

# Micron FPGA test on protoDUNE-SP

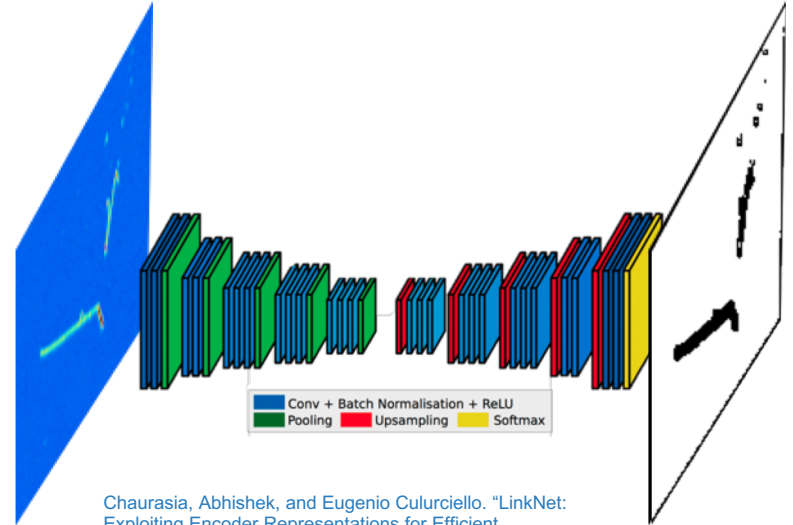
Manuel J. Rodriguez, Saul Alonso



# OUR PLANS

## *Data selection and trigger generation*

- Focus on identifying areas of interest where there is activity on the detector.
- Fully Convolutional Networks to do image segmentation (**UNets**).
- **Input:** raw signals.
- **Goal:** checking the raw signals to get information from the waveforms.
  - **Locate where there are hits!**

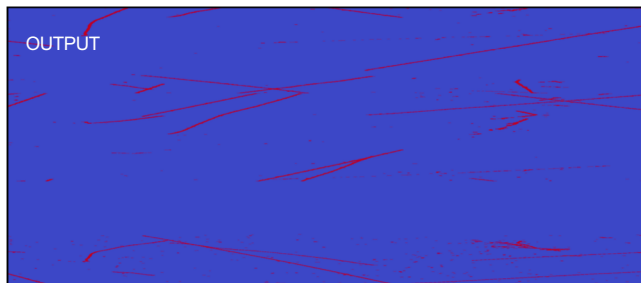
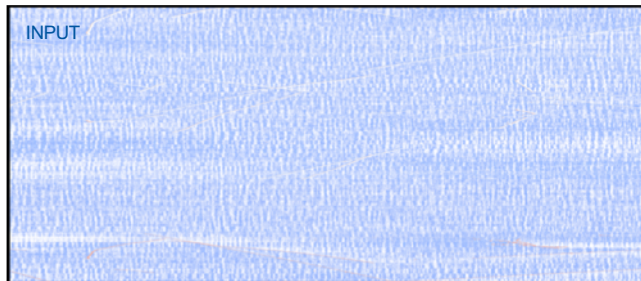


Chaurasia, Abhishek, and Eugenio Culurciello. "LinkNet: Exploiting Encoder Representations for Efficient Semantic Segmentation." 2017 IEEE Visual Communications and Image Processing (VCIP) (2017): n. pag. Crossref. Web. [arXiv:1707.03718](https://arxiv.org/abs/1707.03718)

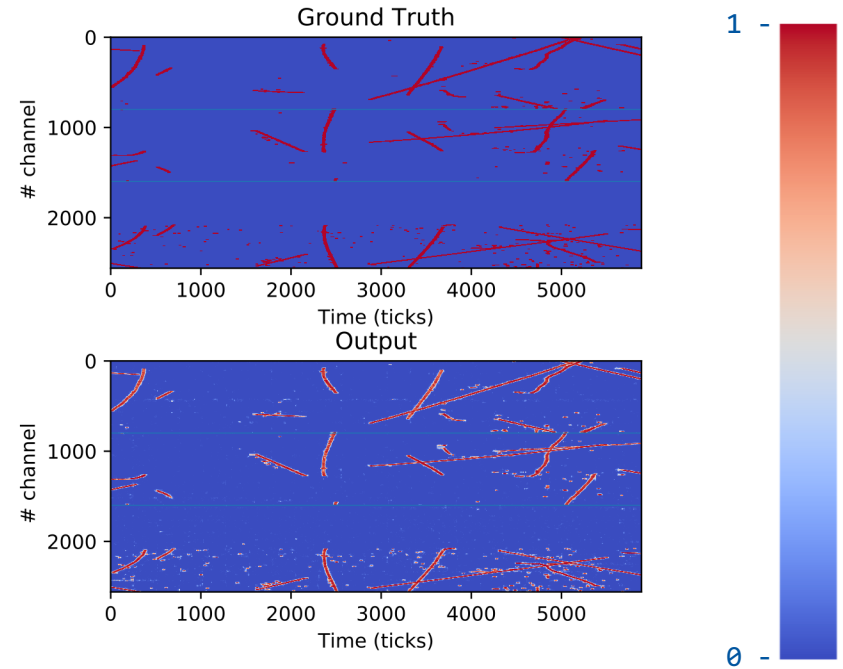
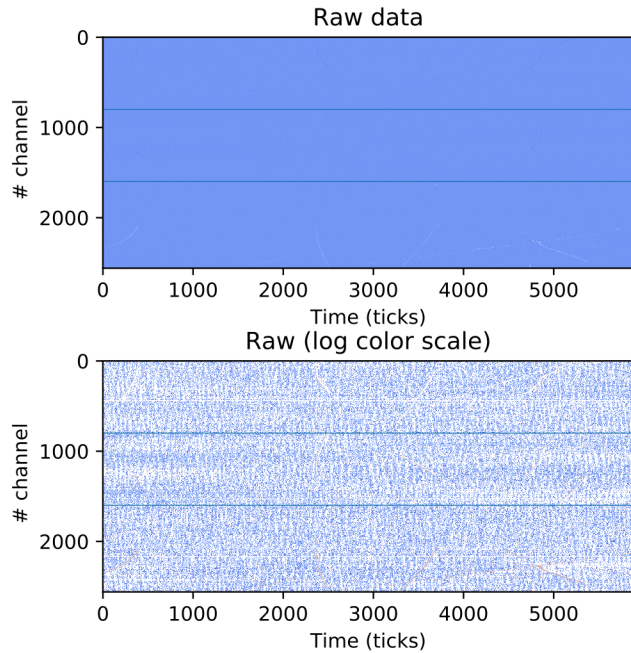
# OUR PLANS

## *Data selection and trigger generation*

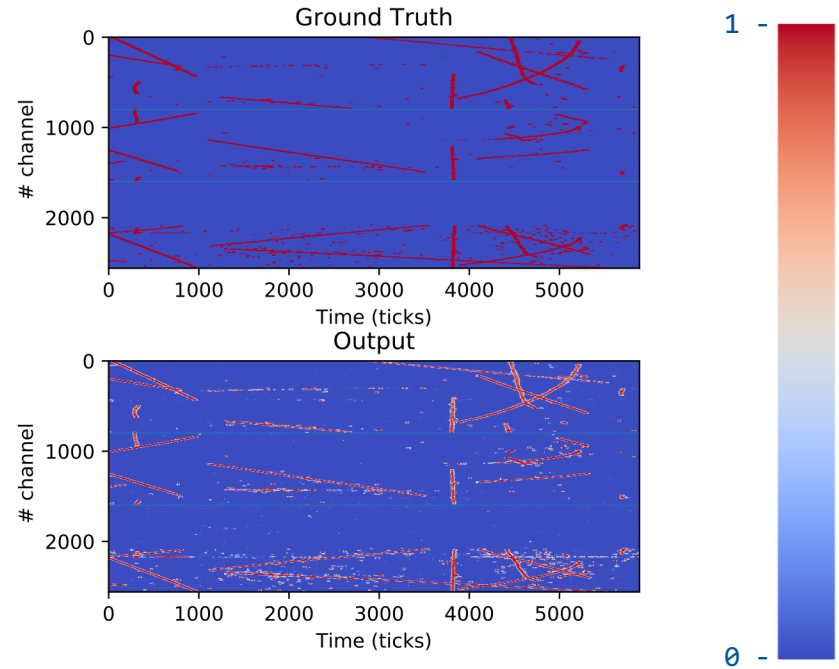
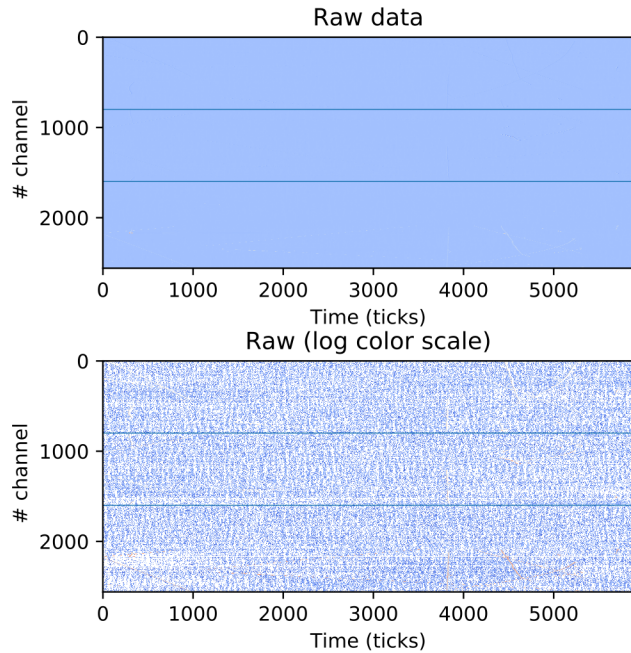
- Focus on identifying areas of interest where there is activity on the detector.
- Fully Convolutional Networks to do image segmentation (**UNets**).
- **Input:** raw signals.
- **Goal:** checking the raw signals to get information from the waveforms.
  - **Locate where there are hits!**



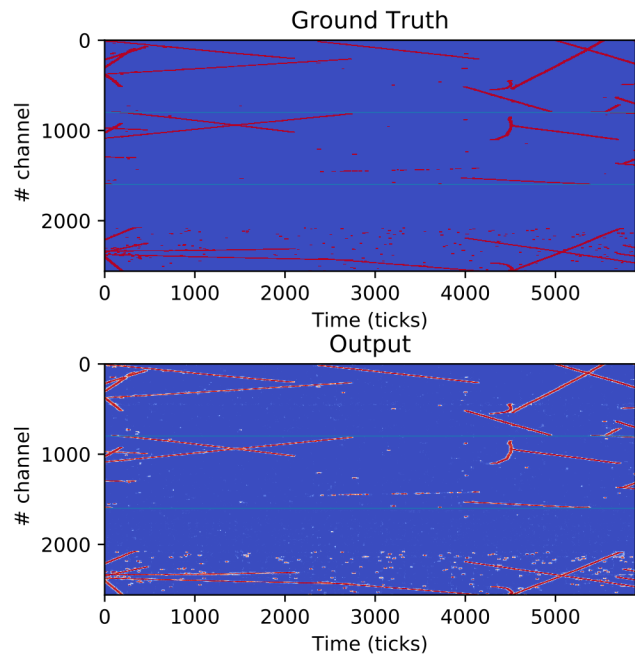
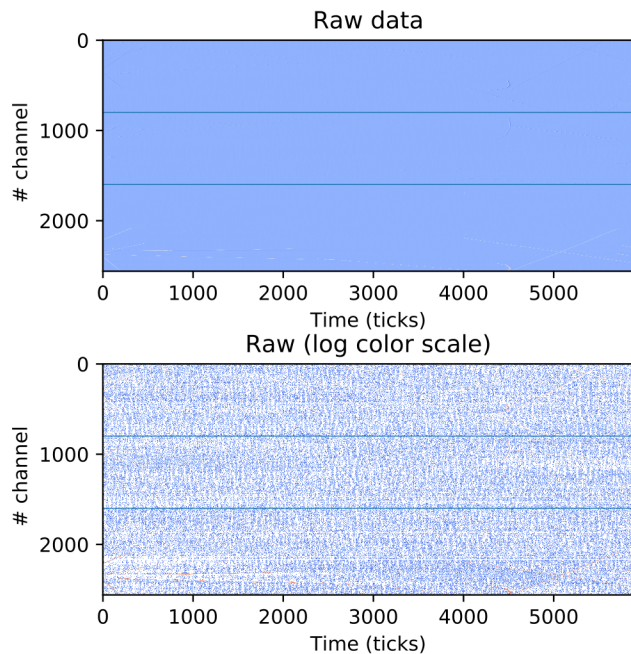
# OFFLINE RESULTS



# OFFLINE RESULTS



# OFFLINE RESULTS



# ONLINE SOLUTIONS

- We aim to find these Regions of Interest (RoI) on real time.
- To analyze a whole trigger window of 3 ms we need to run the inference over 15'360'000 pixels (2560 channels times 6000ms clock ticks)
- After some research and reducing the network to its minimum, this cannot be done with all the incoming data. We need triggered data.
- We our goal is to run it at 12.5 Hz ,meaning that we have 80 ms to run the inference, per trigger window.

# MICRON DLA

*Direct deployment of neural networks on the inference engine*

Micron Deep Learning Accelerator<sup>[1]</sup>:

- No HDL programming.
- Natively supported neural networks.
- Most of the common layers are supported.
- Any framework that supports export to ONNX.
- Inference engine as an accelerator.



*“Machine learning powers your world”*

<sup>[1]</sup><https://fwdnxt.com/>



# INFERENCE ENGINE

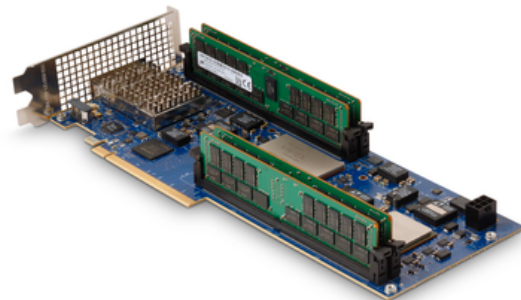
*An FPGA ready for machine learning!*

Micron Advanced Computing Solutions (ACS)

SB-852<sup>[1]</sup>:

- Xilinx Virtex Ultrascale+ UV9P.
  - 64GB DDR4 SODIMM (up to 512GB).
  - 2GB Hybrid Memory Cube.
  - 2 QSFP transceiver connectors.
  - PCIe x16 Gen3 to the host.
- 
- With the 2 Clusters version, the inference will take 700ms.

The Micron logo consists of the word "Micron" in a bold, blue, sans-serif font, with a stylized blue swoosh above the letter "i".



**X**

<sup>[1]</sup><https://www.micron.com/products/advanced-solutions/advanced-computing-solutions/hpc-single-board-accelerators/sb-852>

# INFERENCE ENGINE

*An FPGA ready for machine learning!*

Micron Advanced Computing Solutions (ACS)

AC-511 (x3)<sup>[1]</sup>:

- Xilinx Virtex Ultrascale+ UV7P.
  - 16GB DDR4 SODIMM
  - 2GB Hybrid Memory Cube.
  - PCIe x8 Gen3 to the host.
  - SDAccel (OpenCL) support
- 
- With the 4 Clusters version,
  - the inference will take 100ms.

➔ Almost nominal!



<sup>[1]</sup> <https://www.micron.com/products/advanced-solutions/advanced-computing-solutions/ac-series-hpc-modules/ac-511>

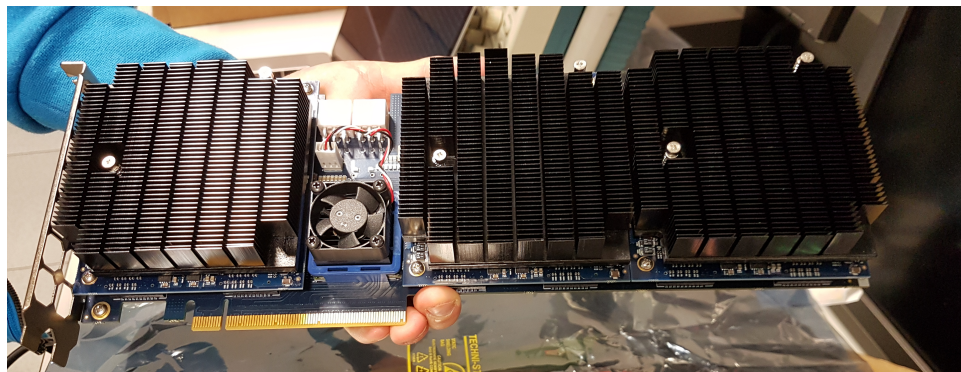
# INFERENCE ENGINE

*An FPGA ready for machine learning!*

## Micron Advanced Computing Solutions (ACS)

### AC-511 (x3)<sup>[1]</sup>:

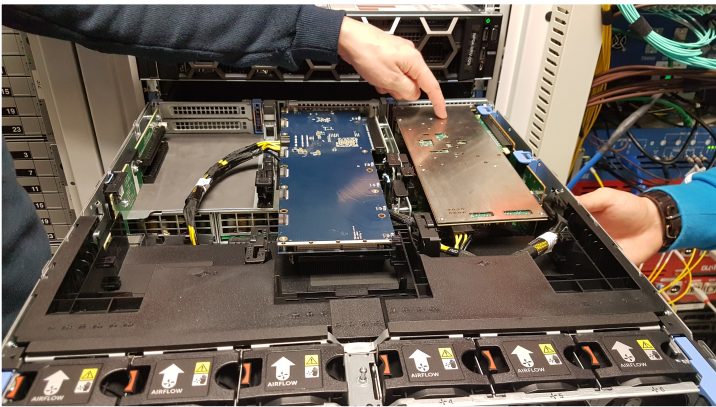
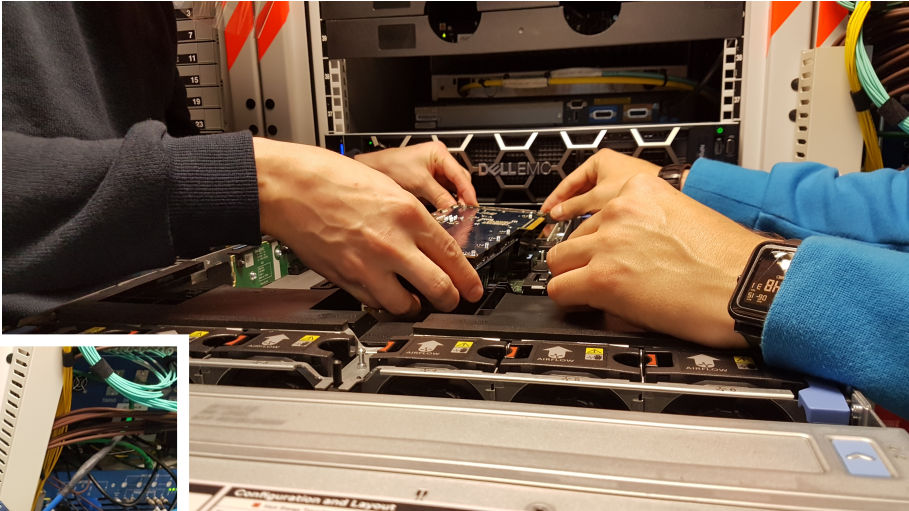
- Xilinx Virtex Ultrascale+ UV7P.
  - 16GB DDR4 SODIMM
  - 2GB Hybrid Memory Cube.
  - PCIe x8 Gen3 to the host.
  - SDAccel (OpenCL) support
- 
- With the 4 Clusters version,
  - the inference will take 100ms.



➔ Almost nominal!

<sup>[1]</sup> <https://www.micron.com/products/advanced-solutions/advanced-computing-solutions/ac-series-hpc-modules/ac-511>

# PROTODUNE-SP TEST



# PROTODUNE-SP TEST

- We installed the driver for the Micron board and...
- We lost np04-srv-028

```
GNU GRUB version 1.99,5.11.0.175.1.0.0.13.18988
Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists possible
device or file completions. ESC at any time exits.
grub> ls
(hd0) (hd0,gpt9) (hd0,gpt2) (hd0,gpt1) (fd0)
grub> _
```

# PROTODUNE-SP TEST

- We managed to fix the Grub, but Dracut wasn't happy either...
- The only solution:  
-> To call to our great System Administrators

```
[ 382.827287] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 383.343763] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 383.861111] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 384.377864] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 384.894978] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 385.415927] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 385.932972] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 386.453971] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 386.978967] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 387.495628] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 388.015859] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 388.536645] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 389.058236] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 389.575878] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 318.892188] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 318.688021] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 311.137785] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 311.666541] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 312.193794] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 312.722578] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 313.258481] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 313.778515] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 314.299388] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 314.824583] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 315.341764] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 315.858386] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 316.375358] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 316.891884] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 317.409889] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 317.925782] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 318.442779] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 318.959574] dracut-initqueue[795]: Warning: dracut-initqueue timeout - starting timeout scripts
[ 318.959647] dracut-initqueue[795]: Warning: Could not boot.
Starting Setup Virtual Console...
OK | Started Setup Virtual Console.
Starting Dracut Emergency Shell...
Warning: /dev/disk/by-id/md-uuid-52a163a3-92788b33-b577e39f-a61f14dc does not exist
Generating "/run/inittarafs/rdsosreport.txt"
Entering emergency mode. Exit the shell to continue.
Type "journalctl" to view system logs.
You might want to save "/run/inittarafs/rdsosreport.txt" to a USB stick or /boot
after mounting them and attach it to a bug report.
dracut:~# exit
```

# PROTODUNE-SP TEST

- The diagnosis was that the driver module (which is compiled on the host to ensure compatibility) was corrupted. Therefore the system failed to load the module and all its dependencies.
- We reinstalled it and it worked.

# TESTING THE BOARD ON srv-028

- We tested it over and over and over again.
- However, every time we tried to run the FPGA it was throwing a “bad fpga seq”
- At this point Micron joined the test.
- We tried together to debug it without any success. Even with a simple demo firmware on the FPGA it was failing.
- They thought that it could be a hardware failure.

```
[mjrodrig@np04-srv-028 ProtoDUNE-scripts]$ ./threadedbatchdemo -i test/ -s
tinylinknet_20200528.bin -r 1024x2560x1 -f 3 -C 4 -B
=====
ie_init: Initialize Micron DLA system
DLA binary to be read is tinylinknet_20200528.bin
-----
Using FPGA 0x511 Device 0511
^C
```

```
[Jun16 16:31] pico: bad fpga seq for fpga 1 stream 254! expected 0x190,
got 0x180. last_host_seq: 0x180 (desc seq: 0x190)
[ +0.000003] pico:pico_newfw(): pico_newfw_internal() return error: -
10011
[ +0.000002] pico: couldn't send 'read' command to system PicoBus: -10011
[ +9.303503] pico: interrupted while waiting for dma
```



# TESTING THE BOARD ON srv-028

- We tried replacing one of the three FPGA, that seems faulty, but we still were having the same issue.
- Micron is still investigating this issue.
- Solution: Try the old SB-852



Not ideal at all.

```
[mjrodrig@np04-srv-028 ProtoDUNE-scripts]$ ./threadedbatchdemo -i test/ -s
tinylinknet_20200528.bin -r 1024x2560x1 -f 3 -C 4 -B
=====
ie_init: Initialize Micron DLA system
DLA binary to be read is tinylinknet_20200528.bin
-----
Using FPGA 0x511 Device 0511
^C
```

```
[Jun16 16:31] pico: bad fpga seq for fpga 1 stream 254! expected 0x190,
got 0x180. last_host_seq: 0x180 (desc seq: 0x190)
[ +0.000003] pico:pico_newfw(): pico_newfw_internal() return error: -
10011
[ +0.000002] pico: couldn't send 'read' command to system PicoBus: -10011
[ +9.303503] pico: interrupted while waiting for dma
```

# TESTING THE BOARD ON srv-028

- We didn't manage to use the 4 Cluster version on the SB-852 (actually this firmware was experimental)
- With the 2 Cluster it takes 700 ms per trigger window



Trigger rate at 1.4 Hz 😞

```
[mjrodrig@np04-srv-028 ProtoDUNE-scripts]$ ./threadedbatchdemo -i test/ -s
tinylinknet_20200528.bin -r 1024x2560x1 -f 3 -C 4 -B
=====
ie_init: Initialize Micron DLA system
DLA binary to be read is tinylinknet_20200528.bin
-----
Using FPGA 0x511 Device 0511
^C
```

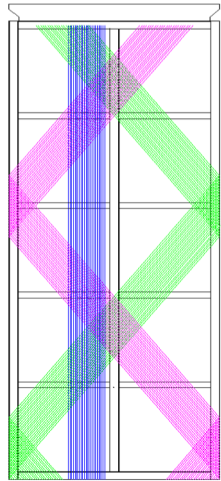
```
[Jun16 16:31] pico: bad fpga seq for fpga 1 stream 254! expected 0x190,
got 0x180. last_host_seq: 0x180 (desc seq: 0x190)
[ +0.000003] pico:pico_newfw(): pico_newfw_internal() return error: -
10011
[ +0.000002] pico: couldn't send 'read' command to system PicoBus: -10011
[ +9.303503] pico: interrupted while waiting for dma
```

# THE REORDERING ISSUE

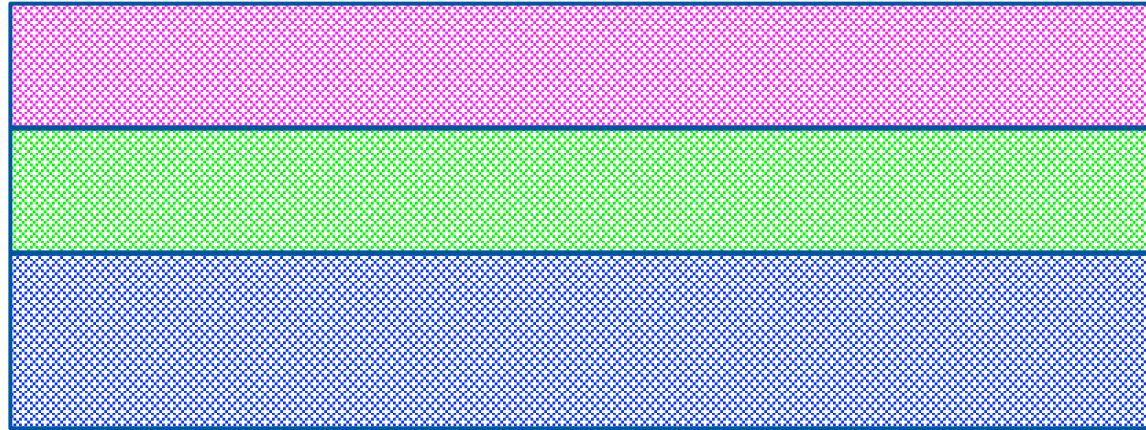
- After all the issues, we managed to send data to the FPGA, in a one shot approach.
- However, we found a totally different issue.

# THE REORDERING ISSUE

- Our images in our dataset are like this:

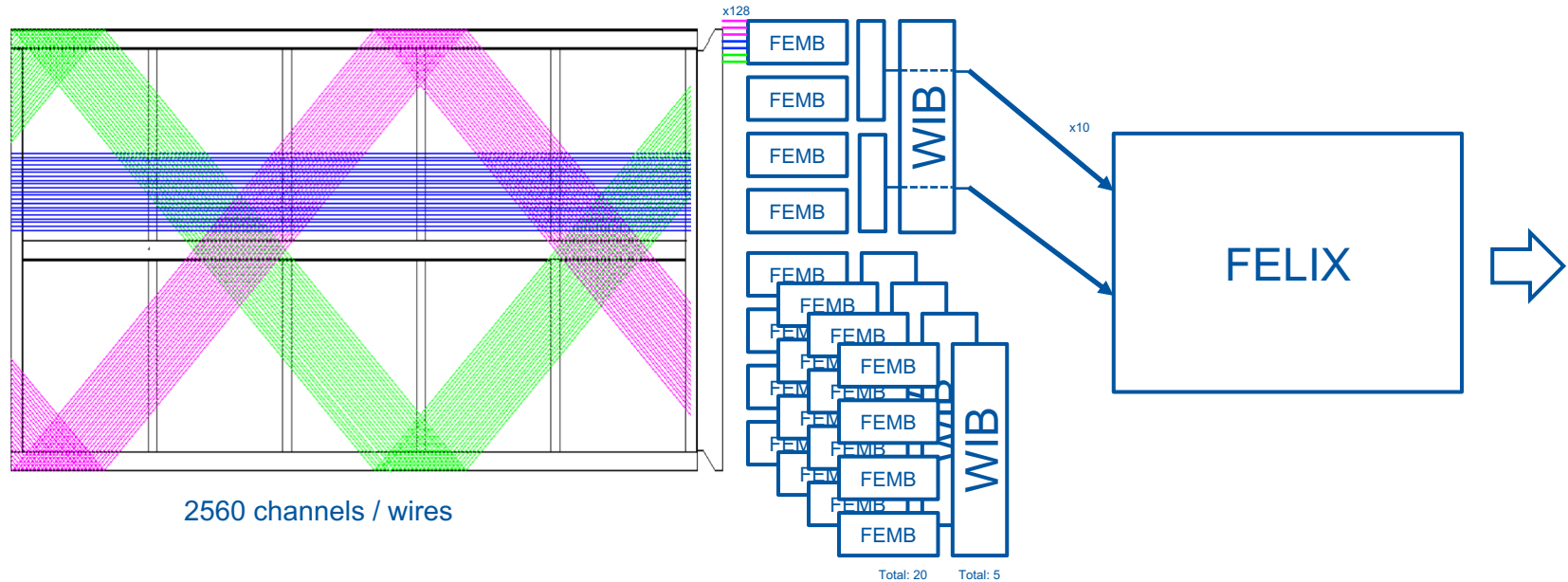


Channels

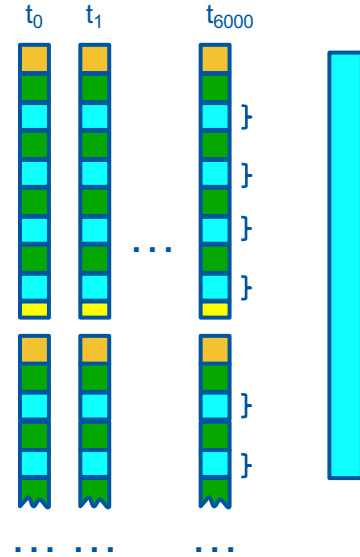
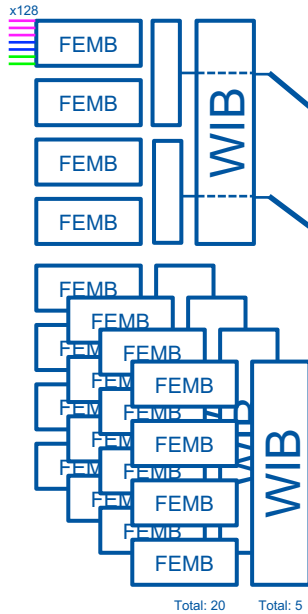


Time  
window

# THE REORDERING ISSUE



# THE REORDERING ISSUE



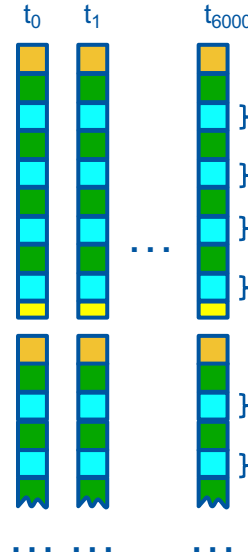
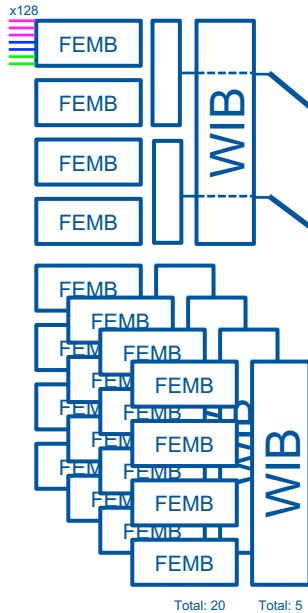
Bit

0	0x0	0x0	0x0	SOF	
1	Reserved (8)	SlotNo	CrateNo	FiberNo/Version = 0x1	0x0
2	WIB Errors			Reserved (14)	0x0
3	Timestamp [31:0]				
4	Timestamp [62:48] or WIB counter			Timestamp [47:32]	
5	COLDATA Block 1				
33	COLDATA Block 2				
61	COLDATA Block 3				
89	COLDATA Block 4				
117	CRC-20 [19:0]			EOF	
118	0x0	0x0	0x0	K28.5	
119	0x0	0x0	0x0	K28.5	

Bit

1	Chan B [7:0]	Chan A [7:0]	Reserved (8)	Chan B [15:8]	Chan A [15:8]
2	COLDATA Count Count		Error Register		
3	Reserved (16)				
4	ADC0	ADC1	ADC2	ADC3	ADC4
5	ADC2 CH[23:0]	ADC1 CH[11:8]	ADC0 CH[3:0]	ADC3 CH[11:8]	ADC4 CH[11:8]
6	ADC2 CH[3:0]	ADC1 CH[3:0]	ADC2 CH[11:4]	ADC1 CH[11:4]	ADC0 CH[11:4]
7	ADC2 CH[11:4]	ADC1 CH[11:4]	ADC2 CH[3:0]	ADC1 CH[3:0]	ADC0 CH[3:0]
28	ADC3 CH[11:4]	ADC7 CH[11:4]	ADC8 CH[3:0]	ADC9 CH[3:0]	ADC7 CH[11:8]

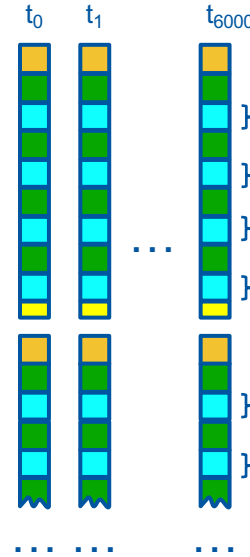
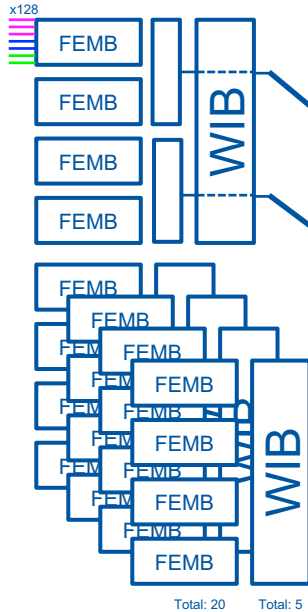
# THE REORDERING ISSUE



Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Word 0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
Word 1	Reserved (8)	SlotNo	CrateNo	FiberNo/Version = 0x1	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
Word 2	WIB Errors				Reserved (14)				Timestamp [31:0]				Timestamp [47:32]				Timestamp [62:48] or WIB counter				Timestamp [47:32]				Timestamp [62:48] or WIB counter							
Word 3	COLDATA Count				COLDATA Count				COLDATA Count				COLDATA Count				COLDATA Count				COLDATA Count											
Word 4	ADC1 CH[11:8]				ADC1 CH[7:5]				ADC1 CH[3:0]				ADC1 CH[11:8]				ADC1 CH[7:5]				ADC1 CH[3:0]				ADC1 CH[11:8]							
Word 5	ADC2 CH[11:8]				ADC2 CH[7:5]				ADC2 CH[3:0]				ADC2 CH[11:8]				ADC2 CH[7:5]				ADC2 CH[3:0]				ADC2 CH[11:8]							
Word 6	ADC3 CH[11:8]				ADC3 CH[7:5]				ADC3 CH[3:0]				ADC3 CH[11:8]				ADC3 CH[7:5]				ADC3 CH[3:0]				ADC3 CH[11:8]							
Word 7	ADC4 CH[11:8]				ADC4 CH[7:5]				ADC4 CH[3:0]				ADC4 CH[11:8]				ADC4 CH[7:5]				ADC4 CH[3:0]				ADC4 CH[11:8]							
Word 8	ADC5 CH[11:8]				ADC5 CH[7:5]				ADC5 CH[3:0]				ADC5 CH[11:8]				ADC5 CH[7:5]				ADC5 CH[3:0]				ADC5 CH[11:8]							
Word 9	ADC6 CH[11:8]				ADC6 CH[7:5]				ADC6 CH[3:0]				ADC6 CH[11:8]				ADC6 CH[7:5]				ADC6 CH[3:0]				ADC6 CH[11:8]							
Word 10	ADC7 CH[11:8]				ADC7 CH[7:5]				ADC7 CH[3:0]				ADC7 CH[11:8]				ADC7 CH[7:5]				ADC7 CH[3:0]				ADC7 CH[11:8]							
Word 11	ADC8 CH[11:8]				ADC8 CH[7:5]				ADC8 CH[3:0]				ADC8 CH[11:8]				ADC8 CH[7:5]				ADC8 CH[3:0]				ADC8 CH[11:8]							
Word 12	ADC9 CH[11:8]				ADC9 CH[7:5]				ADC9 CH[3:0]				ADC9 CH[11:8]				ADC9 CH[7:5]				ADC9 CH[3:0]				ADC9 CH[11:8]							
Word 13	ADC10 CH[11:8]				ADC10 CH[7:5]				ADC10 CH[3:0]				ADC10 CH[11:8]				ADC10 CH[7:5]				ADC10 CH[3:0]				ADC10 CH[11:8]							
Word 14	ADC11 CH[11:8]				ADC11 CH[7:5]				ADC11 CH[3:0]				ADC11 CH[11:8]				ADC11 CH[7:5]				ADC11 CH[3:0]				ADC11 CH[11:8]							
Word 15	ADC12 CH[11:8]				ADC12 CH[7:5]				ADC12 CH[3:0]				ADC12 CH[11:8]				ADC12 CH[7:5]				ADC12 CH[3:0]				ADC12 CH[11:8]							
Word 16	ADC13 CH[11:8]				ADC13 CH[7:5]				ADC13 CH[3:0]				ADC13 CH[11:8]				ADC13 CH[7:5]				ADC13 CH[3:0]				ADC13 CH[11:8]							
Word 17	ADC14 CH[11:8]				ADC14 CH[7:5]				ADC14 CH[3:0]				ADC14 CH[11:8]				ADC14 CH[7:5]				ADC14 CH[3:0]				ADC14 CH[11:8]							
Word 18	ADC15 CH[11:8]				ADC15 CH[7:5]				ADC15 CH[3:0]				ADC15 CH[11:8]				ADC15 CH[7:5]				ADC15 CH[3:0]				ADC15 CH[11:8]							
Word 19	ADC16 CH[11:8]				ADC16 CH[7:5]				ADC16 CH[3:0]				ADC16 CH[11:8]				ADC16 CH[7:5]				ADC16 CH[3:0]				ADC16 CH[11:8]							
Word 20	ADC17 CH[11:8]				ADC17 CH[7:5]				ADC17 CH[3:0]				ADC17 CH[11:8]				ADC17 CH[7:5]				ADC17 CH[3:0]				ADC17 CH[11:8]							
Word 21	ADC18 CH[11:8]				ADC18 CH[7:5]				ADC18 CH[3:0]				ADC18 CH[11:8]				ADC18 CH[7:5]				ADC18 CH[3:0]				ADC18 CH[11:8]							
Word 22	ADC19 CH[11:8]				ADC19 CH[7:5]				ADC19 CH[3:0]				ADC19 CH[11:8]				ADC19 CH[7:5]				ADC19 CH[3:0]				ADC19 CH[11:8]							
Word 23	ADC20 CH[11:8]				ADC20 CH[7:5]				ADC20 CH[3:0]				ADC20 CH[11:8]				ADC20 CH[7:5]				ADC20 CH[3:0]				ADC20 CH[11:8]							
Word 24	ADC21 CH[11:8]				ADC21 CH[7:5]				ADC21 CH[3:0]				ADC21 CH[11:8]				ADC21 CH[7:5]				ADC21 CH[3:0]				ADC21 CH[11:8]							
Word 25	ADC22 CH[11:8]				ADC22 CH[7:5]				ADC22 CH[3:0]				ADC22 CH[11:8]				ADC22 CH[7:5]				ADC22 CH[3:0]				ADC22 CH[11:8]							
Word 26	ADC23 CH[11:8]				ADC23 CH[7:5]				ADC23 CH[3:0]				ADC23 CH[11:8]				ADC23 CH[7:5]				ADC23 CH[3:0]				ADC23 CH[11:8]							
Word 27	ADC24 CH[11:8]				ADC24 CH[7:5]				ADC24 CH[3:0]				ADC24 CH[11:8]				ADC24 CH[7:5]				ADC24 CH[3:0]				ADC24 CH[11:8]							
Word 28	ADC25 CH[11:8]				ADC25 CH[7:5]				ADC25 CH[3:0]				ADC25 CH[11:8]				ADC25 CH[7:5]				ADC25 CH[3:0]				ADC25 CH[11:8]							
Word 29	ADC26 CH[11:8]				ADC26 CH[7:5]				ADC26 CH[3:0]				ADC26 CH[11:8]				ADC26 CH[7:5]				ADC26 CH[3:0]				ADC26 CH[11:8]							
Word 30	ADC27 CH[11:8]				ADC27 CH[7:5]				ADC27 CH[3:0]				ADC27 CH[11:8]				ADC27 CH[7:5]				ADC27 CH[3:0]				ADC27 CH[11:8]							
Word 31	ADC28 CH[11:8]				ADC28 CH[7:5]				ADC28 CH[3:0]				ADC28 CH[11:8]				ADC28 CH[7:5]				ADC28 CH[3:0]				ADC28 CH[11:8]							



# THE REORDERING ISSUE



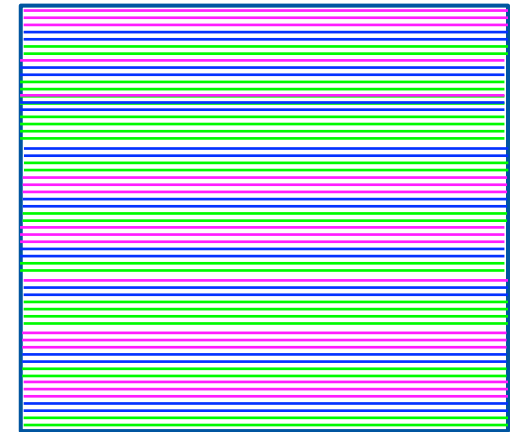
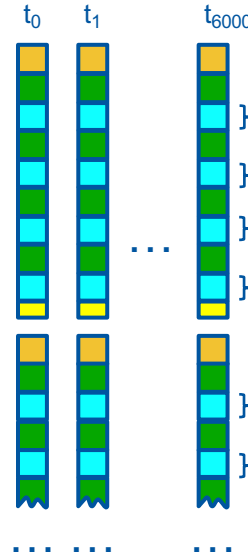
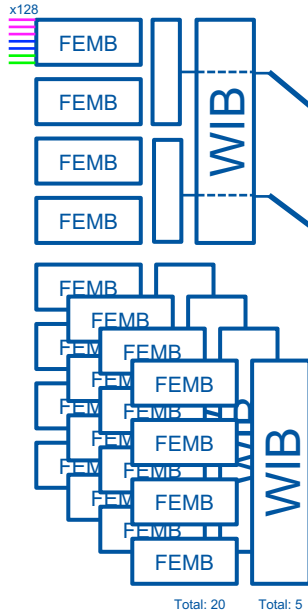
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31					
Word 0	0x0	0x0	0x0	SOF																																	
Word 1	Reserved (8)	SlotNo	CrateNo	FiberNo/Version = 0x1	0x0																																
Word 2	WIB Errors				Reserved (14)																																
Word 3	Timestamp [31:0]				Timestamp [47:32]																																
Word 4	COLDATA Block 1				COLDATA Block 2				COLDATA Block 3				COLDATA Block 4																								
Word 5	ADC0 CH[23:0]				ADC1 CH[111:8]				ADC2 CH[97:0]				ADC3 CH[117:8]				ADC4 CH[17:8]																				
Word 6	ADC2 CH[97:0]				ADC1 CH[97:0]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 7	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 8	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 9	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 10	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 11	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 12	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 13	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 14	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 15	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 16	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 17	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 18	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 19	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 20	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 21	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 22	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 23	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 24	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 25	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 26	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 27	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 28	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 29	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 30	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																
Word 31	ADC2 CH[111:4]				ADC1 CH[111:4]				ADC2 CH[211:4]				ADC1 CH[211:4]				ADC3 CH[3:0]				ADC4 CH[11:8]																

**NOT CORRECT**

ADC values for the time window



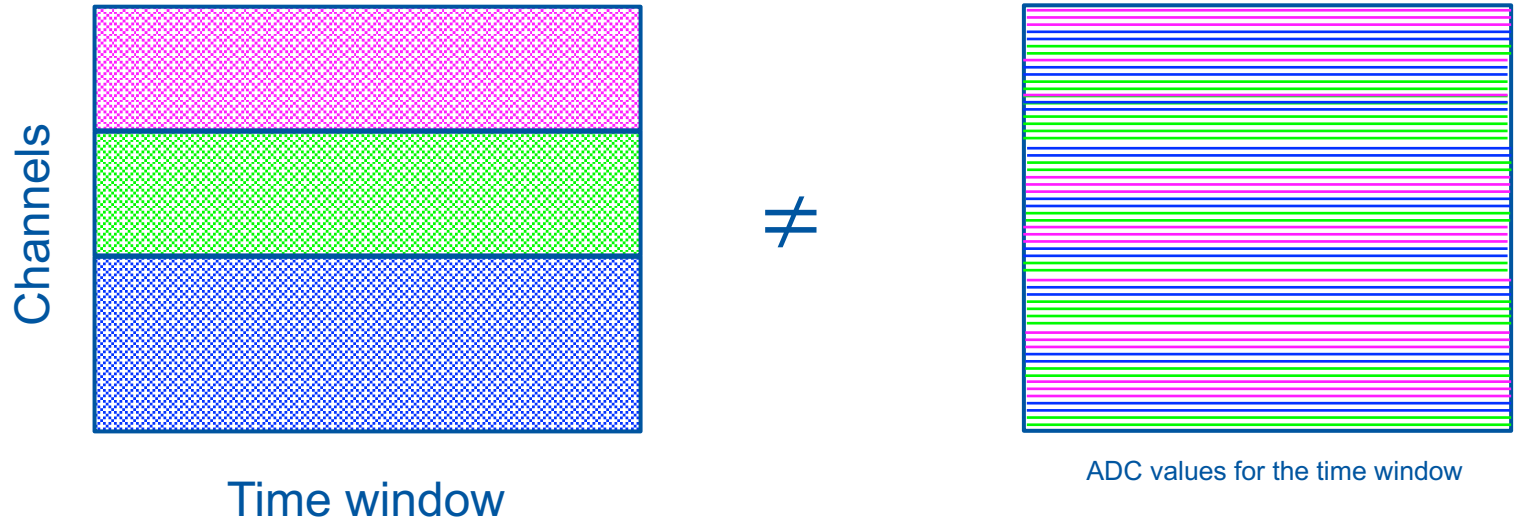
# THE REORDERING ISSUE



ADC values for the time window

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Word 0	0x0	0x0	0x0	SOF																												
Word 1	Reserved (8)	SlotNo	CrateNo	FiberNo/Version = 0x1	0x0																											
Word 2	WIB Errors														Reserved (14)																	
Word 3	Timestamp [31:0]														Timestamp [47:32]																	
Word 4	COLDATA Block 1														COLDATA Block 2		COLDATA Block 3		COLDATA Block 4													
Word 5	CRC-32 [19:0]														EOF																	
Word 6	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	
Word 7	ADC0	ADC1	ADC2	ADC3	ADC4	ADC5	ADC6	ADC7	ADC8	ADC9	ADC10	ADC11	ADC12	ADC13	ADC14	ADC15	ADC16	ADC17	ADC18	ADC19	ADC20	ADC21	ADC22	ADC23	ADC24	ADC25	ADC26	ADC27	ADC28	ADC29	ADC30	
Word 8	ADC31	ADC32	ADC33	ADC34	ADC35	ADC36	ADC37	ADC38	ADC39	ADC40	ADC41	ADC42	ADC43	ADC44	ADC45	ADC46	ADC47	ADC48	ADC49	ADC50	ADC51	ADC52	ADC53	ADC54	ADC55	ADC56	ADC57	ADC58	ADC59	ADC60	ADC61	
Word 9	ADC62	ADC63	ADC64	ADC65	ADC66	ADC67	ADC68	ADC69	ADC70	ADC71	ADC72	ADC73	ADC74	ADC75	ADC76	ADC77	ADC78	ADC79	ADC80	ADC81	ADC82	ADC83	ADC84	ADC85	ADC86	ADC87	ADC88	ADC89	ADC90	ADC91	ADC92	
Word 10	ADC93	ADC94	ADC95	ADC96	ADC97	ADC98	ADC99	ADC100	ADC101	ADC102	ADC103	ADC104	ADC105	ADC106	ADC107	ADC108	ADC109	ADC110	ADC111	ADC112	ADC113	ADC114	ADC115	ADC116	ADC117	ADC118	ADC119	ADC120	ADC121	ADC122	ADC123	

# THE REORDERING ISSUE



# THE REORDERING ISSUE

- Removing the headers is fine
- Reordering the data (2560 channels time 6000 ticks) is not.
- Possible solutions:
  - Retrain the network using the online channel number
  - Do the reorder on FPGA (FELIX or Inference Engine)

# CONCLUSIONS

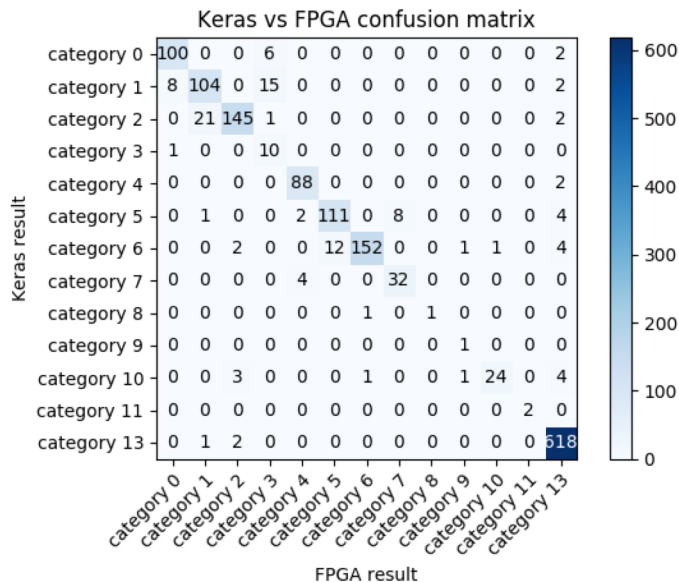
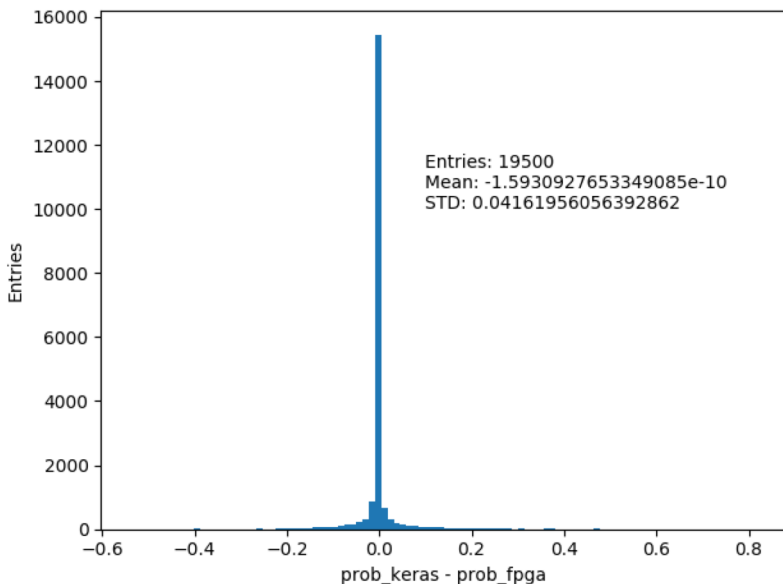
- We wanted to test the integration of the Micron DLA on the protoDUNE DAQ chain.
- The hardware we used was an unreleased version made for this test with some issues unseen before. Thanks to the test, Micron can study it and debug it to make the system more robust.
- We faced as well a different issue not taken into account. In my opinion, it was great that we worked with online raw data. This gave us a much better understanding on how that data is coming from the detector.
- It's a pity that we don't have more time to test. However, thanks to the binary data recorded we can continue evolving the system.

**THANK YOU**

**BACKUP**

# GPU-FPGA RESULTS COMPARISON

*How good our FPGA behaves*



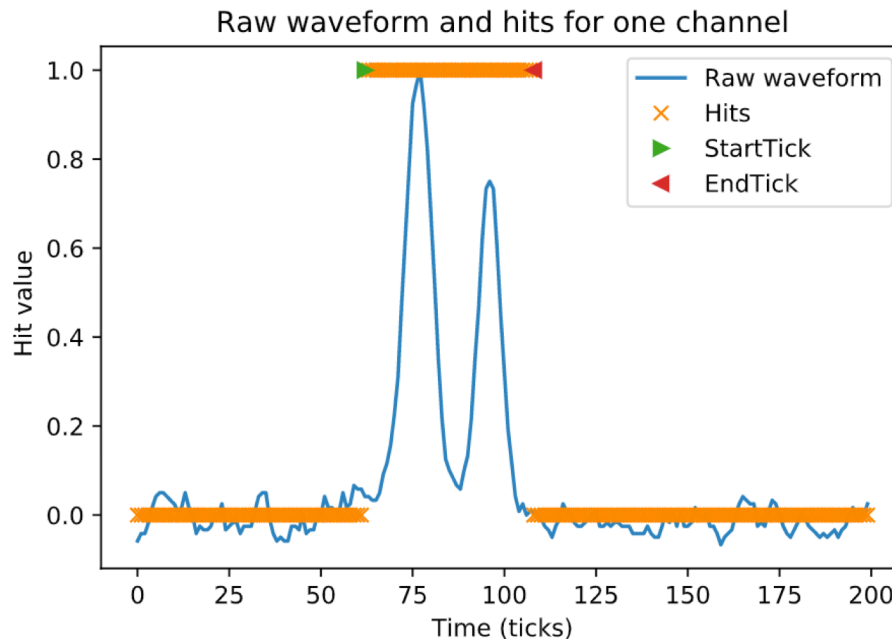
# Our dataset

- On the hits file we have:

```
(int)hit.Channel(), hit.StartTick(),  
hit.EndTick(), (int)hit.SummedADC(),  
(int)hit.RMS()
```

- We take the StartTick and EndTick and we mark the whole range as

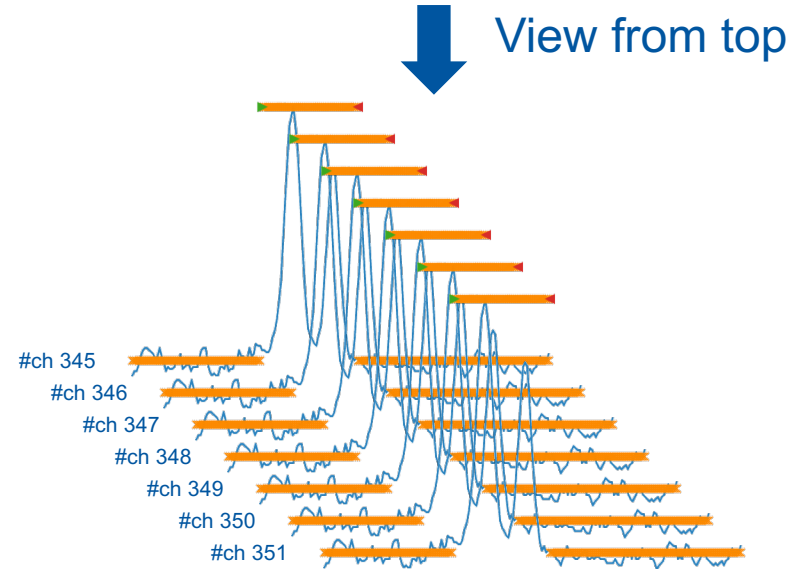
```
hit(channel,[startTick,endTick]) = TRUE
```





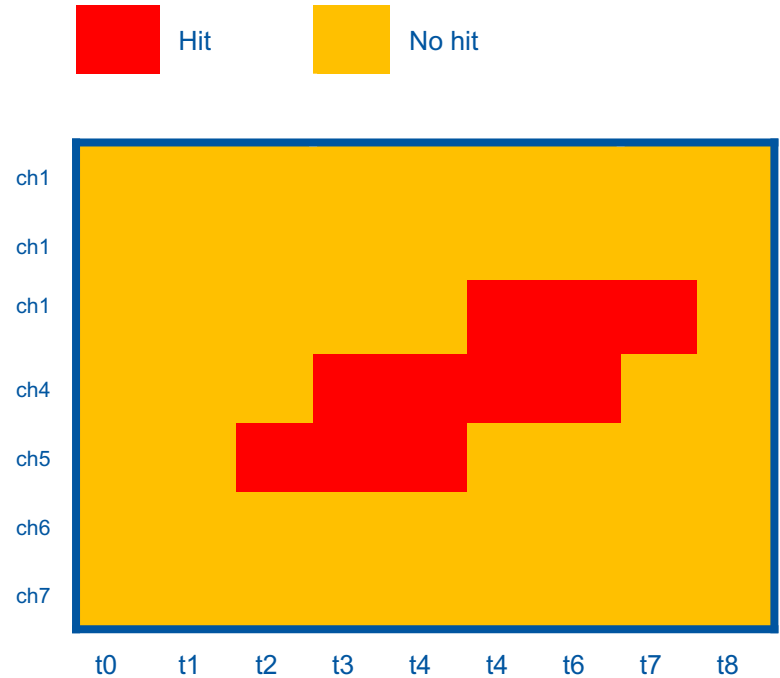
# Region of interest

- Once we have the hits for all the channels.
- We artificially augment the hits area in time and channels to get our region of interest.
- $\forall i, j$ : If  $\text{hit}(i, j) == 1$ 
  - $\text{hit}(i + 1, j) = 1$
  - $\text{hit}(i - 1, j) = 1$
  - $\text{hit}(i, j + 1) = 1$
  - $\text{hit}(i, j - 1) = 1$
  - $\text{hit}(i + 1, j + 1) = 1$
  - [...]



# Region of interest

- Once we have the hits for all the channels
- We artificially augment the hits area in time and channels to get our region of interest
- $\forall i, j$ : If  $\text{hit}(i, j) == 1$ 
  - $\text{hit}(i + 1, j) = 1$
  - $\text{hit}(i - 1, j) = 1$
  - $\text{hit}(i, j + 1) = 1$
  - $\text{hit}(i, j - 1) = 1$
  - $\text{hit}(i + 1, j + 1) = 1$
  - [...]



# Region of interest

- Once we have the hits for all the channels
- We artificially augment the hits area in time and channels to get our region of interest
- $\forall i, j$ : If  $\text{hit}(i, j) == 1$ 
  - $\text{hit}(i + 1, j) = 1$
  - $\text{hit}(i - 1, j) = 1$
  - $\text{hit}(i, j + 1) = 1$
  - $\text{hit}(i, j - 1) = 1$
  - $\text{hit}(i + 1, j + 1) = 1$
  - [...]
- We use the augmented area as our ground truth for the neural network

