

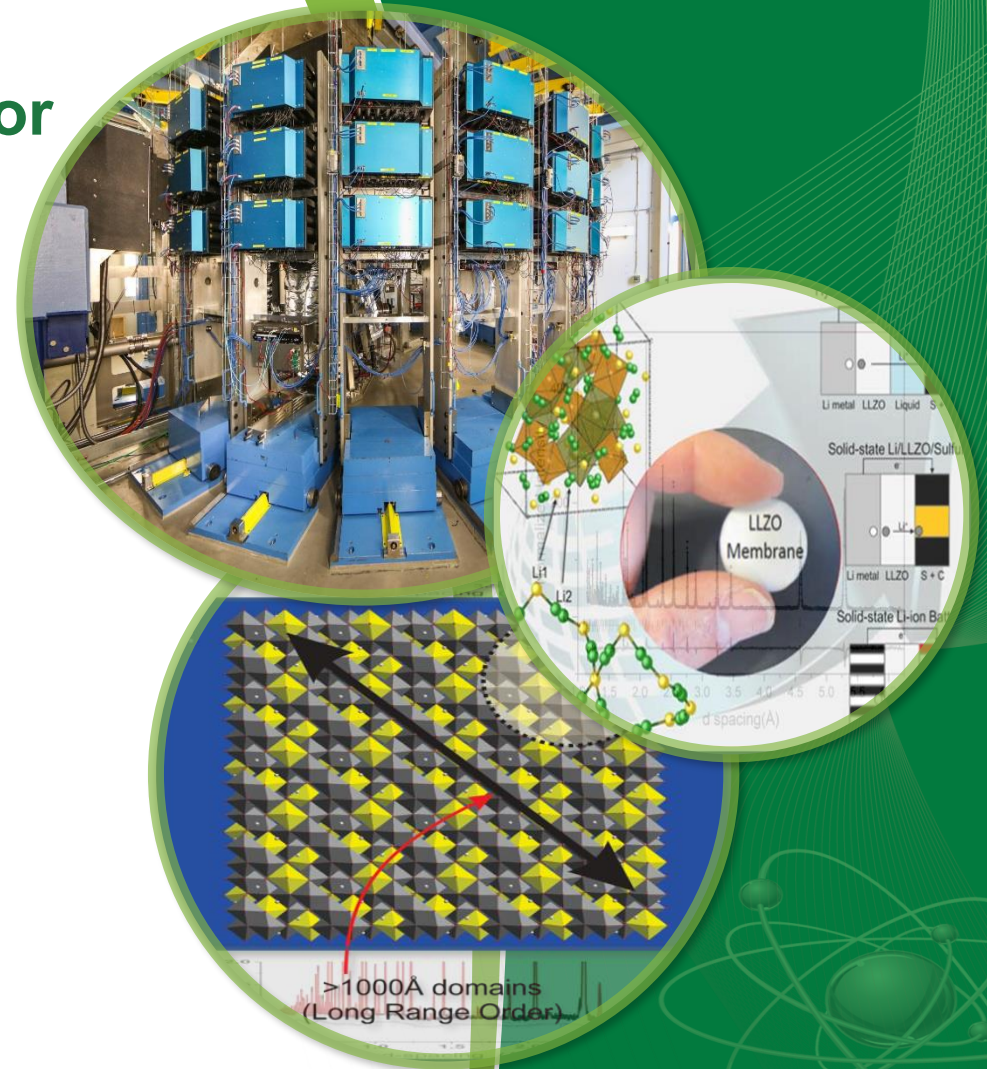
# Deployment of ADTimePix3 areaDetector Driver at Neutron and X-ray Facilities

<https://github.com/areaDetector/ADTimePix3>

**Kazimierz Gofron**

**ORNL**

ICALEPCS 2023, Cape Town, South Africa  
October 9, 2023



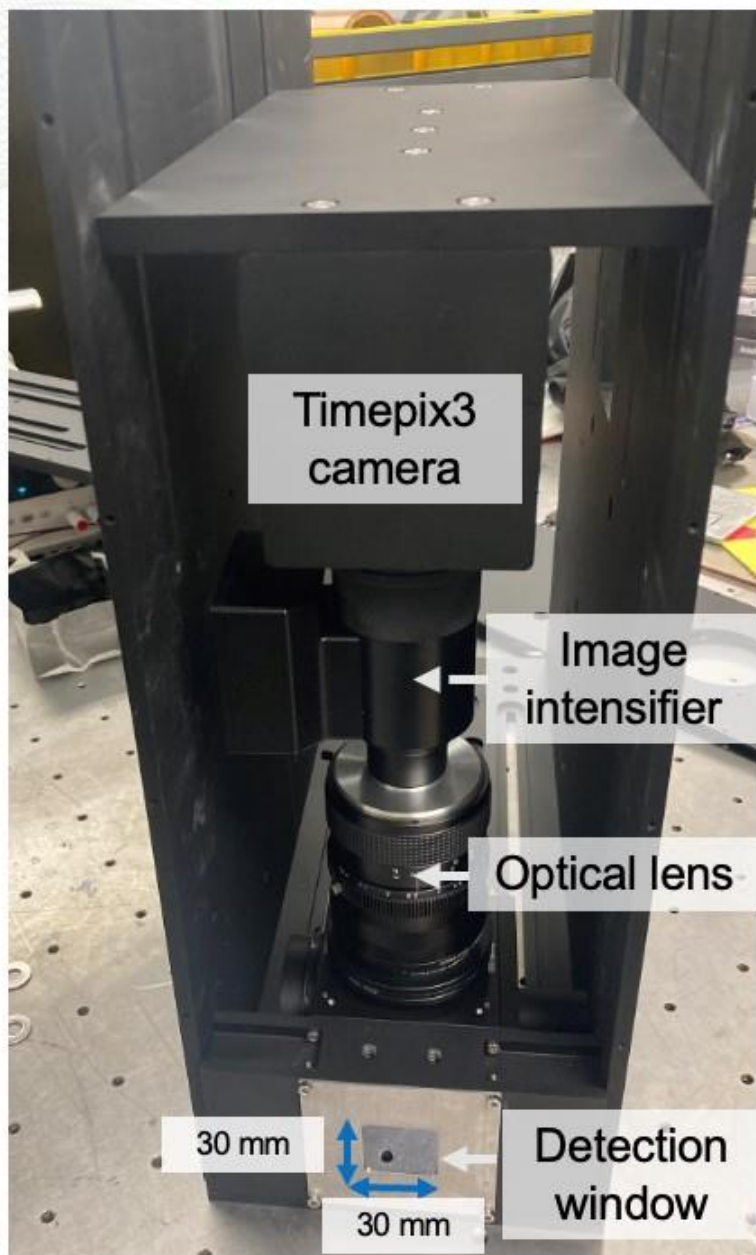
# ASI tpx3Cam for SNS, and HFIR reactor beamlines



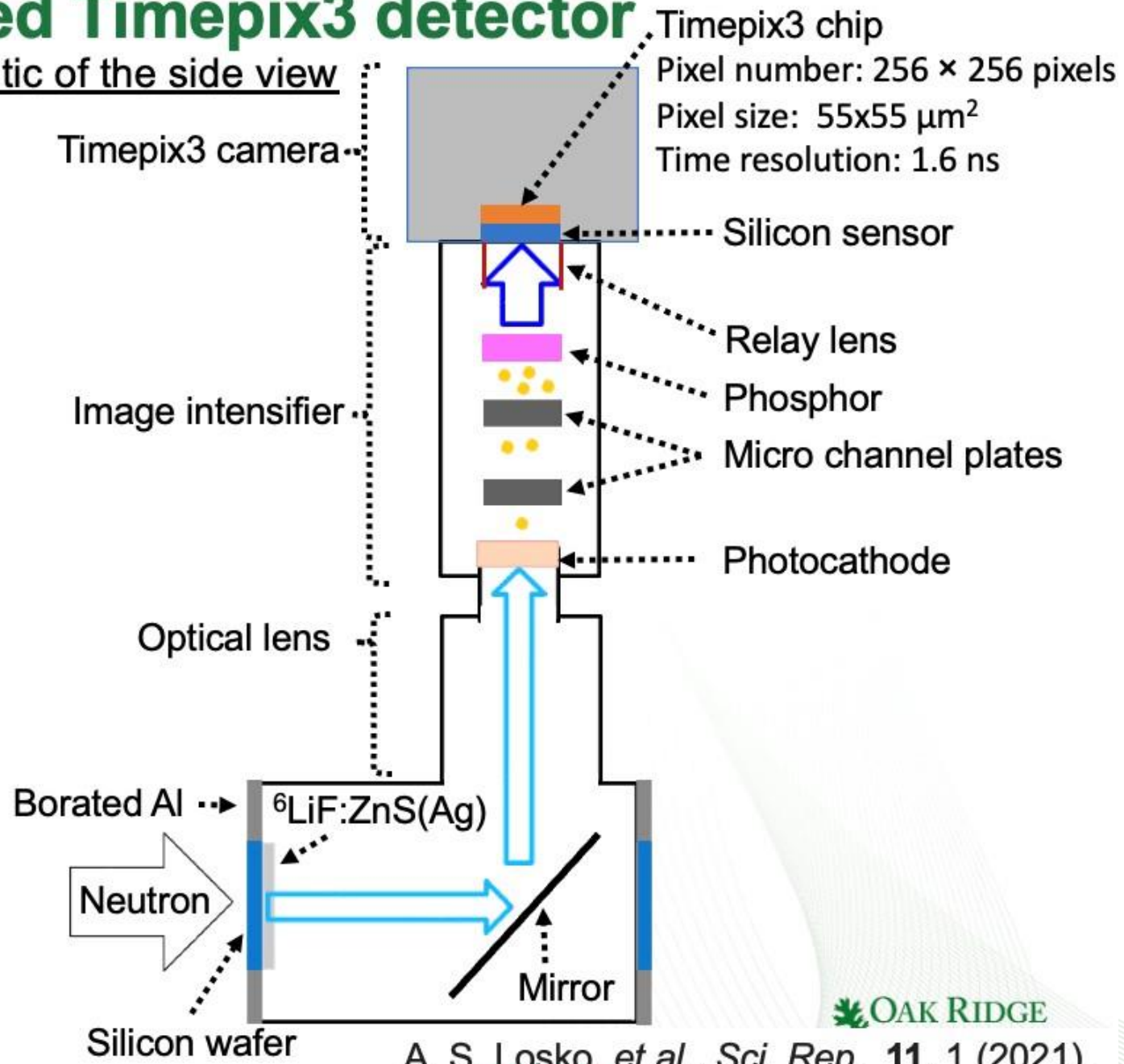
- Thermal neutron detection process
  - ${}^6\text{Li} (n + {}^4\text{He}) {}^3\text{H} + 4.78 \text{ [MeV]}$  (5-9  $\mu\text{m}$ )
  - Scintillator containing  ${}^6\text{Li}$  generates light converted in photocathode and amplified by dual MCP stack (intensifier).
  - Electron cluster detected by Tpx3Cam
  - No vacuum pump – sealed intensifier
  - Single chip tpx3Cam
- Alternative  ${}^{10}\text{B}$  doped glass: 1<sup>st</sup> MCP glass, no scintillator
  - ${}^{10}\text{B} (n + {}^4\text{He}) {}^7\text{Li} + 2.31 \text{ [MeV]}$
  - A.S. Tremsin et al. / Nuclear Instruments and Methods in Physics Research A 592 (2008) 374–384
  - Four chip in-vacuum TimePix3



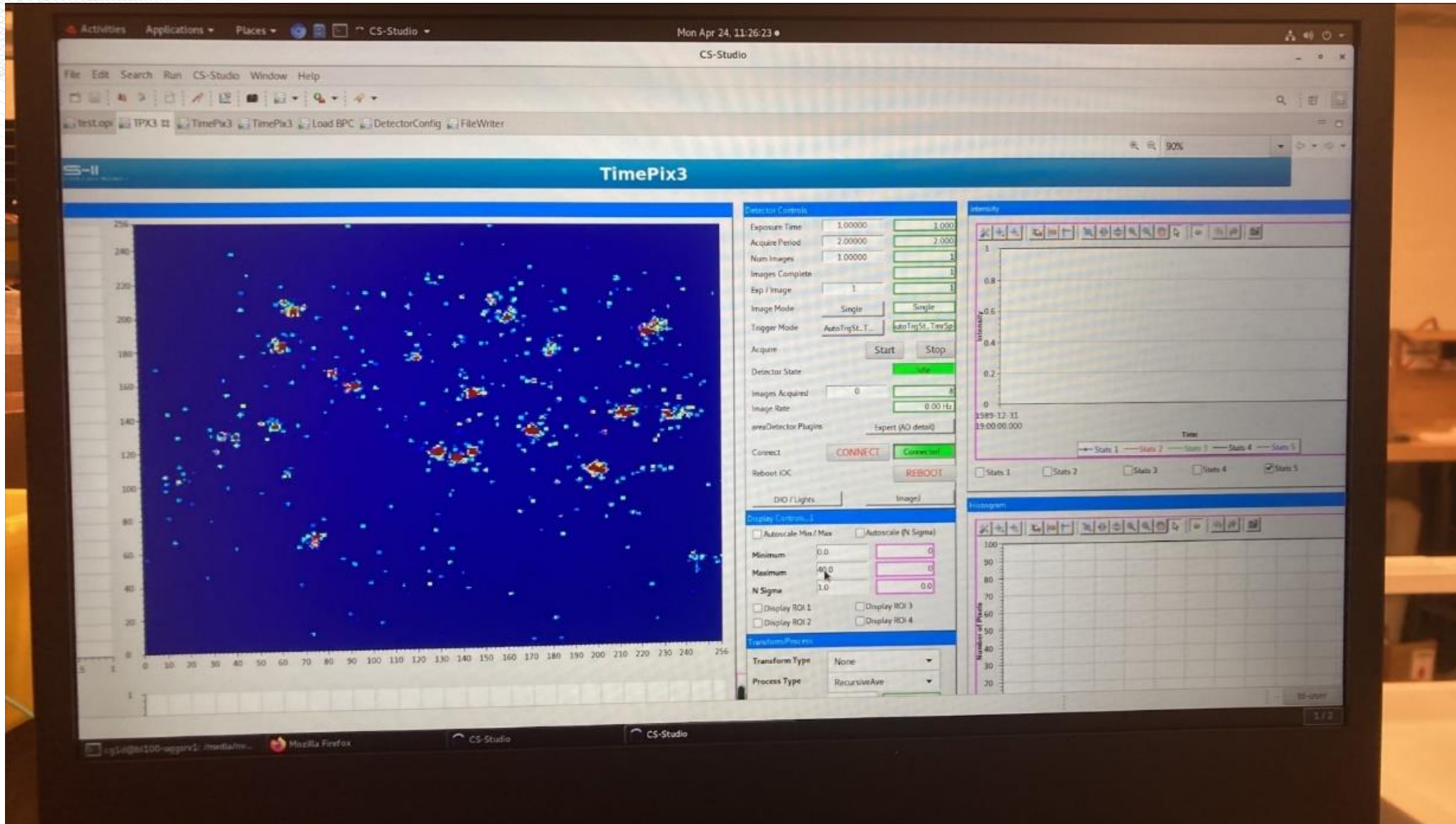
# Developed scintillator-based Timepix3 detector



## Schematic of the side view

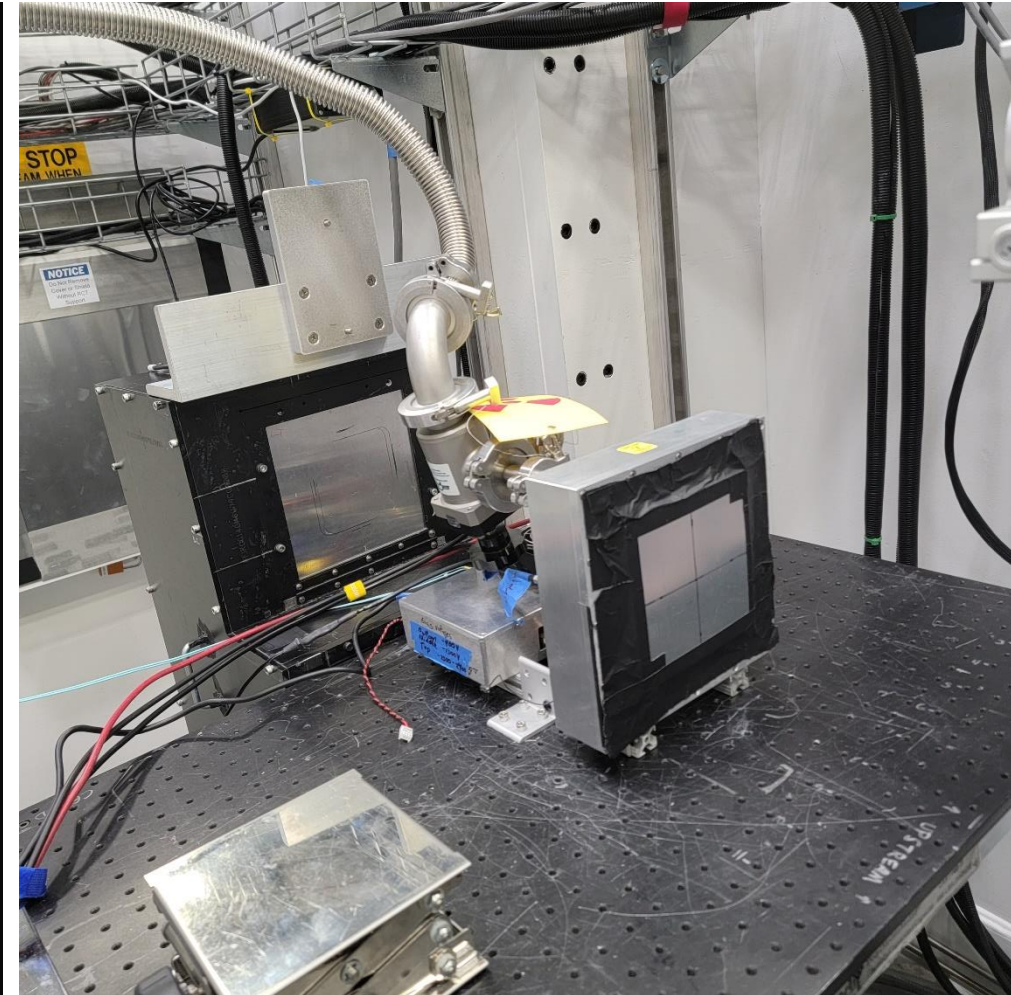
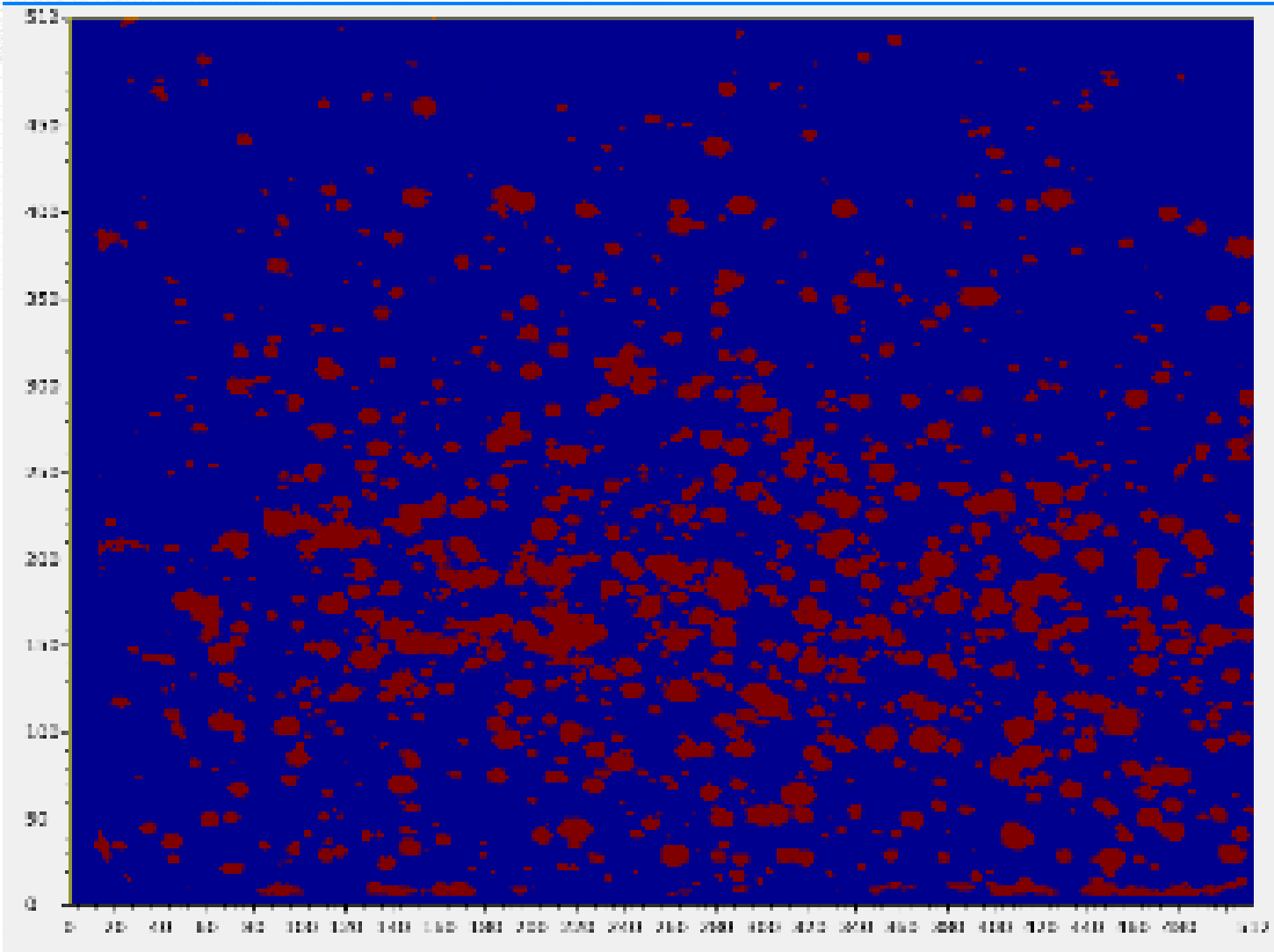


# ASI tpx3Cam thermal neutron 'clusters' – preview



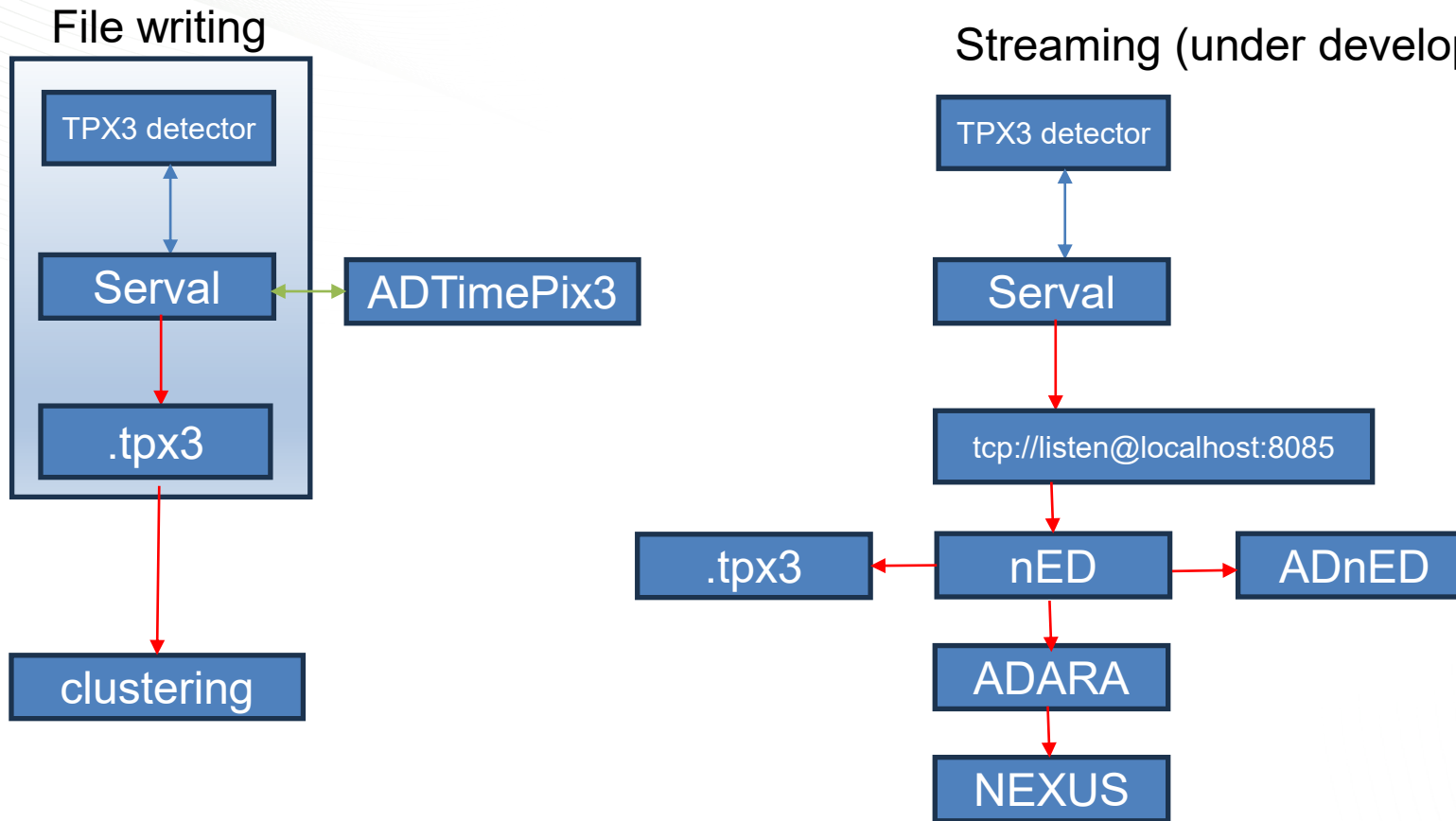


# ASI tpx3Cam thermal neutron 'clusters': $^{10}\text{B}$ detector



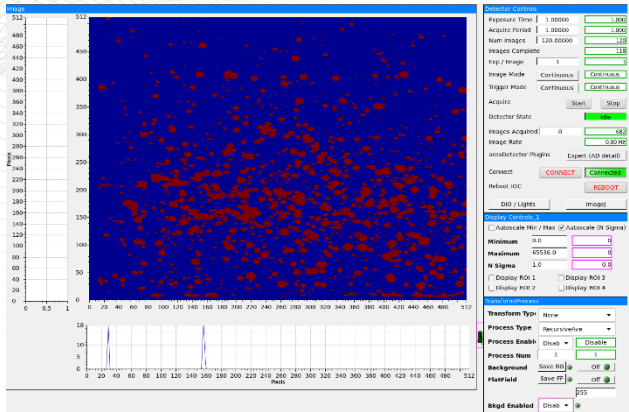
HFIR: CG-1D beamline

# File writing and streaming of hits



- Funama Fumiaki (tpx3Cam optical)
  - Real time data pipeline
  - Clustering
  - nED
  - ADnED
  - ADARA

# ADTimePix3 screens



### Area Detector Control - CG1D:CS:tpx3:cam1:

**Setup**

Shutter mode: None  
 Status: Det. Closed EPICS Closed  
 Open/Close: Open Close  
 Delay: Open 0.000 Close 0.000  
 EPICS shutter setup

**Detector Health**

Local Temperature: 24.278  
 FPGA Temperature: 39.127  
 Fan1 Speed: 0.000  
 Fan2 Speed: 0.000  
 Bias Voltage: 0.000000  
 Humidity: 40.000  
 Chip Temperatures: (44.45, 46.45)  
 VDD: (1.664, 0.553, 0.806)

**Detector info**

ifaceName: Spidr  
 SW\_version: 22092000  
 FW\_version: 22092021  
 PixCount: 282144  
 RowLen: 2  
 NumberOfChips: 4  
 NumberOfRows: 512  
 MpxType: 6  
 Boards->ChipboardId: 41000011  
 Boards->IPAddress: 192.168.100.10  
 Boards->Chip1: "id":14506,"index":0,"Name":"W0"  
 Boards->Chip2: "id":14505,"index":1,"Name":"W0"  
 Boards->Chip3: "id":14504,"index":2,"Name":"W0"  
 Boards->Chip4: "id":14491,"index":3,"Name":"W0"  
 SuppAcqModes: 63  
 ClockReadout: 125.0  
 MaxPulseCount: 2147483647  
 MaxPulseHeight: 1.0  
 MaxPulsePeriod: 34.35973836  
 TimerMaxVal: 34.35973836  
 TimerMinVal: 8E-9  
 TimerStep: 8E-9  
 ClockTempix: 125.0

**Detector Status**

Status: DetConfig  
 File: FileWriter

### Detector config

BiasVoltage: 0  
 BiasEnabled: false  
 ChainMode: NONE  
 TriggerIn: 0  
 TriggerOut: 0  
 Polarity: Negative  
 TriggerMode: Continuous  
 ExposureTime: 1.000  
 TriggerPeriod: 1.000  
 nTriggers: 120  
 DetectorOrientation: Disconnected  
 PeriphClk0: false  
 TriggerDelay: 0.000  
 Tdc0: P0123  
 Toc1: P0123  
 GlobalTimestampInterval: 1.000  
 ExternalReferenceClock: false  
 LogLevel: 0

### CG1D:CS:tpx3:cam1

Detector Health

Detector Config

Detector Chip: CHIP0

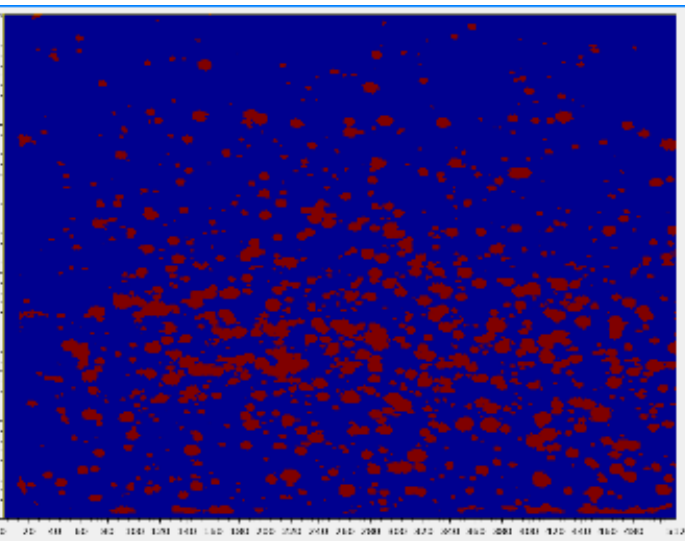
Detector Chip: CHIP1

Detector Chip: CHIP2

Detector Chip: CHIP3

Measurement Info

Dashboard



### CG1D:CS:tpx3:cam1:

BPC File path: /opt/tpx3/snap/ snap\_1.bpc  
 BPC File name: snap\_1.bpc

DACS File path: /opt/tpx3/snap/ snap\_1.bpc.dacs  
 DACS File name: snap\_1.bpc.dacs

Upload BPC file: Upload  
 Upload DACS file: Upload

Write status code: 200  
 Write message: Successfully uploaded DAC settings.

### Area Detector Control - CG1D:CS:tpx3:cam1:

**Detector Health**

Local Temperature: 24.235  
 FPGA Temperature: 39.127  
 Fan1 Speed: 0.000  
 Fan2 Speed: 0.000  
 Bias Voltage: 0.000000  
 Humidity: 40.000  
 Chip Temperatures: (44.45, 46.45)  
 VDD: (1.664, 0.553, 0.806)

**Detector info**

ifaceName: Spidr  
 SW\_version: 22092000  
 FW\_version: 22092021  
 PixCount: 282144  
 RowLen: 2  
 NumberOfChips: 4  
 NumberOfRows: 512  
 MpxType: 6  
 Boards->ChipboardId: 41000011  
 Boards->IPAddress: 192.168.100.10  
 Boards->Chip1: "id":14506,"index":0,"Name":"W0"  
 Boards->Chip2: "id":14505,"index":1,"Name":"W0"  
 Boards->Chip3: "id":14504,"index":2,"Name":"W0"  
 Boards->Chip4: "id":14491,"index":3,"Name":"W0"  
 SuppAcqModes: 63  
 ClockReadout: 125.0  
 MaxPulseCount: 2147483647  
 MaxPulseHeight: 1.0  
 MaxPulsePeriod: 34.35973836  
 TimerMaxVal: 34.35973836  
 TimerMinVal: 8E-9  
 TimerStep: 8E-9  
 ClockTempix: 125.0

**Detector config**

PixelFormat: 16  
 BiasVoltage: 0  
 BiasEnabled: false  
 ChainMode: NONE  
 TriggerIn: 0  
 TriggerOut: 0  
 Polarity: Negative  
 TriggerMode: Continuous  
 ExposureTime: 1.000  
 TriggerPeriod: 1.000  
 nTriggers: 120  
 PeriphClk0: false  
 TriggerDelay: 0.000  
 Tdc: "P0123","P0123"  
 TimestampInterval: 1.000  
 ExternalReferenceClock: false  
 LogLevel: 0

**Detector Layout**

DetectorOrientation: UP  
 Chip0: ("Chip0","Detector","W0B7","X:25")  
 Chip1: ("Chip1","Detector","W0B7","X:25")  
 Chip2: ("Chip2","Detector","W0B7","X:0")  
 Chip3: ("Chip3","Detector","W0B7","X:0")

**Measurement Info**

PixelEventRate: 18827  
 TdcEventRate: 1.8  
 TdcEventRate: 0  
 StartDate: 202506242568  
 ElapsedTime: 121.778  
 TimeLeft: 1.000  
 FrameCount: 119  
 DroppedFrames: 0  
 Status: DA RECORDING

**Detector Chip: CHIP0**

Bias\_CP\_PLL: 128  
 Bias\_DiscS1\_OFF: 8  
 Bias\_DiscS1\_ON: 128  
 s\_DiscS2\_OFF: 8  
 Bias\_DiscS2\_ON: 128  
 Bias\_Enum: 10  
 Bias\_PreProcDAC: 146  
 Bias\_Preamp\_OFF: 8  
 Bias\_Preamp\_ON: 128  
 Bias\_TBufferIn: 128  
 Bias\_TBufferOut: 128  
 PLL\_Vctrl: 128  
 VPreamp\_NCAS: 128  
 VTP\_coarse: 128  
 VTP\_fine: 156  
 VTK: 128  
 VThreshold\_coarse: 7  
 VThreshold\_fine: 278  
 Adjust: -1

**Detector Chip: CHIP1**

Bias\_CP\_PLL: 128  
 Bias\_DiscS1\_OFF: 8  
 Bias\_DiscS1\_ON: 128  
 s\_DiscS2\_OFF: 8  
 Bias\_DiscS2\_ON: 128  
 Bias\_Enum: 10  
 Bias\_PreProcDAC: 141  
 Bias\_Preamp\_OFF: 8  
 Bias\_Preamp\_ON: 128  
 Bias\_TBufferIn: 128  
 Bias\_TBufferOut: 128  
 PLL\_Vctrl: 128  
 VPreamp\_NCAS: 128  
 VTP\_coarse: 128  
 VTP\_fine: 156  
 VTK: 128  
 VThreshold\_coarse: 7  
 VThreshold\_fine: 285  
 Adjust: -1

**Detector Chip: CHIP2**

Bias\_CP\_PLL: 128  
 Bias\_DiscS1\_OFF: 8  
 Bias\_DiscS1\_ON: 128  
 s\_DiscS2\_OFF: 8  
 Bias\_DiscS2\_ON: 128  
 Bias\_Enum: 10  
 Bias\_PreProcDAC: 141  
 Bias\_Preamp\_OFF: 8  
 Bias\_Preamp\_ON: 128  
 Bias\_TBufferIn: 128  
 Bias\_TBufferOut: 128  
 PLL\_Vctrl: 128  
 VPreamp\_NCAS: 128  
 VTP\_coarse: 128  
 VTP\_fine: 156  
 VTK: 128  
 VThreshold\_coarse: 7  
 VThreshold\_fine: 285  
 Adjust: -1

**Detector Chip: CHIP3**

Bias\_CP\_PLL: 128  
 Bias\_DiscS1\_OFF: 8  
 Bias\_DiscS1\_ON: 128  
 s\_DiscS2\_OFF: 8  
 Bias\_DiscS2\_ON: 128  
 Bias\_Enum: 10  
 Bias\_PreProcDAC: 141  
 Bias\_Preamp\_OFF: 8  
 Bias\_Preamp\_ON: 128  
 Bias\_TBufferIn: 128  
 Bias\_TBufferOut: 128  
 PLL\_Vctrl: 128  
 VPreamp\_NCAS: 128  
 VTP\_coarse: 128  
 VTP\_fine: 156  
 VTK: 128  
 VThreshold\_coarse: 7  
 VThreshold\_fine: 285  
 Adjust: -1

**Dashboard**

RecSpace: 0.000E+00  
 WriteSpeed: 8.000E+00  
 LowerLimit: 8.000E+00  
 DiskLimitReached: false



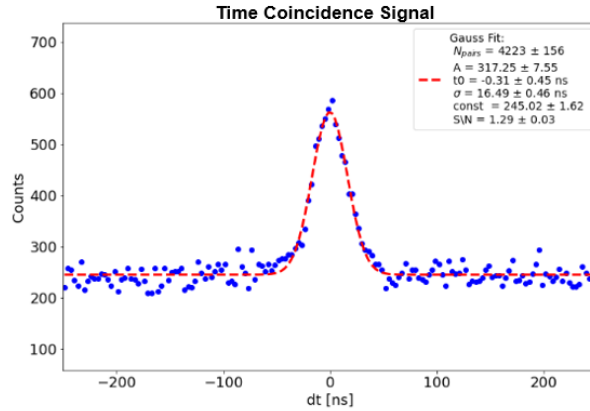
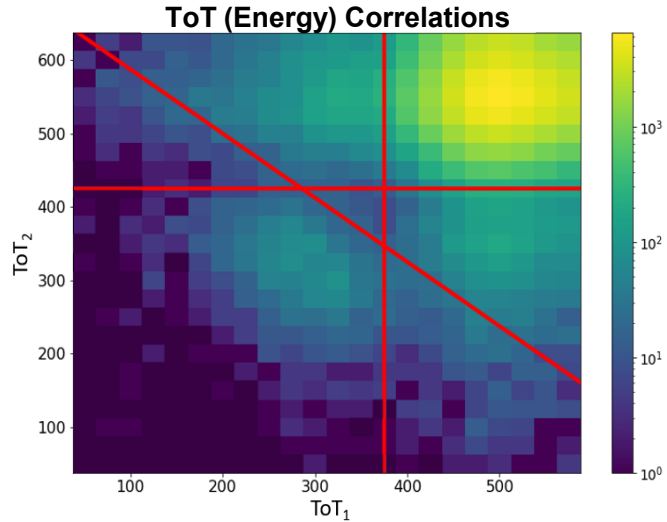
# ASI TimePix3 four chip: direct detection of X-ray

- An EPICS areaDetector driver for TimePix3 quad 512 x 512 detector from ASI.
- ASC <https://www.amscins.com/>
- NSLS2/CHX – Coherent Hard X-ray Scattering beamline.
- TDC – Timing from accelerator (260 ps); DIO: 2 x 3 (timing signals)

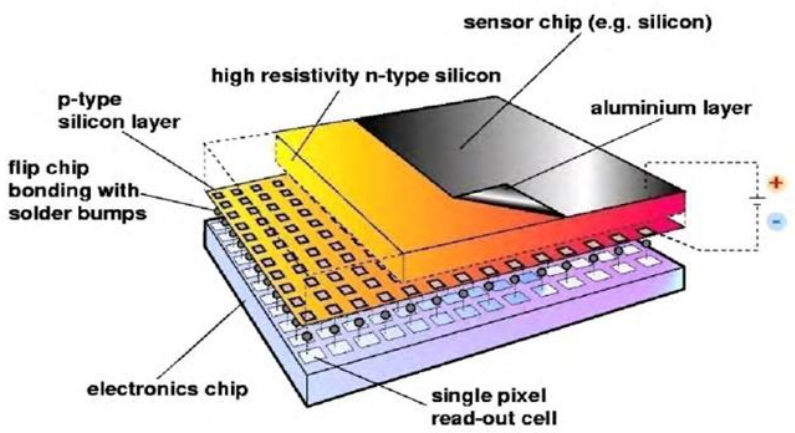




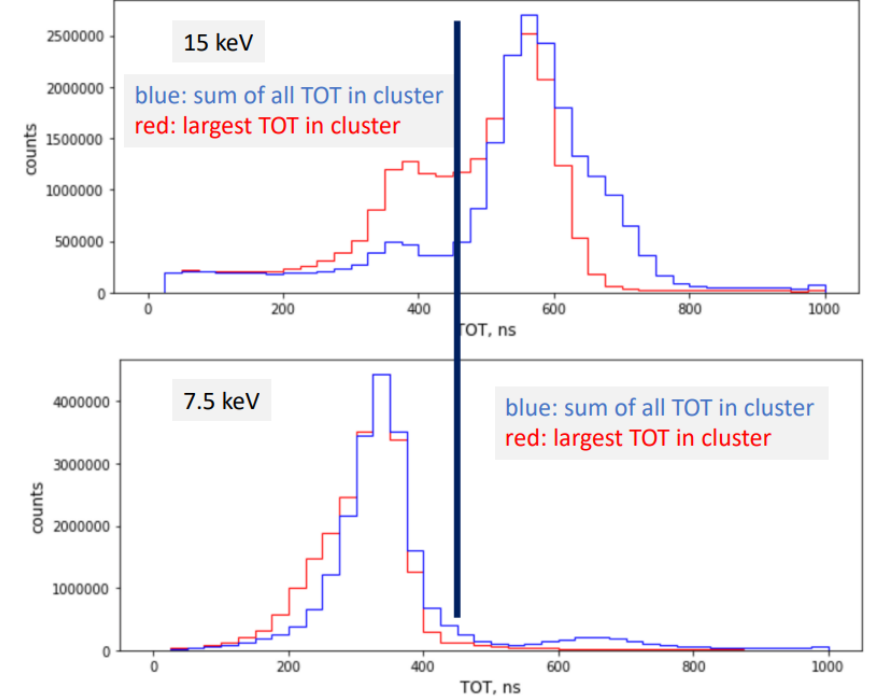
# ASI TimePix3 four chip – direct detection



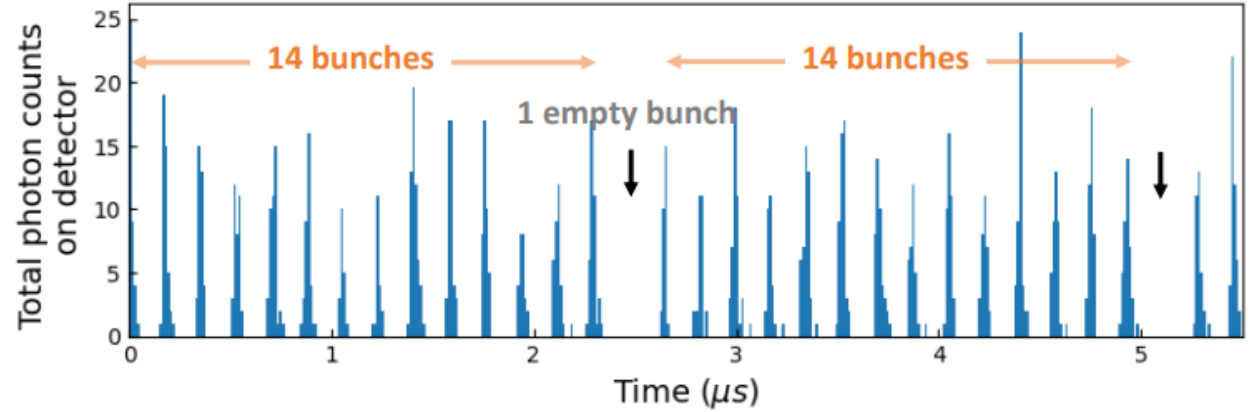
<b>Readout</b>	x, y, ToT, ToA
<b>ToA precision</b>	1.56 ns
<b>Pixel size</b>	55 $\mu\text{m}$ x 55 $\mu\text{m}$
<b>Pixel matrix</b>	512 x 512 (4 chips)
<b>Data rate</b>	40 Mhits/s/cm <sup>2</sup>
<b>Energy range</b>	2 keV – 60 keV



## ToT Distributions (Energy Filtering)



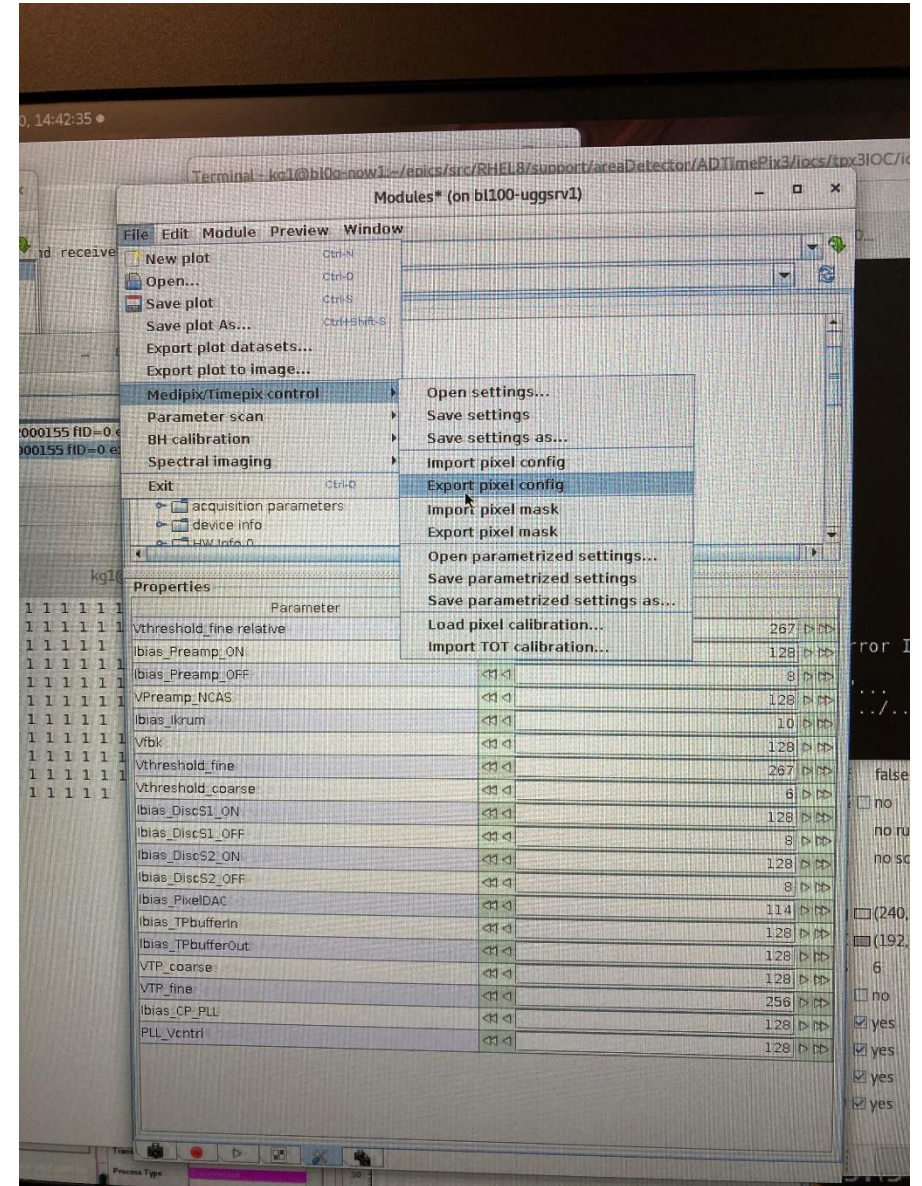
## Ultrafast ToA (Distinguishing Synchrotron Bunches)



Courtesy of Justin Goodrich, Andrei Fluerașu, et. al. NSLS2

# ASI TimePix3 Calibration

- SoPhy vendor software
  - DACS calibration file
  - Binary Pixel Configuration file
- ASI provides calibrations with detector





# ADTimePix3 EPICS areaDetector driver

- The ADTimePix3 areaDetector driver was developed using an emulator and SERVAL software provided by Amsterdam Scientific Instruments (ASI) [9]. The Java emulator software to substitute for the real detector.
- Selected cpr [11] and json [12] software libraries as candidates for handling our representational state transfer (REST) [13] application programming interface (API) communication—commands and responses are JSON [6] format.
- Used Python cookiecutter [14] template for an EPICS areaDetector [7] driver, ADDriverTemplate [15], we created the skeleton of a driver.
- Determined the linker commands to link those dependencies into our driver
- An executable was generated, and the external dependencies were linking correctly
- Wrote the input/output controller (IOC) shell function that initializes the driver to collect some basic diagnostic information upon start-up. Once communication with SERVAL from the EPICS IOC was established, additional readback functionality and control commands were incorporated
- A REST API made it easy to break down the work into individual POST/GET requests. Each request was added to the driver, with various EPICS records tied to each field in the command and response.
- The vendor software writes binary .tpx3 data files directly to disk at a preconfigured location and only posts occasional preview images to a network location accessible by SERVAL via a GET request.

# ADTimePix3 EPICS areaDetector driver (cont)

- We used a setup used by the “ADURL” areaDetector [16] driver with GraphicsMagick [17] library to read an image file from a network location into areaDetector. First, we request the preview image using the cpr library and then used the same approach to decode the image. This decoded frame is then passed along to areaDetector plugins via a callback
- The cpr GET requests use the libcurl library but do not close the connection, thereby activating many TIME\_WAIT connections. Therefore, a Session feature of the cpr library was used to maintain a single connection and transport preview images in the callback method.



# SERVAL tuning

- The SERVAL software requires specific IP addresses for the detector readout electronics (192.168.100.10/24) and the host computer (192.168.100.1/24).
- The communication between the readout electronics and the host uses incoming UDP ports 8192, 8193, 8194, and 8195 for detector data
- The host firewall must allow outgoing and incoming TCP traffic on port 50000. SERVAL uses this port to connect to the detector.
- The localhost http port 8081 (*ORNL specific*), which ADTimePix3 communicates to SERVAL web interface.
- Jumbo frames must be enabled, and the maximum transfer unit (MTU) must be set to 9000 on the detector's private 192.168.100.1/24 ethernet interface.
- The receive window memory size was adjusted (`net.core.rmem_max = 26214400`, and `net.core.rmem_default = 26214400`).
- The detector network interface was tuned with ASI-provided `nictune.sh` script (*or using the linux `ethtool` commands*) to minimize detector packet reordering.
- Since version 3.2, SERVAL reports UDP packet loss. SERVAL settings such as a `resourcePoolSize` of about 1,048,576 or 2,097,152 minimize UDP packet loss
- The ADTimePix3 detector control was operated on Ubuntu 18.04, 20.04, and 22.04, and on RHEL 7.9 and 9.0.

# ADTimePix3 dependencies

- Json C++ cpr, and json library use

```
std::string config;
config = this->serverURL + std::string("/detector/config");

// Detector configuration file
r = cpr::Get(cpr::Url{config},
            cpr::Authentication{"user", "pass", cpr::AuthMode::BASIC},
            cpr::Parameters{{"anon", "true"}, {"key", "value"}});

json config_j = json::parse(r.text.c_str());
config_j["BiasVoltage"] = 103;
config_j["BiasEnabled"] = true;

//config_j["Destination"]["Raw"][0]["Base"] = "file:///home/kgofron/Downloads";
//printf("Text JSON server: %s\n", config_j.dump(3, ' ', true).c_str());

r = cpr::Put(cpr::Url{config},
            cpr::Body{config_j.dump().c_str()},
            cpr::Header{{"Content-Type", "text/plain"}});

printf("Status code: %li\n", r.status_code);
printf("Text: %s\n", r.text.c_str());
```

```
"/Config" : {
  "Fan1PWM" : 0,
  "Fan2PWM" : 0,
  "BiasVoltage" : 0,
  "BiasEnabled" : false,
  "ChainMode" : "NONE",
  "TriggerIn" : 0,
  "TriggerOut" : 0,
  "Polarity" : "Positive",
  "TriggerMode" : "PEXSTART_NEXSTOP",
  "ExposureTime" : 0.1,
  "TriggerPeriod" : 0.5,
  "nTriggers" : 0,
  "PeriphClk80" : false,
  "TriggerDelay" : 0.0,
  "Tdc" : [ "PN0123", "PN0123" ],
  "GlobalTimestampInterval" : 0.0,
  "ExternalReferenceClock" : false,
  "LogLevel" : 1
},
"Layout" : {
  "DetectorOrientation" : "UP",
  "Original" : {
    "Chips" : [ {
      "Chip" : 0,
      "X" : 256,
      "Y" : 0,
      "Orientation" : "LtRBtT"
    } ], {
```



# ADTimePix3 opi – disk data structure

- Detector configuration, and controls.
- Currently data written to NVME internal disk for performance
- NVME
  - Raw
  - Img
  - Prv
    - Img
    - Hst {Serval experimental version support}
- Real time data using socket – SNS/HFIR primary
- Data write rates (~ 1GB/s)
- EPICS preview channel used for live viewing
- .tpx3 data format
-

# Summary

- An EPICS ADTimePix3 areaDetector driver for ASI TimePix3 was developed.
  - Uses cpr (Curl for human's library)
  - Uses json library
- X-ray, and charged particles use direct detection
- Thermal neutrons use indirect detection through fission of  $^6\text{Li}$  in scintillator, or  $^{10}\text{B}$  fission reaction
- Primary method of storing data is through .tpx3 file, and/or live stream.



# Acknowledgements

- Jakub Wlodek, NSLS2, Brookhaven National Lab
- CHX/NSLS2 team
  - Andrei Fluerasu, Xiaoqian Chen, Justin C. Goodrich, NSLS2, Brookhaven National Lab
  - Andrei Nomerotski, Physics Department, Brookhaven National Lab
- ORNL
  - Vacaliuc, Bogdan, Greg Guyotte, Klemen Vodopivec, Matt Waddel, James Kohl, Zach Thurman, Alex Sobhani
  - Starra Lyons, Seth Giles
  - Funama Fumiaki, Chong Su-Ann, Zhang, Chen