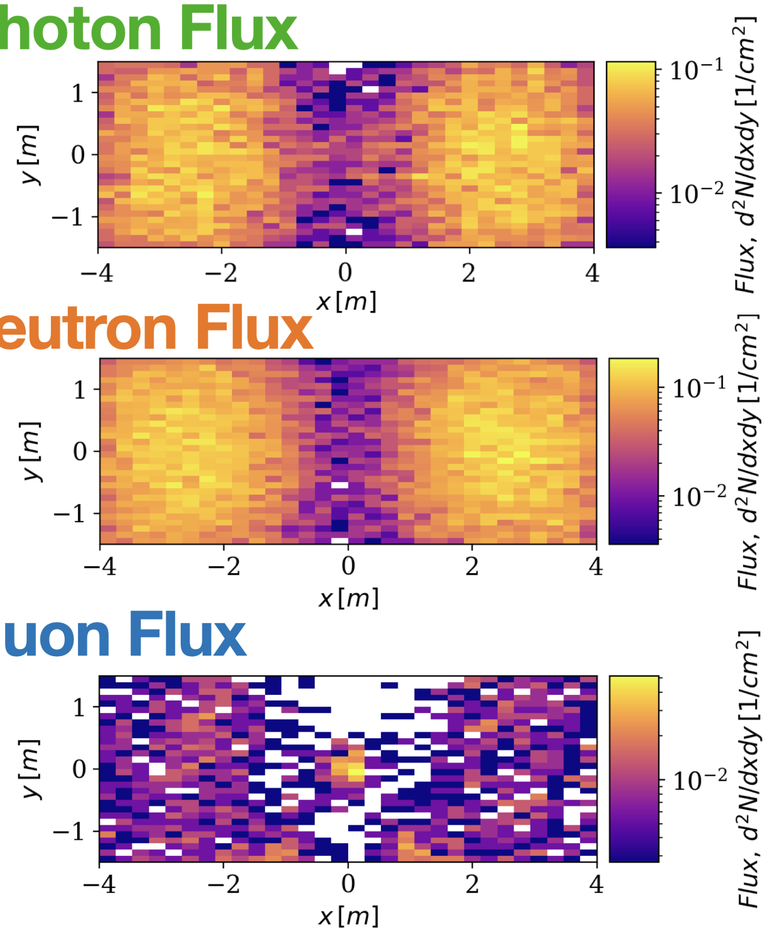
Post-fix synchronization plot

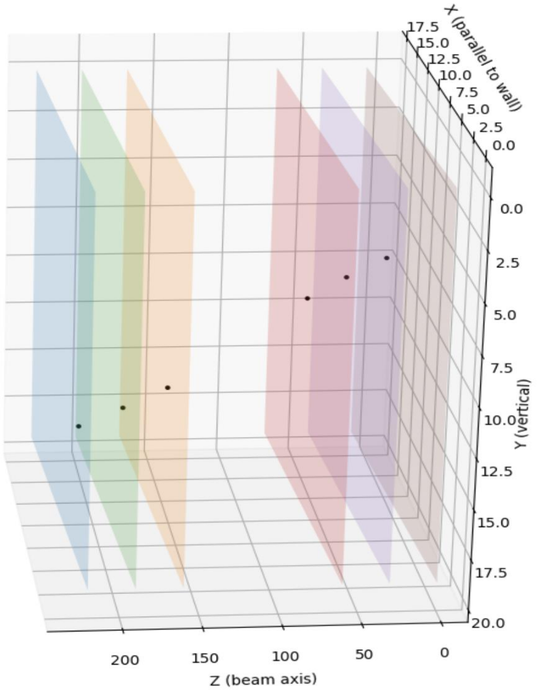
Test Beams

Other challenges at test beams…

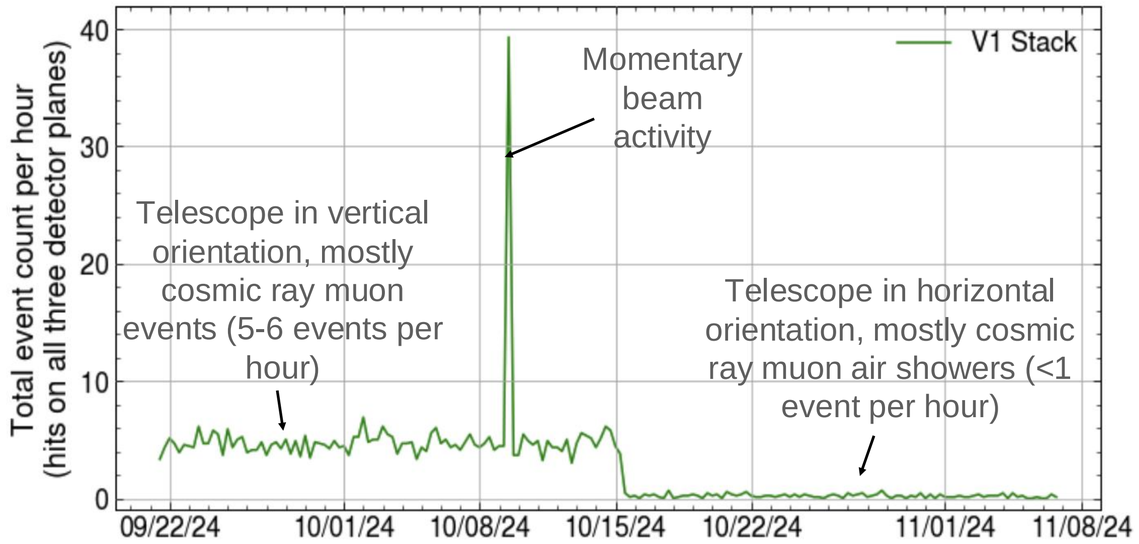
- **Long term stability** of detectors + DAQ
- Local test bench for this: telescope after 10ft concrete **beam dump** at Berkeley Lab Laser Accelerator (BELLA)
- Almost **2 months** of continuous data taking with 6-detector configuration

Simulated particle flux in BELLA target area

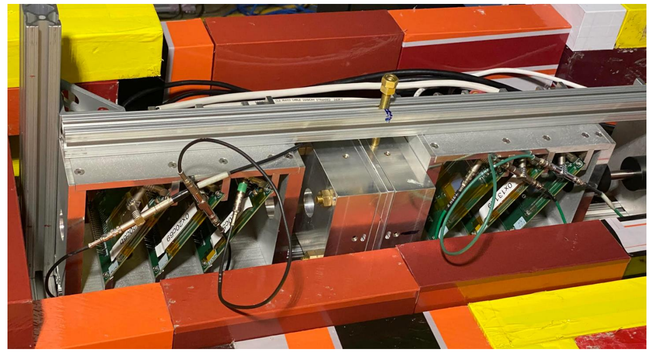




Example muon event



Collected events passing through a whole telescope stack of 3 SCCs per hour, over a 1.5 month data collection period



BELLA beam telescope with 6 ITkPixV2 Single Chip Cards

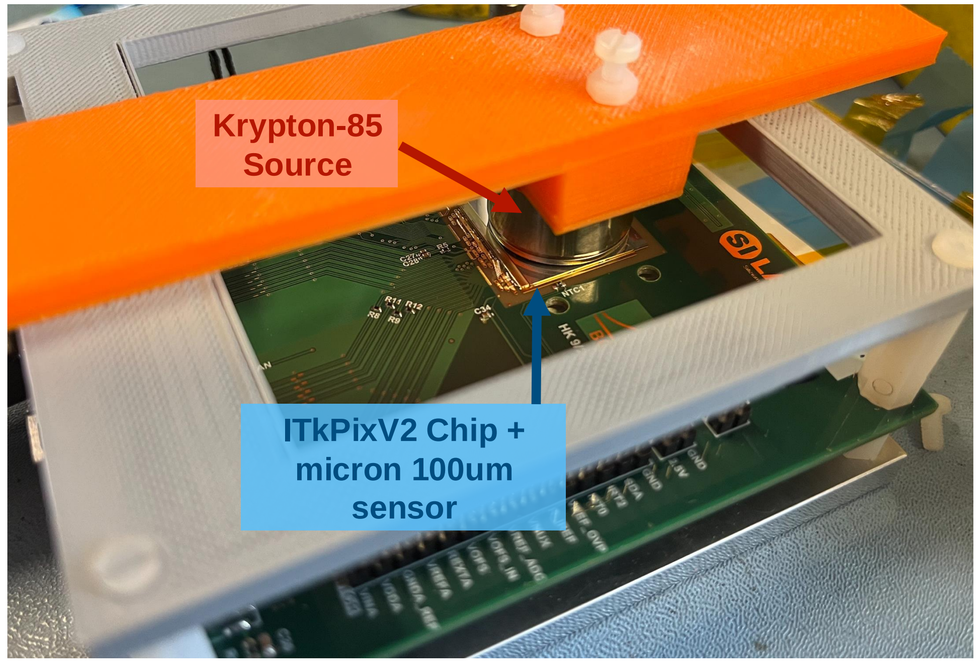
Approaching Full Rate

Despite overcoming huge DAQ challenges, tests in **this talk** so far have reached only

- **60x below** the maximum hit rate of 3GHz/cm^2
- **10x below** maximum trigger rate of 1MHz

We can visualize full-rate with a **Krypton-85 source**, placed on a single chip card

- Gaseous source with spherical container, means spherical radiation pattern
- Collect the amount of data the chip would process in **1 second of operation**, at the **closest layer to the collision point**

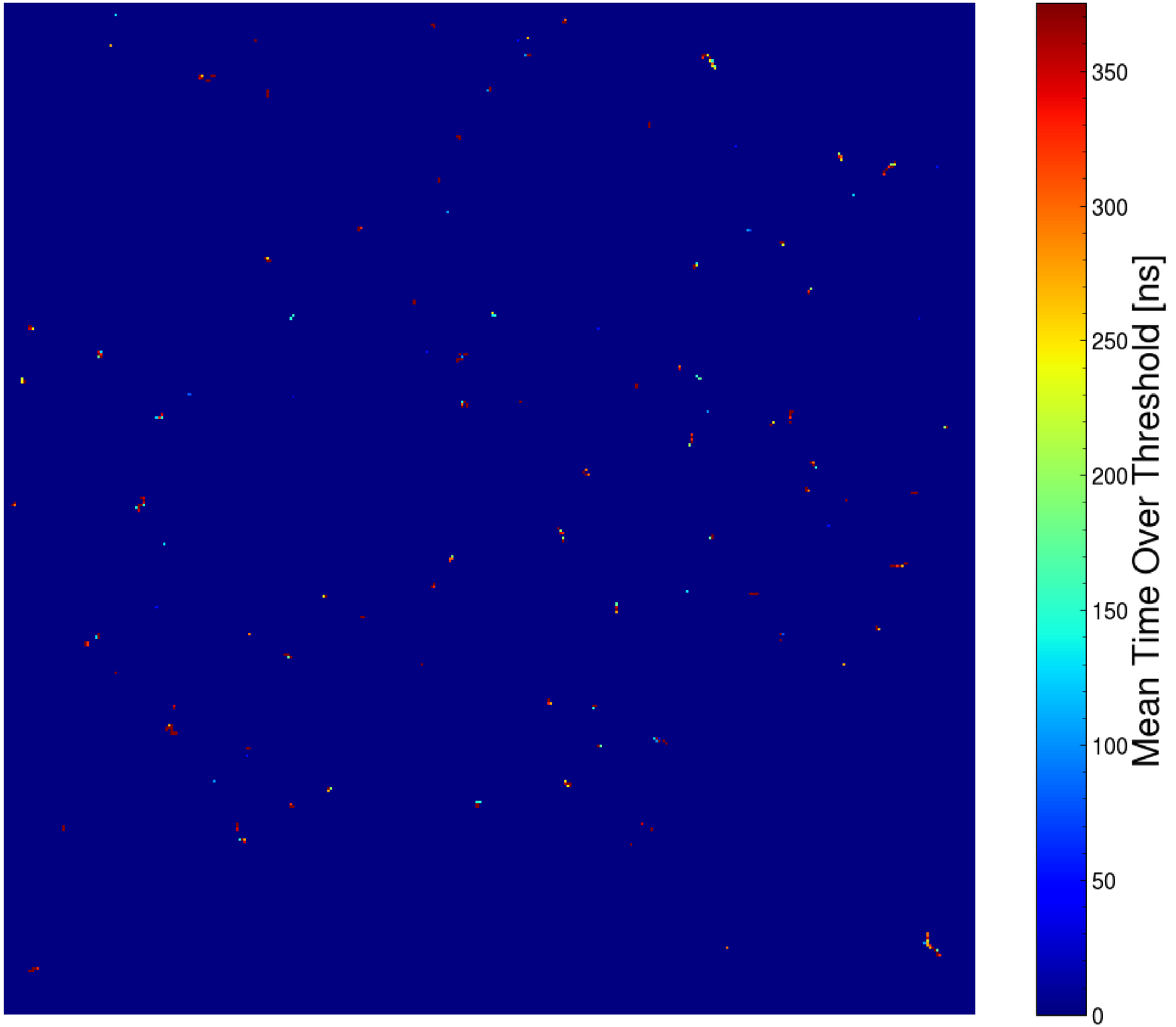


Source test setup, with Krypton-85 source placed directly above ITkPixV2 chip + sensor

- Ignoring **single event upsets (SEUs)**, a significant challenge for the real detector

What does this look like?

1 bunch crossing at full rate



313 hits

25 nanoseconds

2x2 cm sensor

Krypton-85 data

1 second at full rate

12.6 billion hits

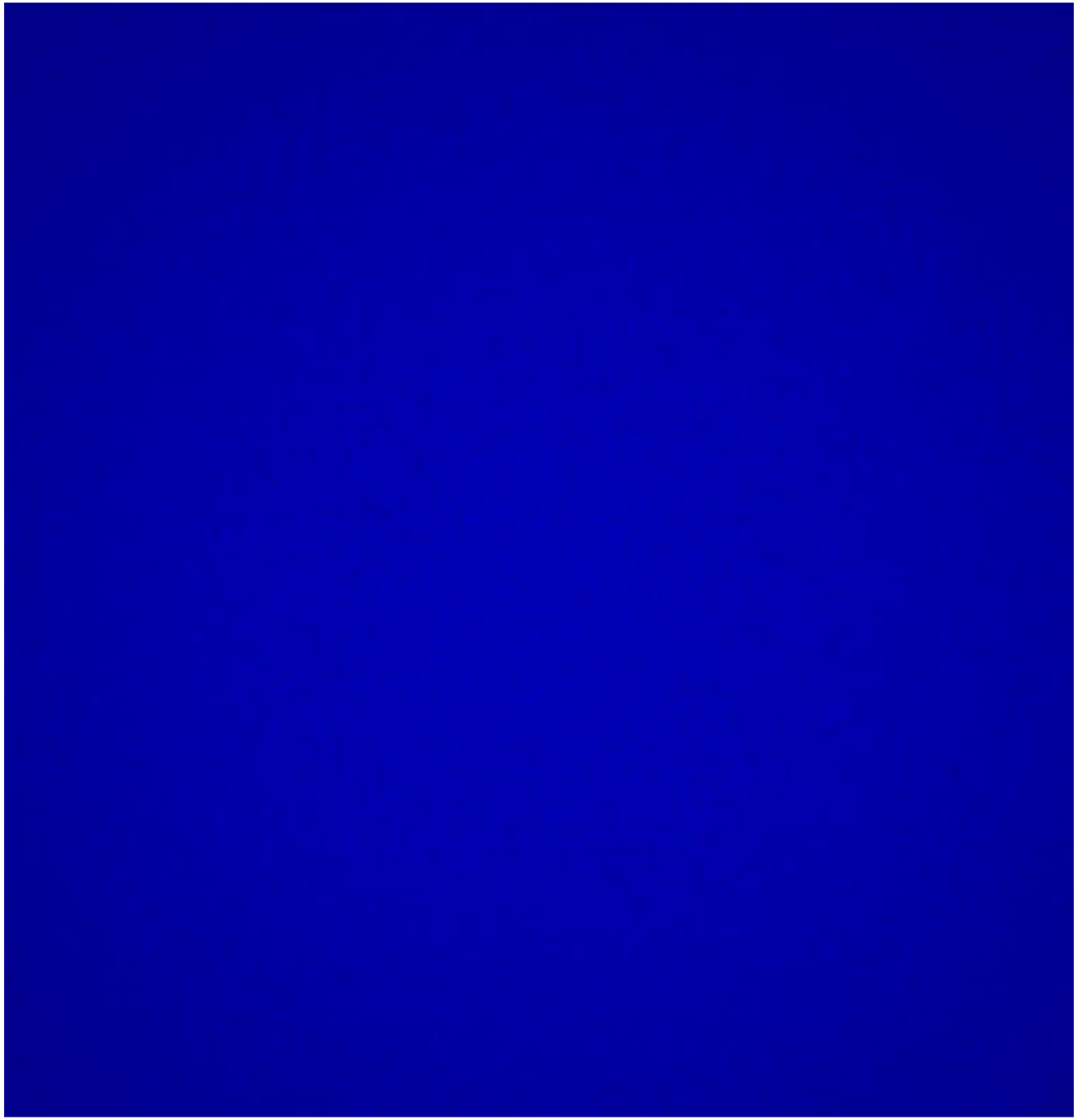
40MHz event rate

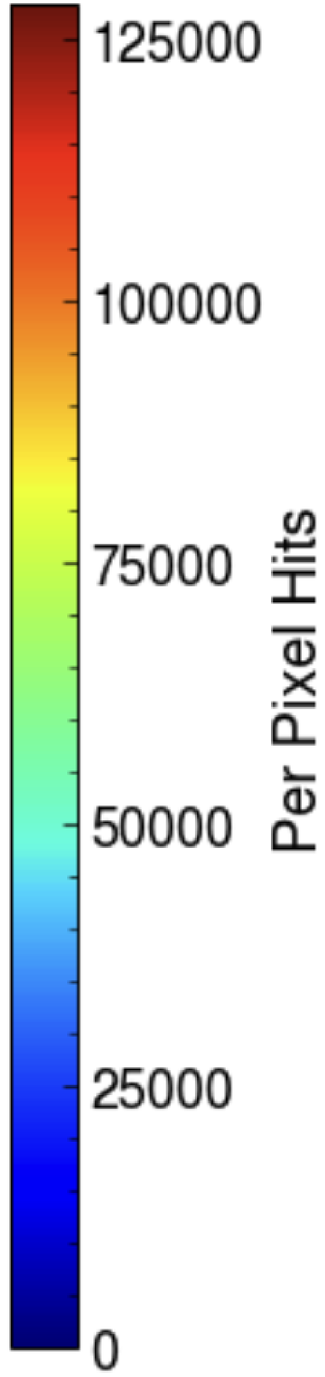
3.276 GHz/cm²

2x2 cm sensor

realtime

Krypton-85 data





too fast?

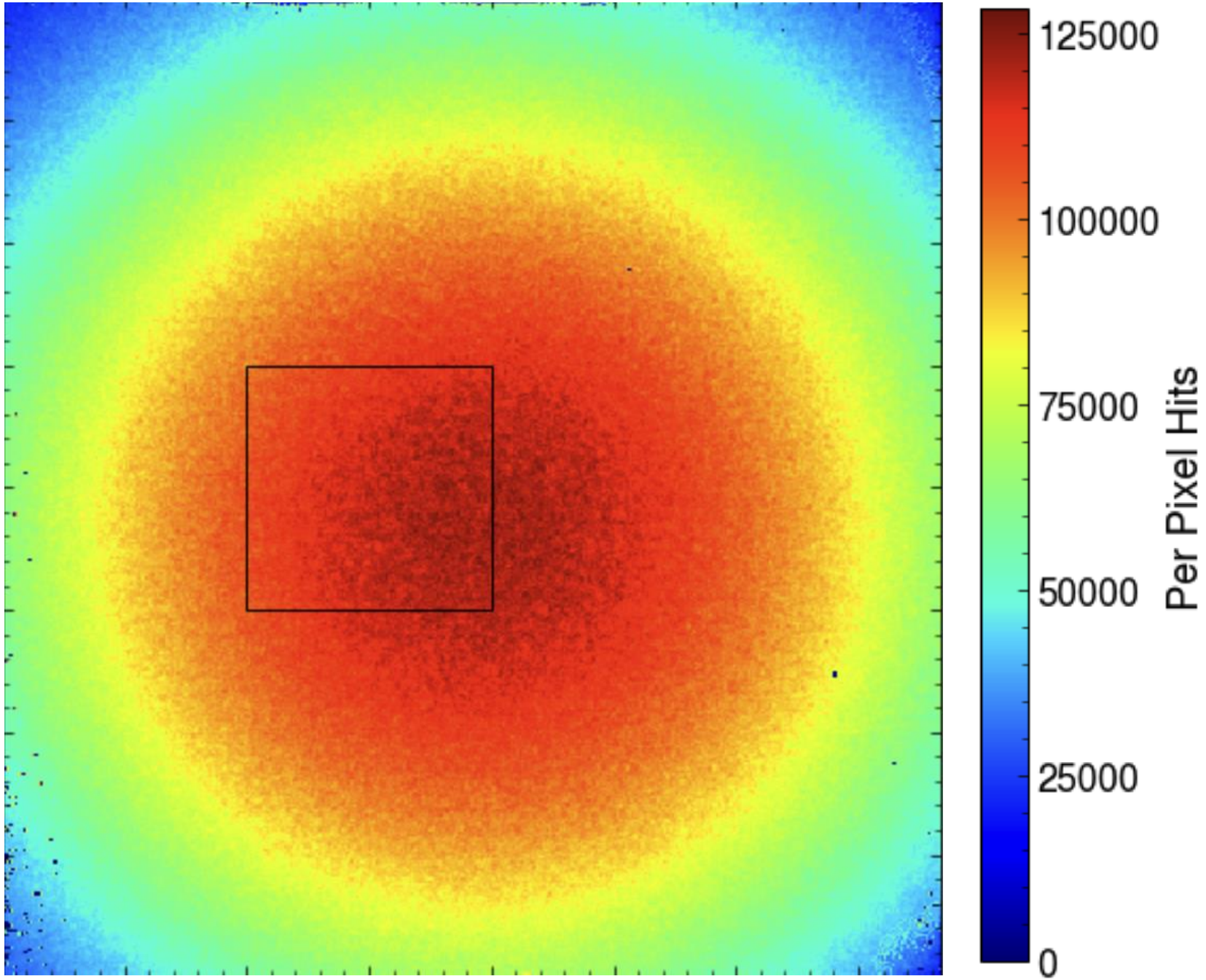
… then take a subset

100x100 pixel subregion

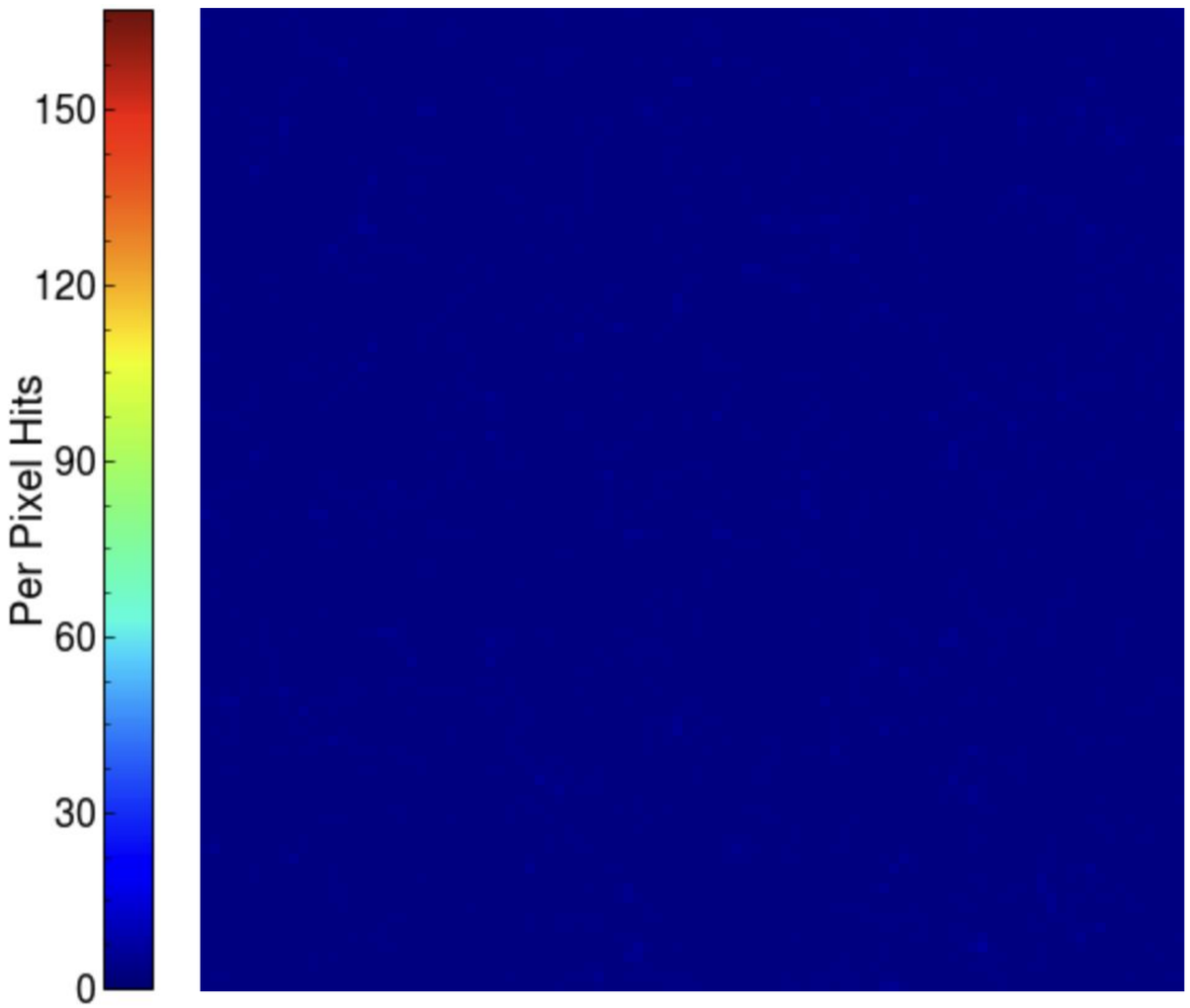
1/15th the area

… and slow it down

1 millisecond



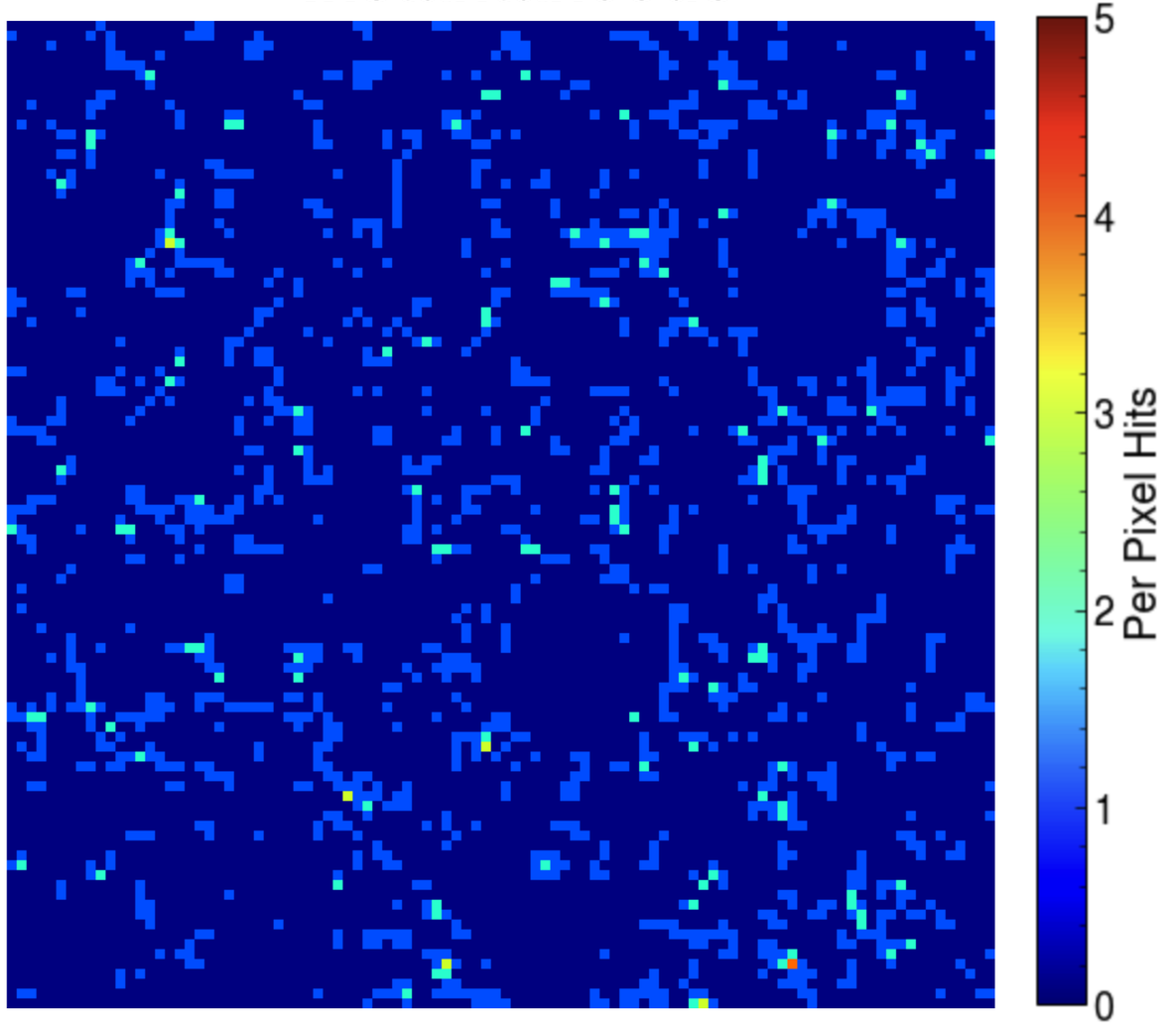
Cumulative



1 millisecond

850 thousand hits

Instantaneous



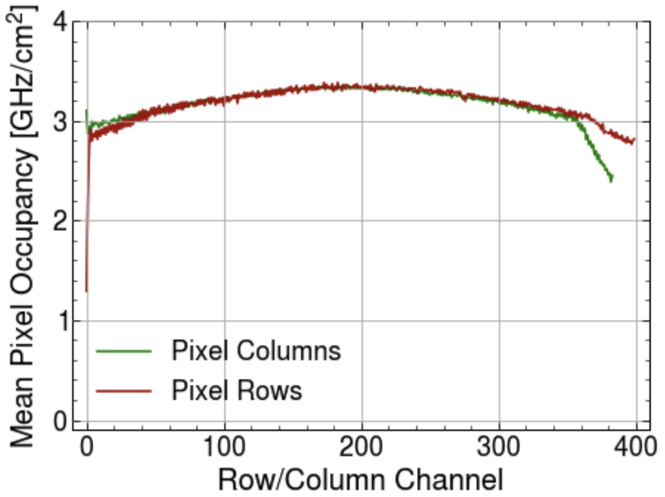
1.5μs per frame

3.4 GHz/cm^2

Testing at Full Rate

We need to **validate our chip design**

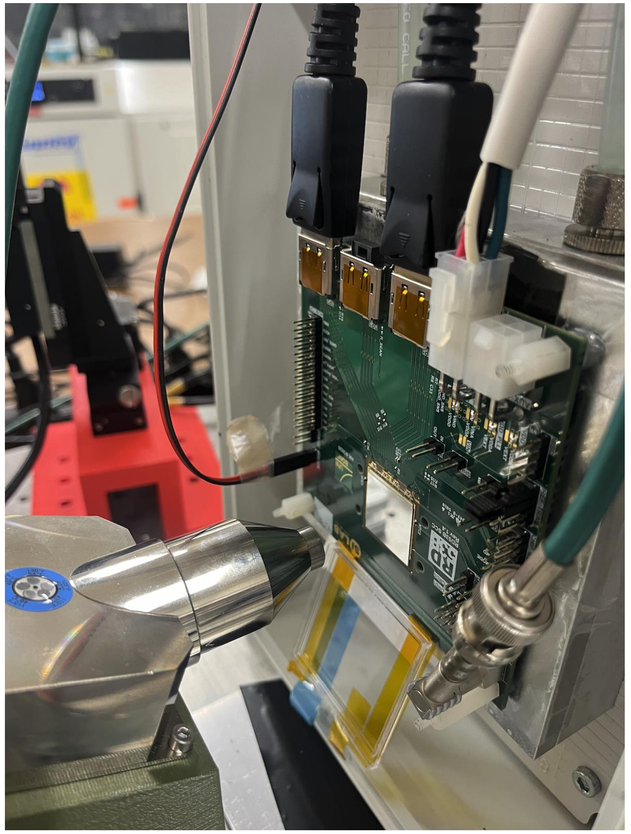
- But, impossible to recreate the “real” environment in every way, without actually building ITK



Mean hitrate for ~3GHz rate, showing close-to-uniform occupancy across the chip

How can we test at this extreme rate?

- Isolate specific tests we want to do, and push the chip



X-ray tube aligned with chip high-rate studies

Simple full-rate measurement:

- Digital **current consumption** of the chip as a function of **hit rate**
- Critical measurement for **cooling budget of ITK:** more current = more heat

Setup:

- To reach ultra-high hit rates: use **X-ray tube** (Amptek mini-x2)
- Make precision current, temperature, etc measurements during tests, triggering at **1MHz**

Thank you to Jennifer Ott/Simone Mazza at University of Hawaii/SCIPP for providing x-rays!

Expectation: **exactly linear relationship** between hit rate and current

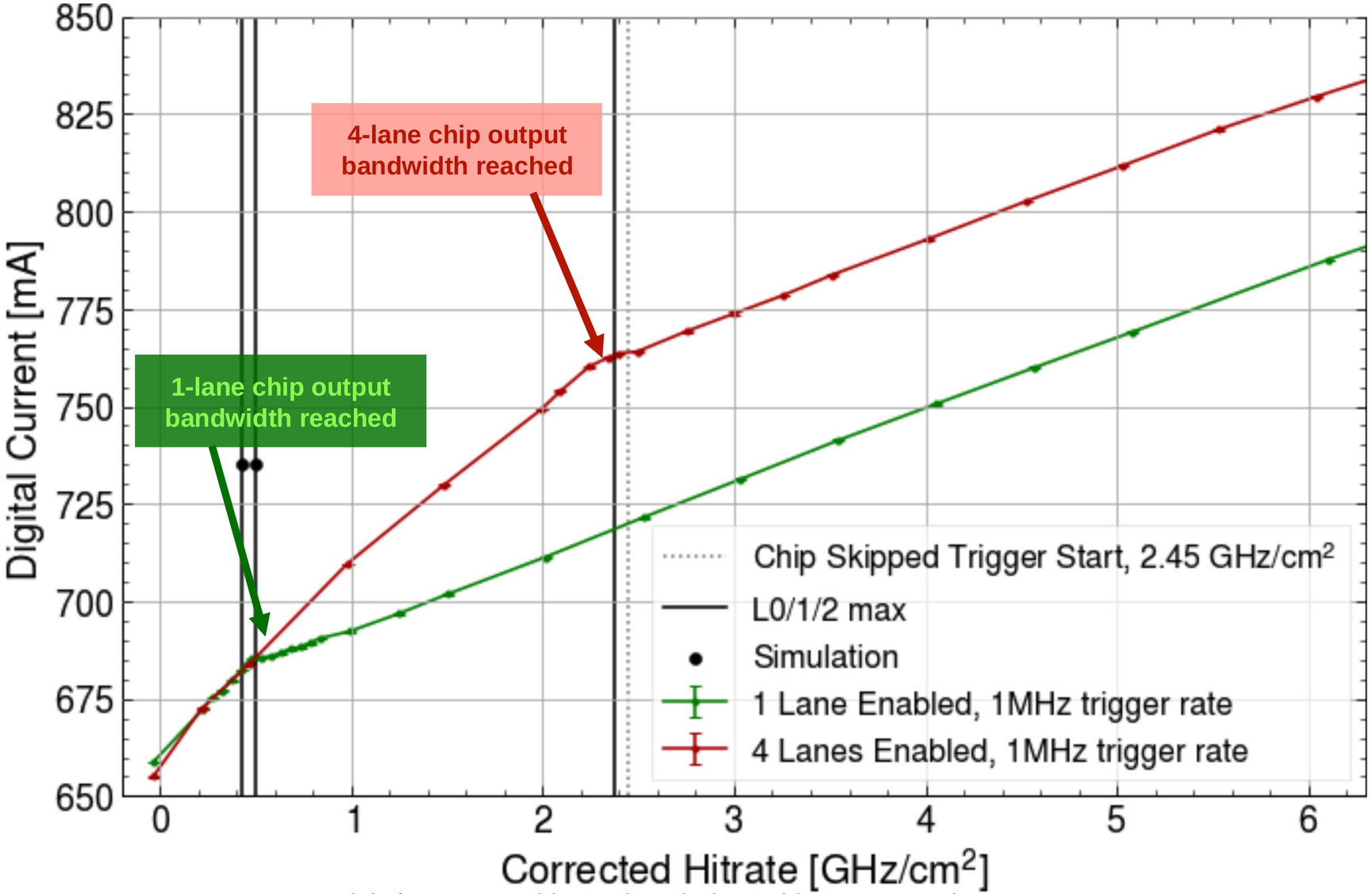
Digital Current Measurement

As expected, slope is linear in hit rate

Can clearly see a “knee” at 0.45 GHz/cm^2

- Knee at 0.5GHz /cm^2 : **1 lane bandwidth limit** of 1.28Gbps
- Knee at 2.5GHz/cm^2: **4 lane bandwidth limit** of 5.12 Gbps

After this limit, lower slope: only seeing **pixel matrix** current (counting of hits, ToT)



Digital current vs. hit rate for ITkPixV2 chip + 100um micron sensor

Why do we need to “correct” the hit rate?

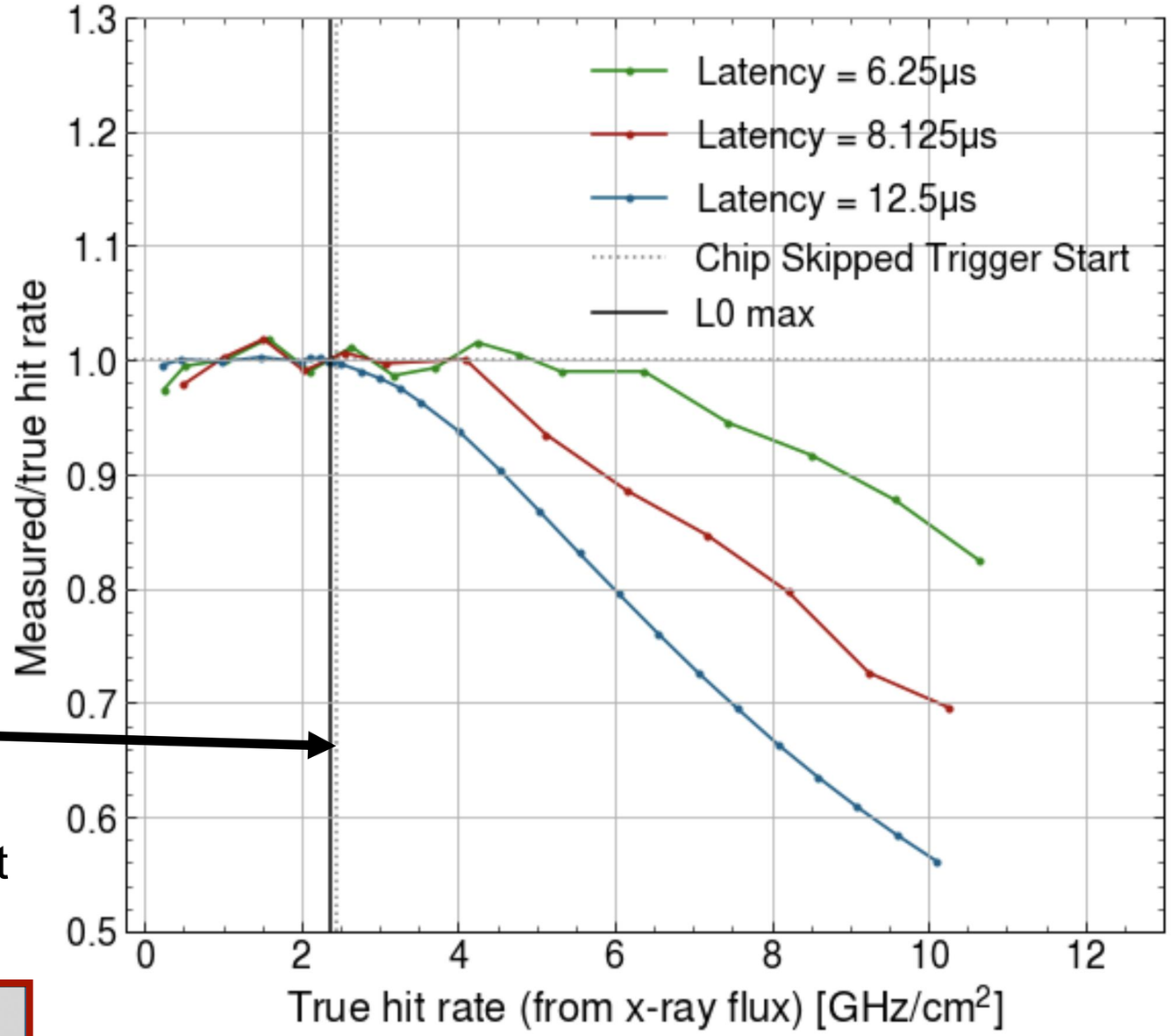
Latency and Pixel Hit Memory Overflow

Latency has a huge impact on the amount of lost hits!

- Dropped hits are caused by **memory overflow**
- Higher latency: hits have to stay in memory longer before they are read

Skipped Triggers

- We start to see “skipped triggers” from the chip at only 2.45 GHz/cm^2…
- This is the point where the chip “can’t keep up” anymore with the combined trigger and hit rate: design spec is 99% efficiency @3GHz



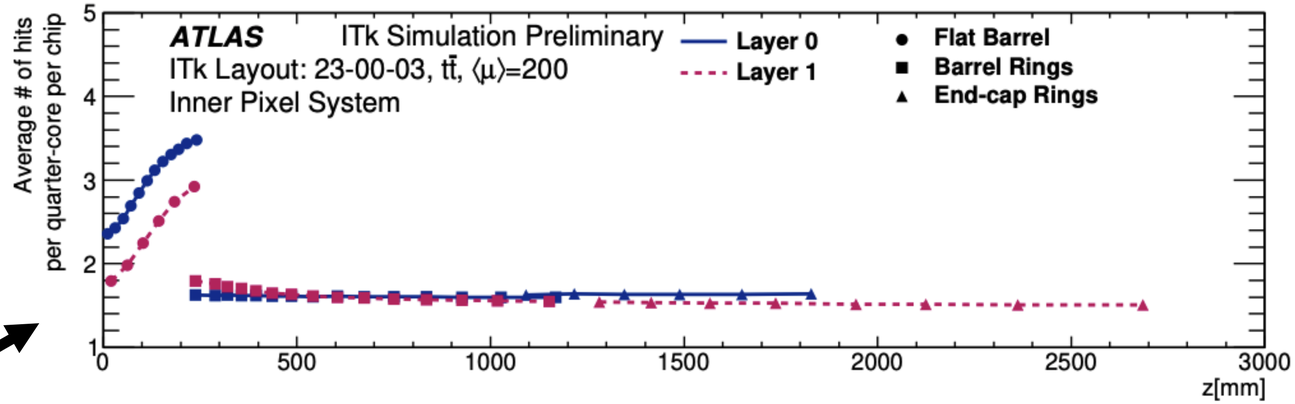
But what about our design requirements?

Hit Rate	0.4 GHz / cm^2	3 GHz / cm^2
Trigger Rate	0.1 MHz	1 MHz

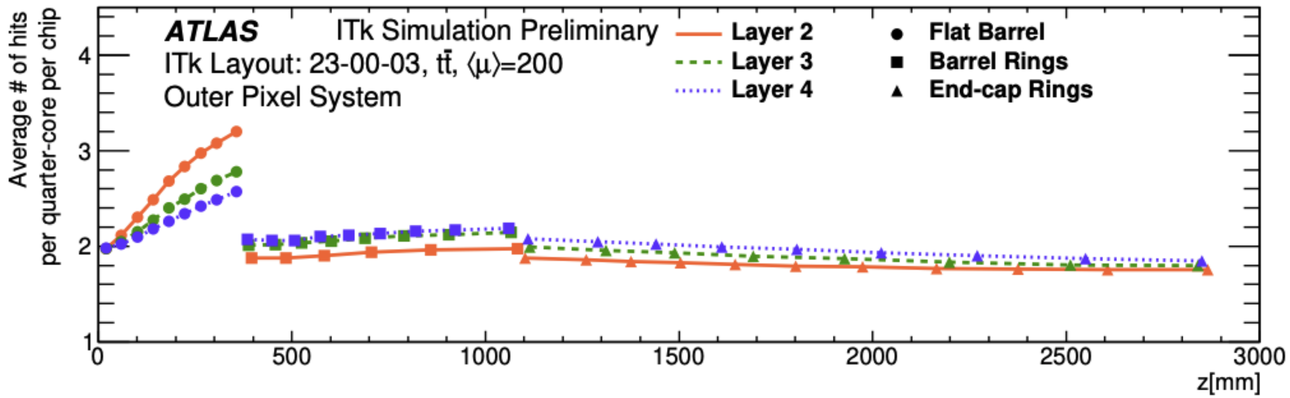
Maximum Rate

Short answer: we are ok

- X-rays are typically **single pixel hits**, which compress less from our encoding than **clusters** (slide 5 and **paper study**)
- In the real detector, we expect to see **mostly clusters**
  - This is the whole point of our chip encoding design!
- X-rays provide the **worst possible encoding case**, at about **1.8 Gbps / GHz/cm²** (clusters are closer to 1Gbps/GHz/cm² - see backup)
  - Chip from this test has a bandwidth limit at ~**4.5 Gbps** -> **within safety factor!**



(a) Inner Pixel System



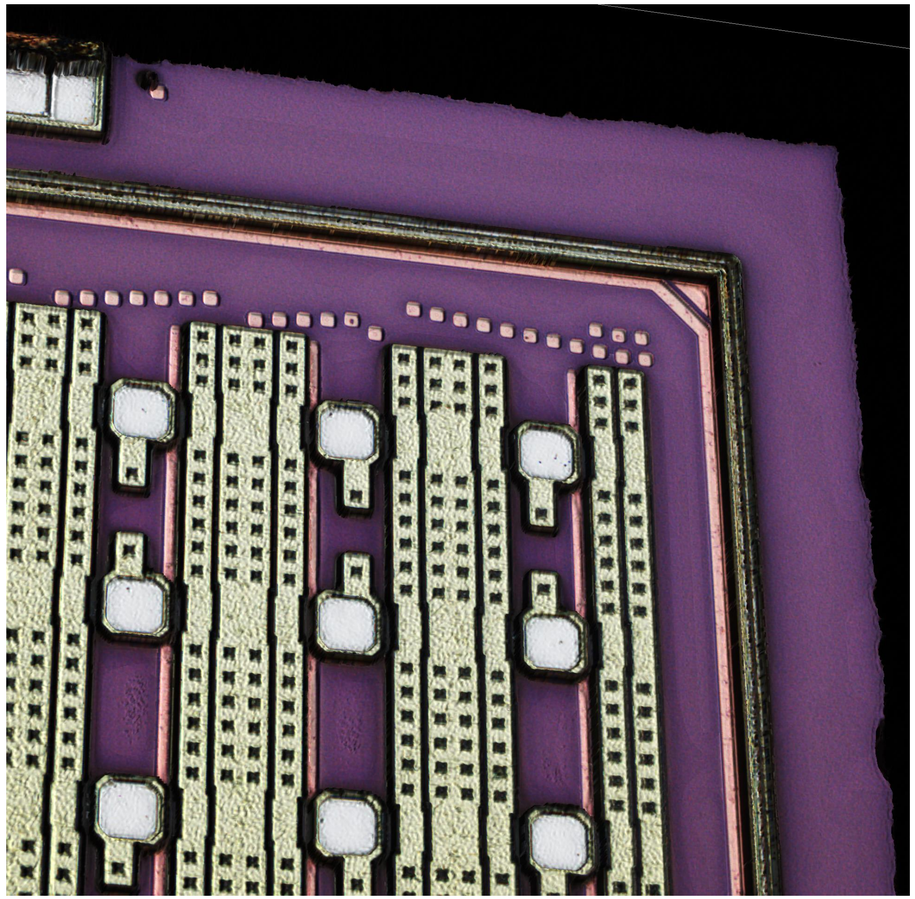
(b) Outer Pixel System

Simulated number of hits on average per hit quarter core for (a) inner pixel system and (b) outer pixel system

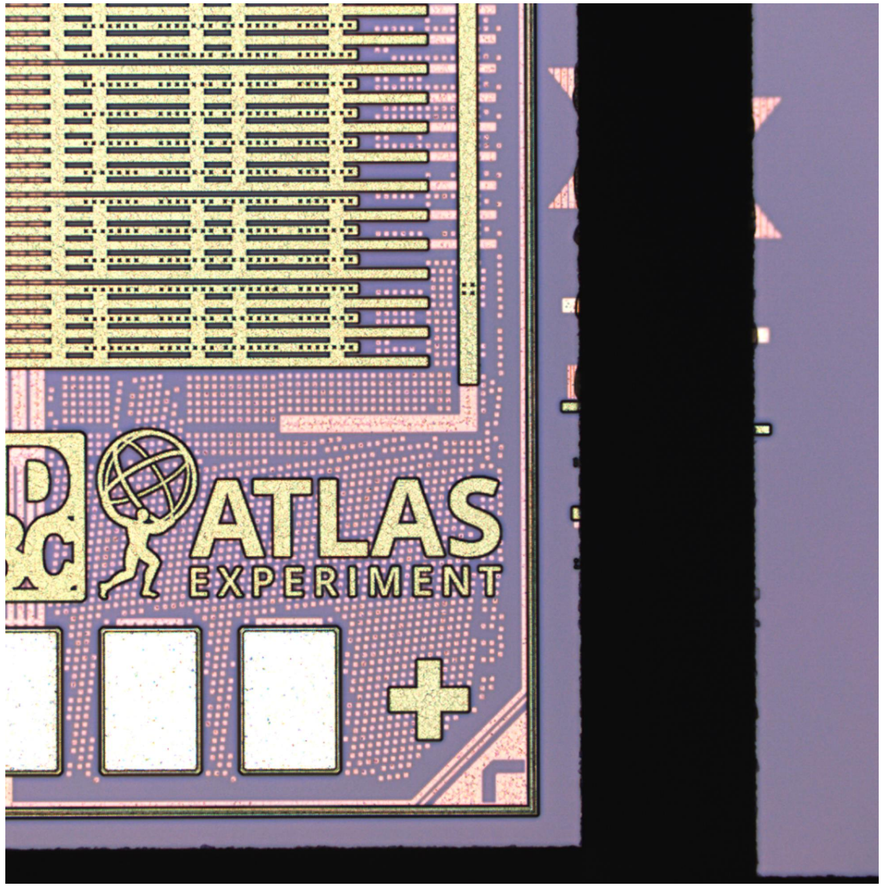
Take away

Measurements like these are critical:

- Helps to **verify our hardware**
- Builds our experience with how our detector will **function** during the HL-LHC



Diced wafer chip edge with pixel bonding pads and cut edge
(credit to Maurice Garcia-Sciveres)



Diced wafer chip edge with wire bond pads (bottom left) and digital chip bottom (top left)
(credit to Maurice Garcia-Sciveres)

Even simple measurements at our operational conditions are **extremely complicated** to make correctly! To truly measure everything at once, we need HL-LHC

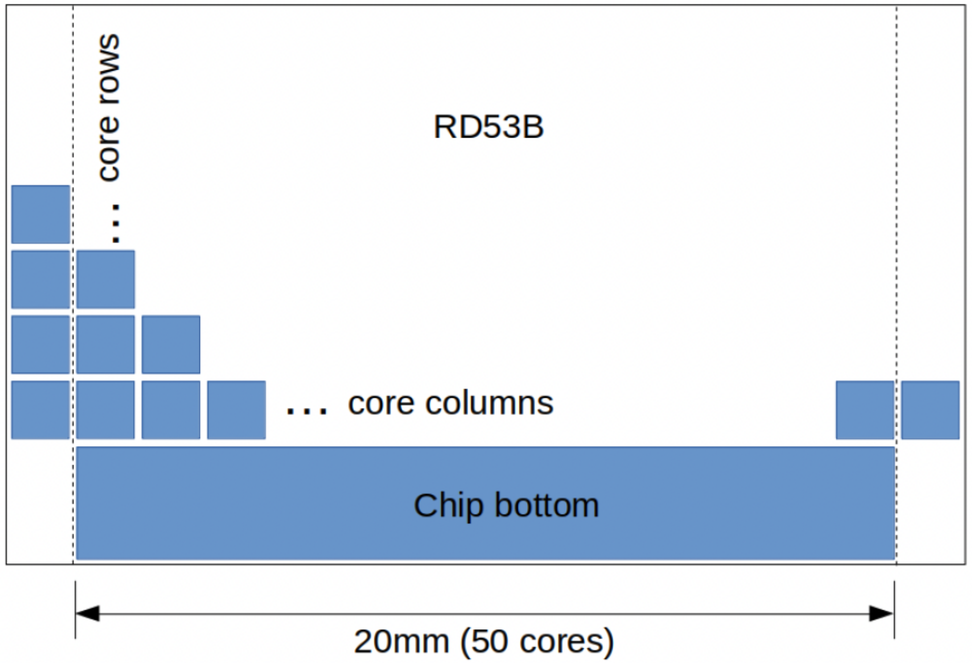
Lots more work to do – but **ITkPixV2 is working well so far**

Questions

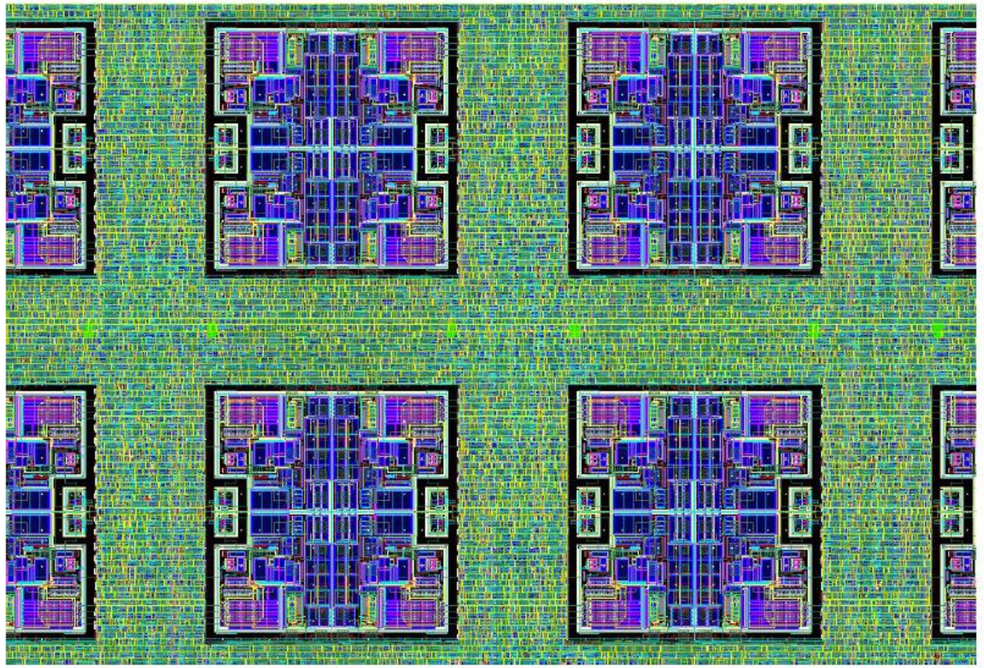
Backup

Chip Design

- ATLAS consists of 48 core rows x 50 core columns, for 384 x 400 pixels
- Each pixel core consists of 8x8 pixels, and contains four *analog islands*
- Pixel matrix contains repeated pixel cores, interconnected for config and readout



Conceptual depiction of RD53C Chip Framework,
from the ATLAS RD53C manual

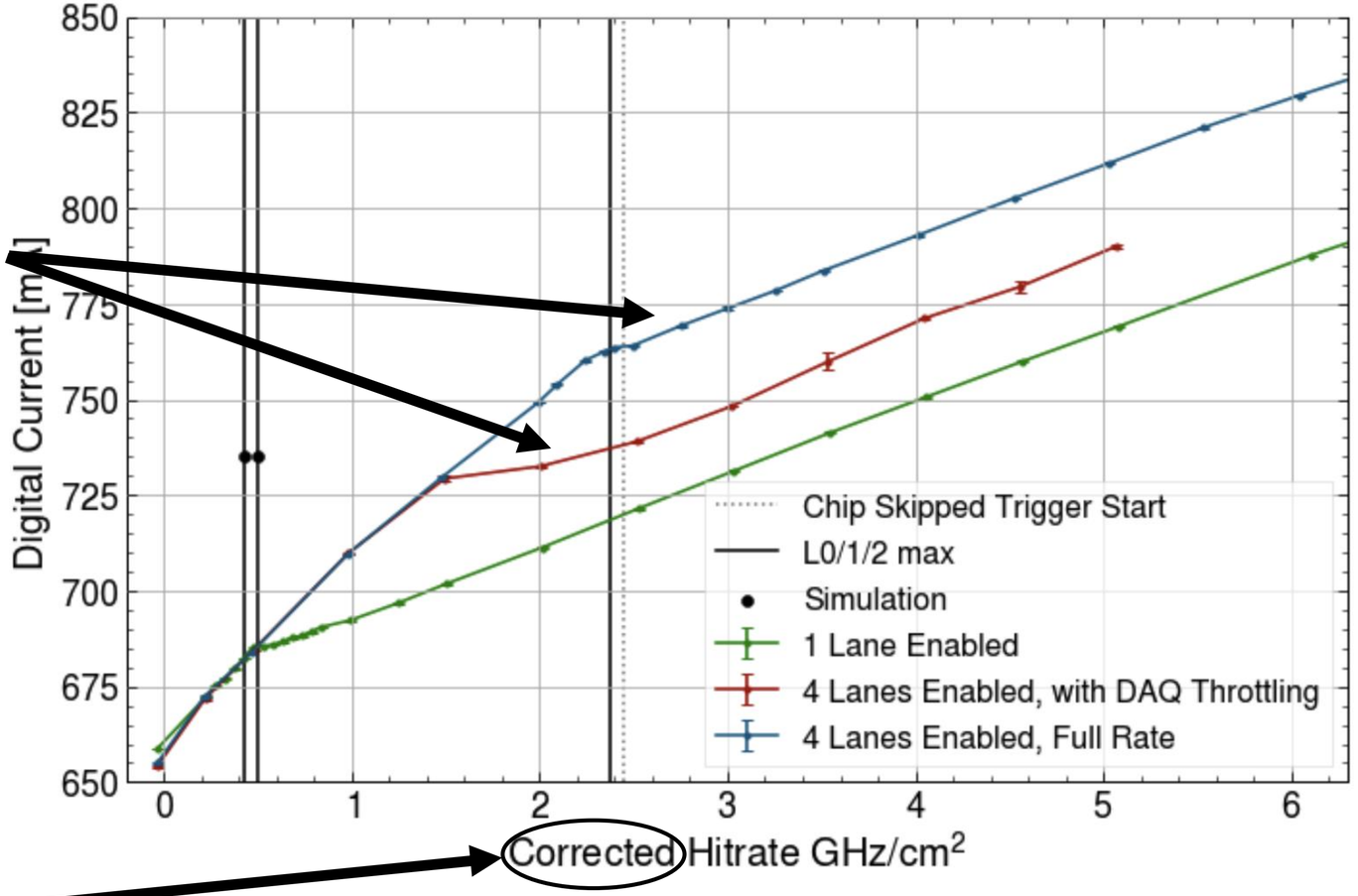


Layout of four complete analog islands within chip digital logic,
from the ATLAS RD53C manual

Knee FAQ

Why does current keep increasing after the “knees”?

- The pixel matrix is still **counting** and **processing** hits, even though maximum output bandwidth is throttled
- This means: the post-knee slope gives an **estimate of the raw pixel current consumption**

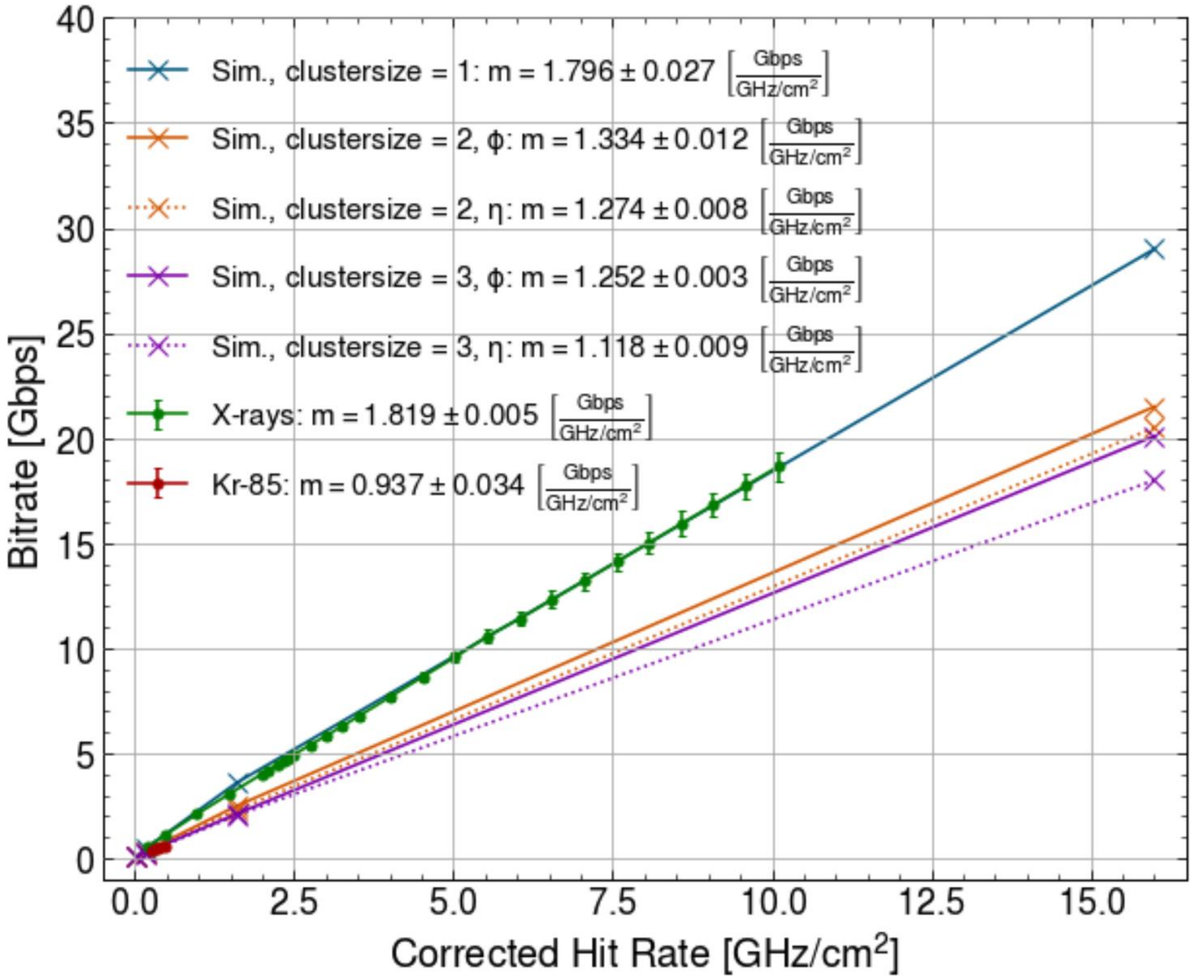


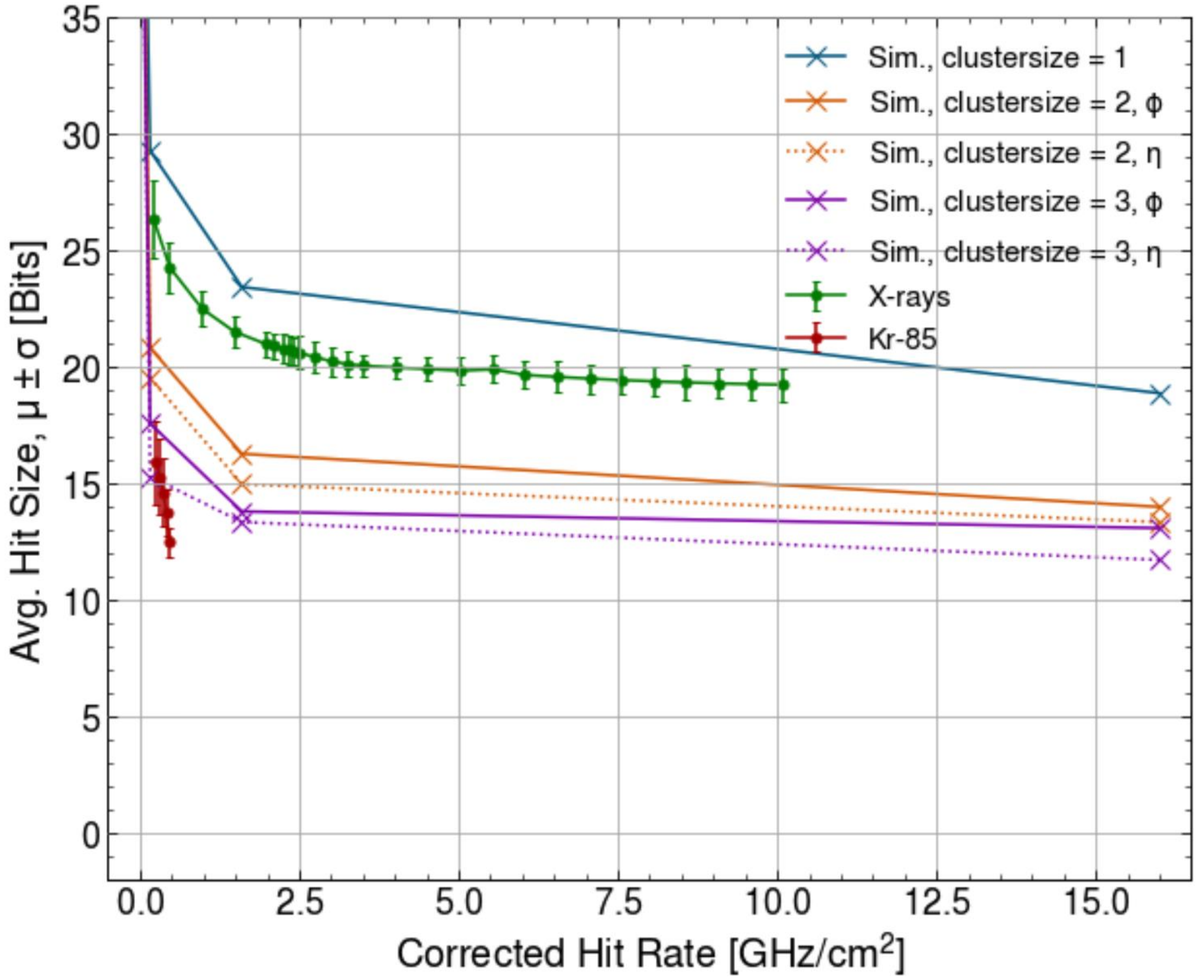
What does “corrected” hitrate mean?

- The **chip memory** is finite, meaning it will fill **faster and faster** as the hit rate increases
- At some point, new hits will begin to overwrite old hits before they can be read
- **Latency:** controls how far back we look in chip memory, and should change our “**correction factor**”

Bitrate vs. Hit rate

We can re-encode our raw data to obtain bitrate as a function of hitrate, using ITkPix encoder produced by Ondrej Kovanda for x-ray + krypton datasets. We can also plot simulated values from **Rd53 testing meeting**

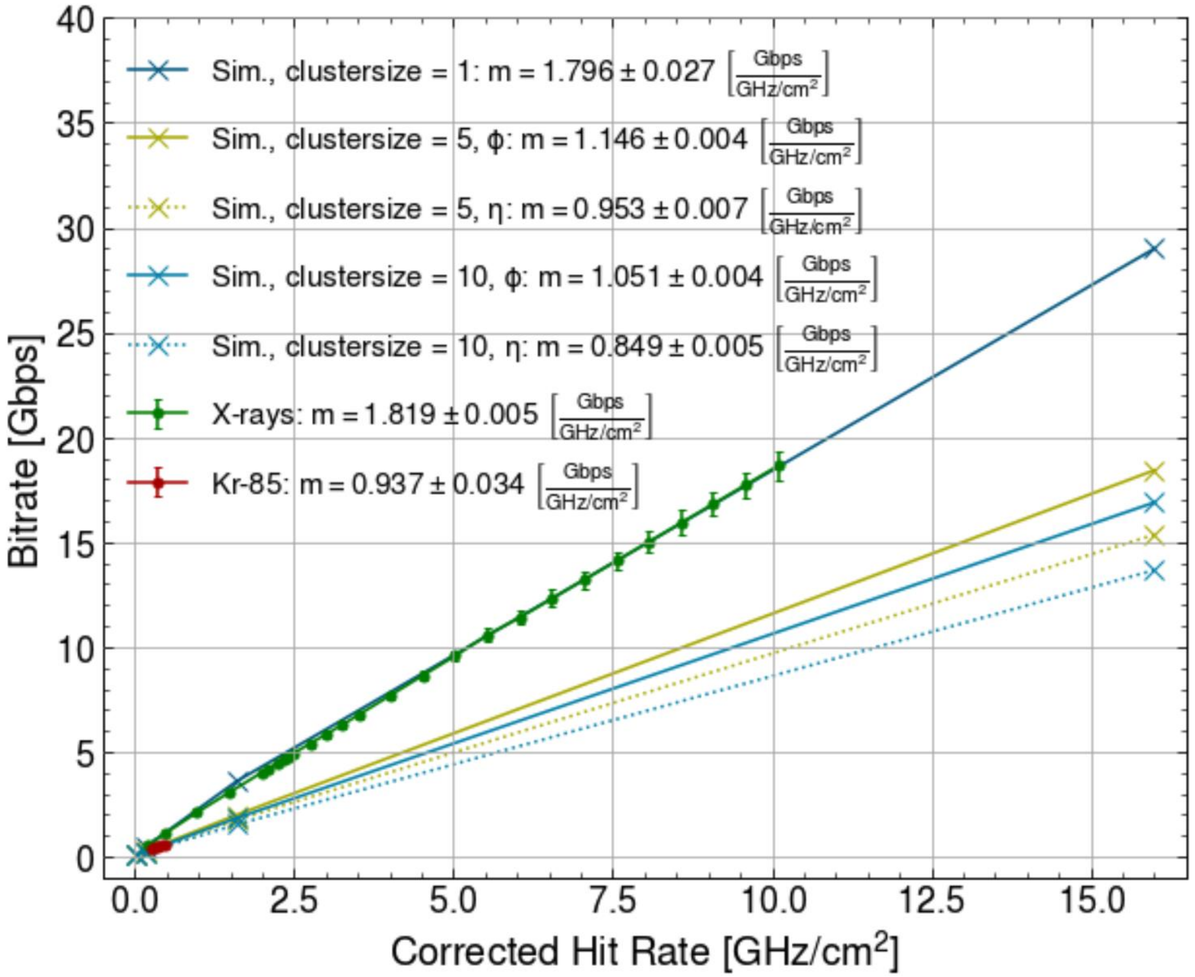


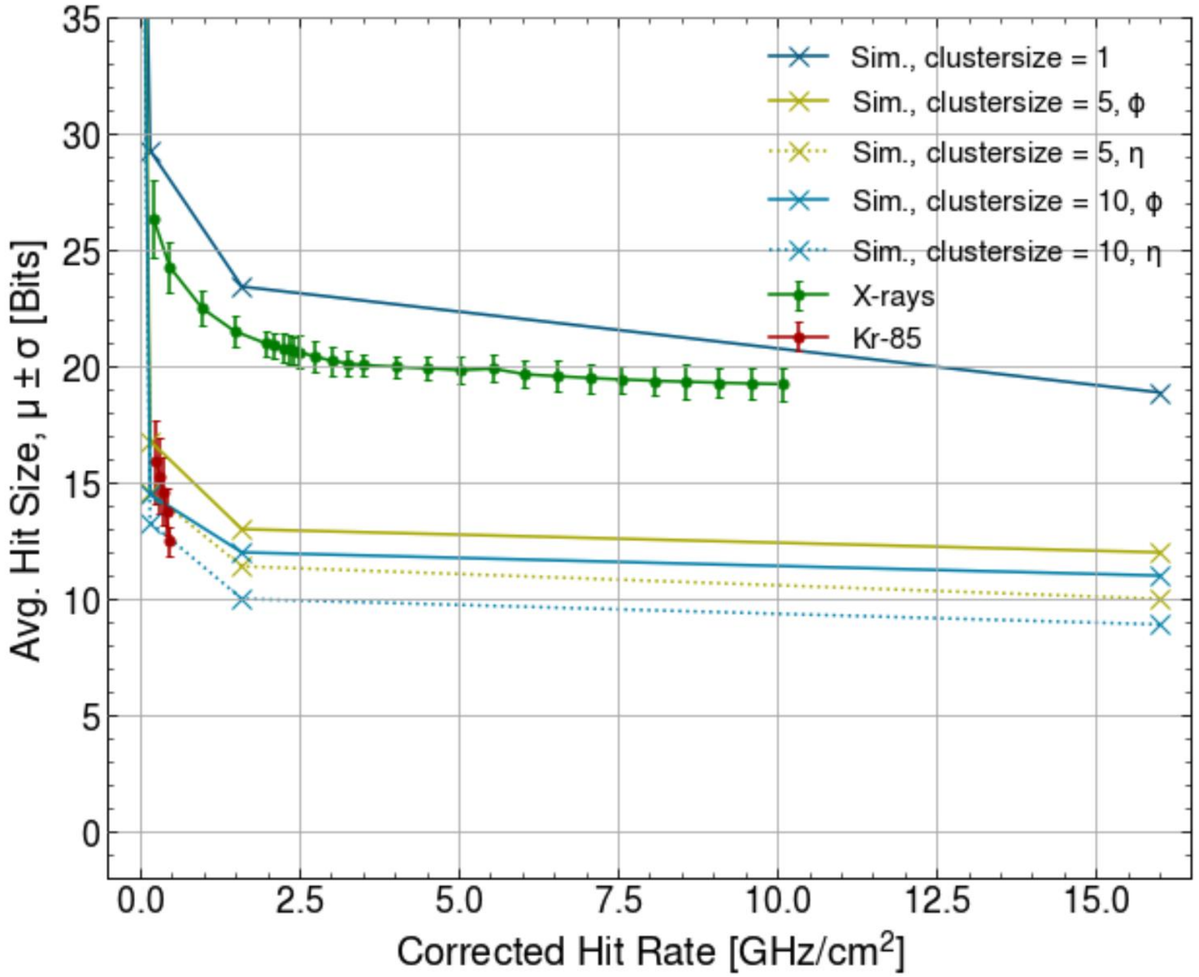


Bitrate (left) and average hit size (right) as a function of hit rate for Krypton source, x-ray source, and simulation values for cluster sizes of 1, 2, and 3 pixels in both Eta and Phi directions.

Bitrate vs. Hitrate

We can re-encode our raw data to obtain bitrate as a function of hitrate, using ITkPix encoder produced by Ondrej Kovanda for x-ray + krypton datasets. We can also plot simulated values from **Rd53 testing meeting**

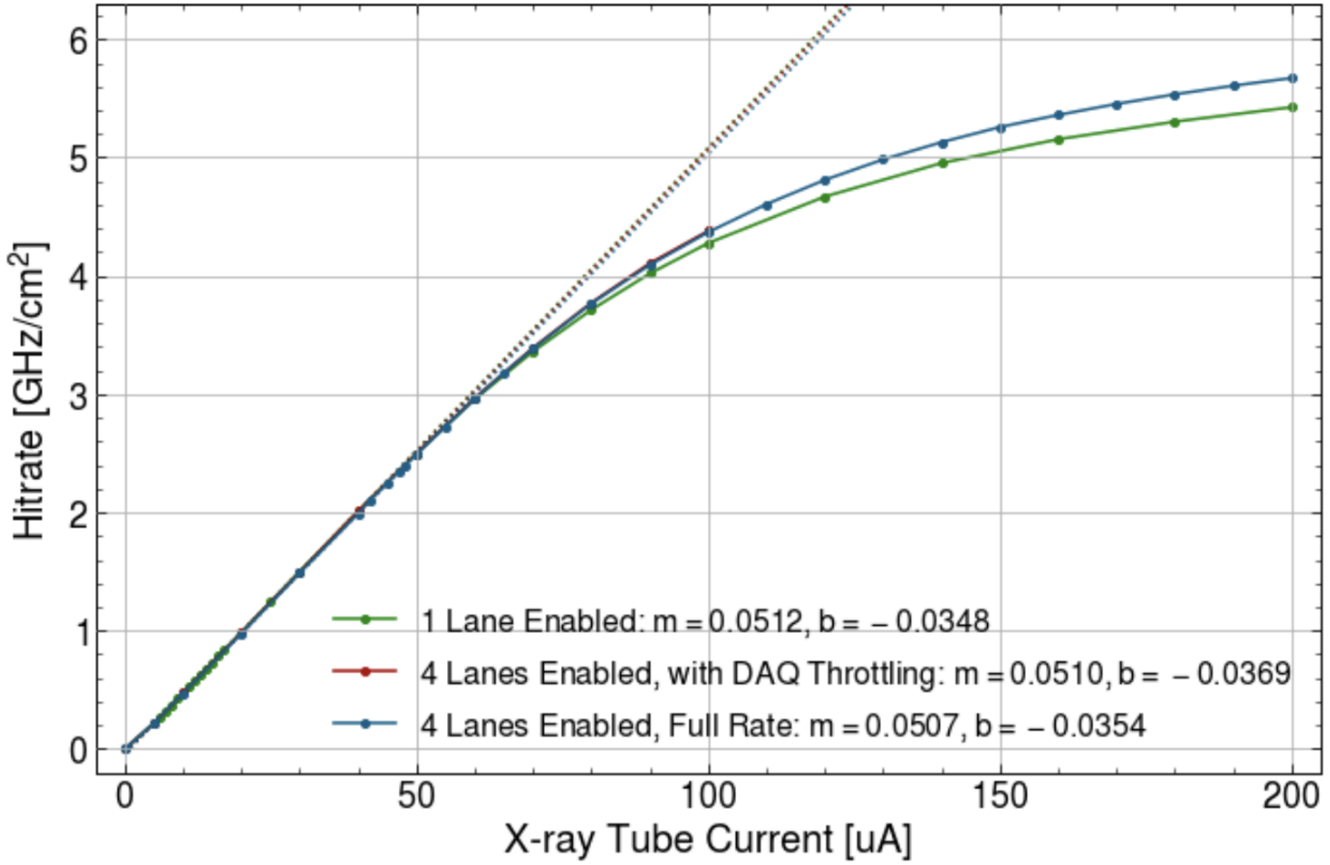


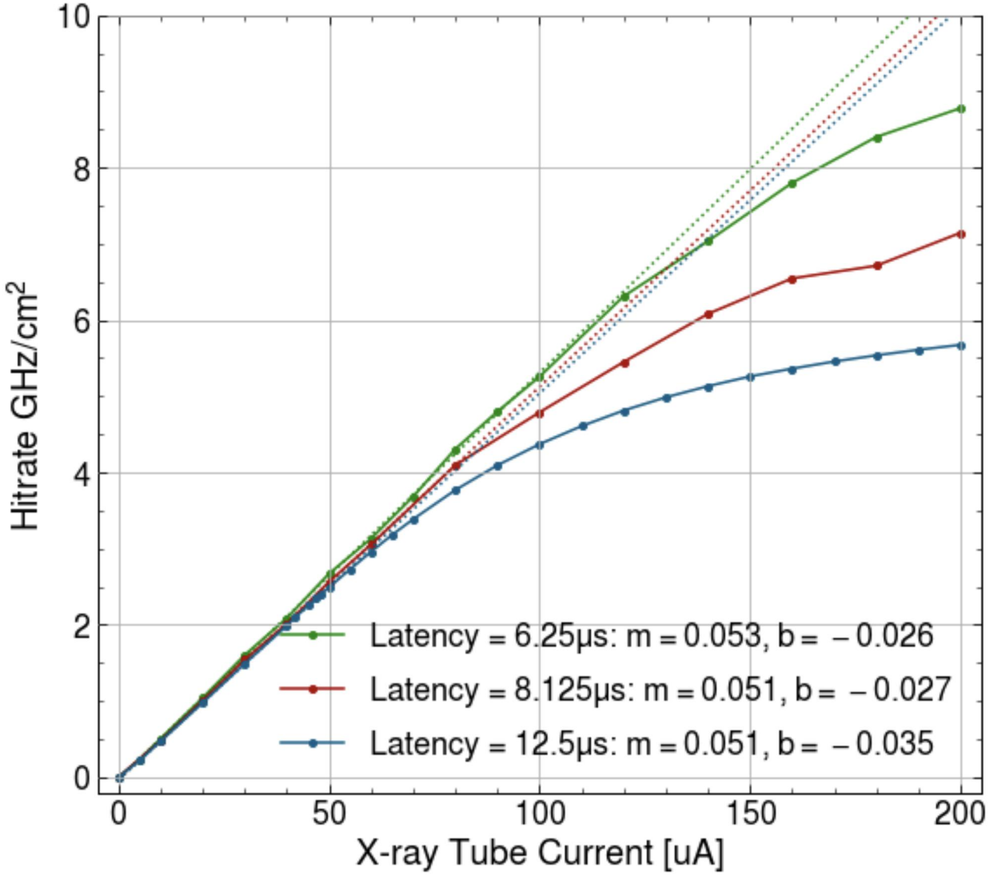


Bitrate (left) and average hit size (right) as a function of hit rate for Krypton source, x-ray source, and simulation values for cluster sizes of 1, 5, and 10 pixels in both Eta and Phi directions.

Hit Rate Correction Factors

Correction factors determined by linears fit to hit rate, from x-ray tube current, are shown below. Identical values for the same latency is expected, within stat. fluctuations





Hit rate as a function of X-ray tube current, with correction factor fits (dotted lines), for a variety of lane configurations (left) and a variety of latencies (right).