Introduction to the Data Acquisition System (Part Deux: Ils reviennent ...)

Martin Schwinzerl, Seyma Essen

Beamline For Schools (BL4S)
CERN (IR-ECO-TSP), Meyrin, Switzerland

September 15st, 2024

What are we about to discuss

1. We have to talk about the trigger
   - Data Fragments & Event-building
   - Synchronous Data Acquisition
   - Hidden Bonus Discussions: Per-Particle Information vs. Collective Data, De-Randomisation & Buffering
2. Combining two Data Acquisition Systems (Sakura Particles)
3. Reading Out The Oscilloscope (Both Teams)
4. The plan for the next days

Short recap from yesterday

- Our setup consists of three main components
  1. The detectors (at the moment only "ours")
  2. The trigger logic unit (aka "The confusingly cabled NIM crate")
  3. The digitizers (aka, mostly "The VME crate with the modules")
  4. Everything else (XSCA tables, gas supply, power supply, HV supply, LASER cross for alignment, concrete blocks, computers, ...)
- We are about to add your detectors to the system and - ideally - perform measurements with all components at the same time
- In order to do so, we have to talk a bit about the trigger
- But first to something not completely different

Short recap from yesterday

- Our setup consists of three main components
  1. The detectors (at the moment only "ours")
  2. The trigger logic unit (aka "The confusingly cabled NIM crate")
  3. The digitizers (aka, mostly "The VME crate with the modules")
  4. Everything else (XSCA tables, gas supply, power supply, HV supply, LASER cross for alignment, concrete blocks, computers, ...)
- We are about to add your detectors to the system and - ideally - perform measurements with all components at the same time
- In order to do so, we have to talk a bit about the trigger
- But first to something not completely different

Short recap from yesterday

- Our setup consists of three main components
  1. The detectors (at the moment only "ours")
  2. The trigger logic unit (aka "The confusingly cabled NIM crate")
  3. The digitizers (aka, mostly "The VME crate with the modules")
  4. Everything else (XSCA tables, gas supply, power supply, HV supply, LASER cross for alignment, concrete blocks, computers, ...)
- We are about to add your detectors to the system and - ideally - perform measurements with all components at the same time
- In order to do so, we have to talk a bit about the trigger
- But first to something not completely different

Short recap from yesterday

- Our setup consists of three main components
  1. The detectors (at the moment only "ours")
  2. The trigger logic unit (aka "The confusingly cabled NIM crate")
  3. The digitizers (aka, mostly "The VME crate with the modules")
  4. Everything else (XSCA tables, gas supply, power supply, HV supply, LASER cross for alignment, concrete blocks, computers, ...)
- We are about to add your detectors to the system and - ideally - perform measurements with all components at the same time
- In order to do so, we have to talk a bit about the trigger
- But first to something not completely different

Short recap from yesterday

- Our setup consists of three main components
  1. The detectors (at the moment only "ours")
  2. The trigger logic unit (aka "The confusingly cabled NIM crate")
  3. The digitizers (aka, mostly "The VME crate with the modules")
  4. Everything else (XSCA tables, gas supply, power supply, HV supply, LASER cross for alignment, concrete blocks, computers, ...)
- We are about to add your detectors to the system and - ideally - perform measurements with all components at the same time
- In order to do so, we have to talk a bit about the trigger
- But first to something not completely different

What is the purpose of Data Acquisition & Data Analysis?

- As we have seen in your brilliant introduction presentations yesterday, you have all already a lot of experience with performing experiments and with evaluating the data. What do you think?
- Or, to start at a slightly different place:
  What is the purpose of (numerical) computing?

What is the purpose of Data Acquisition & Data Analysis?

- As we have seen in your brilliant introduction presentations yesterday, you have all already a lot of experience with performing experiments and with evaluating the data. What do you think?
- Or, to start at a slightly different place:
  What is the purpose of (numerical) computing?

What is the purpose of Data Acquisition & Data Analysis?

- As we have seen in your brilliant introduction presentations yesterday, you have all already a lot of experience with performing experiments and with evaluating the data. What do you think?
- Or, to start at a slightly different place:
  What is the purpose of (numerical) computing?

> *Numerical methods use numbers to simulate mathematical processes, which in turn usually simulate real-world situations. This implies that there is a purpose behind the computing. To cite the motto of the book,*
>
> **The Purpose of Computing Is Insight, Not Numbers.**
>
> R. W. Hamming, Numerical Methods for Scientists and Engineers, 2nd Edition, 1962.

Let's assume you want to capture some data ...

- Acquiring data and information uses lots of resources.
- Particles are passing through our setup one-by-one, we can not exactly "predict" when each of these individual particles will arrive.
- But we have a pretty good idea of typical time between particles:

Let's assume you want to capture some data ...

- Acquiring data and information uses lots of resources.
  Doing something else concurrently is difficult!
- Particles are passing through our setup one-by-one, we can not exactly "predict" when each of these individual particles will arrive.
- But we have a pretty good idea of typical time between particles:

Let's assume you want to capture some data ...

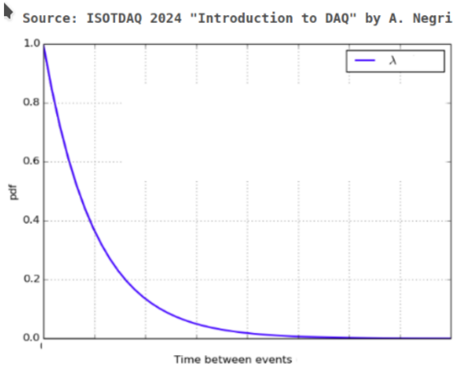
- Acquiring data and information uses lots of resources.
  Doing something else concurrently is difficult!
  (General statement, not limited to our version of data acquisition)
- Particles are passing through our setup one-by-one, we can not exactly "predict" when each of these individual particles will arrive.
- But we have a pretty good idea of typical time between particles:

Let's assume you want to capture some data ...

- Acquiring data and information uses lots of resources.
  Doing something else concurrently is difficult!
  (General statement, not limited to our version of data acquisition)
- Particles are passing through our setup one-by-one, we can not exactly "predict" when each of these individual particles will arrive.
- But we have a pretty good idea of typical time between particles:

Let's assume you want to capture some data ...

- Acquiring data and information uses lots of resources.
  Doing something else concurrently is difficult!
  (General statement, not limited to our version of data acquisition)
- Particles are passing through our setup one-by-one, we can not exactly "predict" when each of these individual particles will arrive.
- But we have a pretty good idea of typical time between particles:

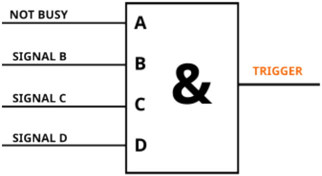


Data Acquisition (DAQ) concepts

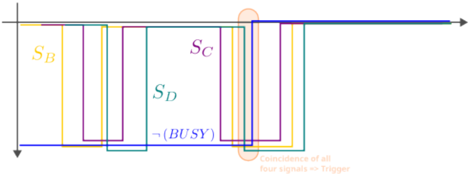
- **Idea #1**: Record as much data as possible and filter out the interesting bits from the raw data at a later stage ⇒ Triggerless DAQ
- **Idea #2**: Only start recording if something "interesting" happens in the first place. ⇒ Synchronous Data Acquisition.
- In both cases, we need to ensure that we can keep up with the incoming flow of data. In a synchronous DAQ system, we call this the ⇒ Busy logic

Data Acquisition (DAQ) concepts

- **Idea #1**: Record as much data as possible and filter out the interesting bits from the raw data at a later stage ⇒ Triggerless DAQ
- **Idea #2**: Only start recording if something “interesting” happens in the first place. ⇒ Synchronous Data Acquisition.
- In both cases, we need to ensure that we can keep up with the incoming flow of data. In a synchronous DAQ system, we call this the ⇒ Busy logic

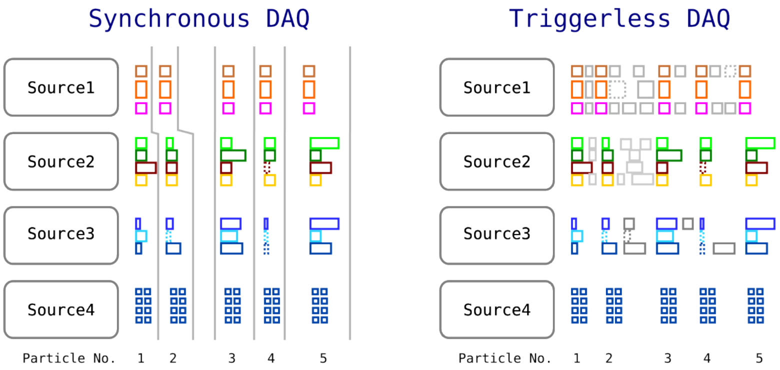


$$\text{Trigger} = S_B \;\&\; S_C \;\&\; S_D \;\&\; \neg(BUSY)$$



Data Fragments & Event Building

- The information for one particle is collected as fragments of data from different sources
- ⇒ TDC, QDC, Oscilloscope, Sakura Particle DAQ, ...
- If we would like to have a complete picture, we have to “bundle” these fragments together so we can build an event



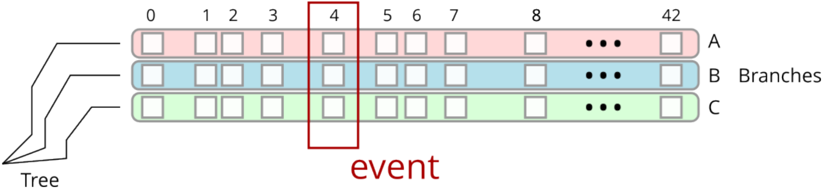
That sounds too easy? Voilà: Problems!

- Signals arrive at different times, not always in the order when we need them
- Taking a measurement itself can take a variable amount of time
- The efficiency of detectors is ususally $\leq 100\%$ percent
- Not all digitizers can help you filter out when nothing happend on a certain channel (zero suppression), so you need to do this in data analysis
- Generally: we shouldn't "mix" results from several individual particles
- We do not only get signals, we also receive noise and other disturbances (reflections, interference from other sources, etc.)

All these things occur in both approaches. But in general, in the synchronous approach it is easier to handle these kind of issues, so this is what we try to do with our setup.

Side Remark: Connection to the ROOT file

- The structure seen in the event building illustration should be a bit familiar to you:



- For our dataset:
  - Branches in a ROOT tree correspond to the results from digitizer channels
  - Entries in a ROOT tree correspond to the particles (events)
  - The values stored in the branch of a ROOT tree for an entry are the measured values for each particle.
- During data analysis, you will iterate through all the results from a measurement and try to find the data for particles that you are interested in

Side Remark: Connection to the ROOT file

From the log book `https://cern.ch/bl4s-2024-logbook-part2`:

Signal	TDC0	TDC1	QDC0	Scaler0	Oscilloscope
S0	0			1 1)	
S1	1			3	
XCET040	2		0	4	
XCET043	3		1	5	
FS0	4				
FS1	5				
S2	6		2	6	
S3	7		3	7	7
CAL17 (was CAL2)	8		4	8	8
S0&S1 4)	9			9	
S1&S2&S3 2)	11	11		10	
S2&S3	10	12		11	
XCET040 & S0 & S1	12			12	
XCET043 & S0 & S1	13			13	
FS0&FS1		10		14	

Signal	TDC0	TDC1	QDC0	Scaler0	Oscilloscope
DWC_UP L, R(cable can be changed), U, D		0-3			
DWC_DOWN L, R, U, D		4-7			
DWC_UP AV & AH	14	8			
DWC_DOWN AV & AH	15	9			
WENDI					
TRIGGER 3)				2	
1 kHz Strobe				15	
SAKURA Left Side (X1) 5)					1
SAKURA Right Side (X2) 5)					2
SAKURA Upper Side (Y1) 5)					3
SAKURA Lower Side (Y2) 5)					4
Mavericks #1					5
Mavericks #2					6

- The Signal Table tells you which channel (number) of which digitizer module is connected to which signal.
- This mapping can change, please check which version of the signal table we used during the measurement.
- Where can we find this information?

Side Remark: Connection to the ROOT file

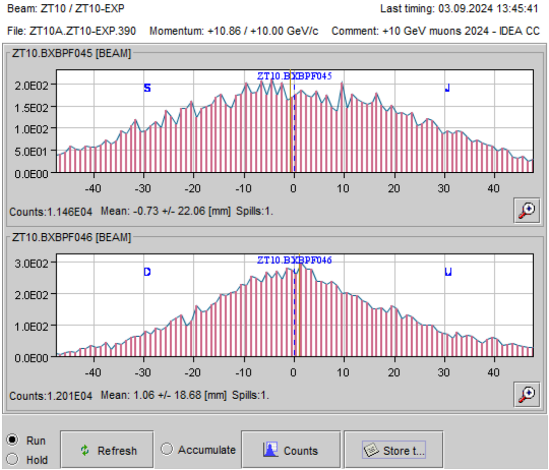
From the log book https://cern.ch/bl4s-2024-logbook-part2:

Signal	TDC0	TDC1	QDC0	Scaler0	Oscilloscope
S0	0			1[1)]	
S1	1			3	
XCET040	2		0	4	
XCET043	3		1	5	
FS0	4				
FS1	5				
S2	6		2	6	
S3	7		3	7	7
CAL17 (was CAL2)	8		4	8	8
S0&S1 [4)]	9			9	
S1&S2&S3 [2)]	11	11		10	
S2&S3	10	12		11	
XCET040 & S0 & S1	12			12	
XCET043 & S0 & S1	13			13	
FS0&FS1		10		14	

Signal	TDC0	TDC1	QDC0	Scaler0	Oscilloscope
DWC_UP L, R(cable can be changed), U, D		0-3			
DWC_DOWN L, R, U, D		4-7			
DWC_UP AV & AH	14	8			
DWC_DOWN AV & AH	15	9			
WENDI					
TRIGGER [3)]				2	
1 kHz Strobe				15	
SAKURA Left Side (X1)[5)]					1
SAKURA Right Side (X2)[5)]					2
SAKURA Upper Side (Y1)[5)]					3
SAKURA Lower Side (Y2)[5)]					4
Mavericks #1					5
Mavericks #2					6

- The Signal Table tells you which channel (number) of which digitizer module is connected to which signal.
- This mapping can change, please check which version of the signal table we used during the measurement.
- Where can we find this information? Also in the log book!

Side Remark: Collective Data vs. Particle-By-Particle Data



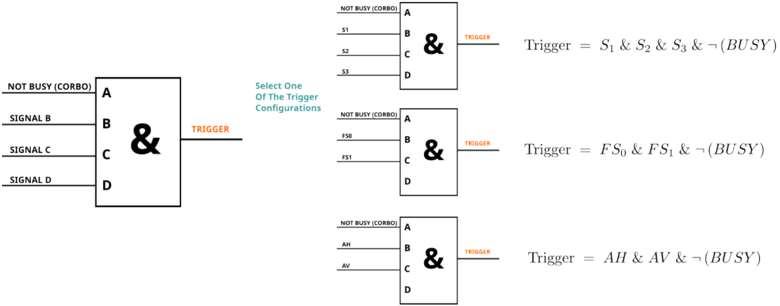
Side Remark: Collective Data vs. Particle-By-Particle Data

Side Remark: Collective Data vs. Particle-By-Particle Data

Side Remark: Collective Data vs. Particle-By-Particle Data

Synchronous DAQ & our setup

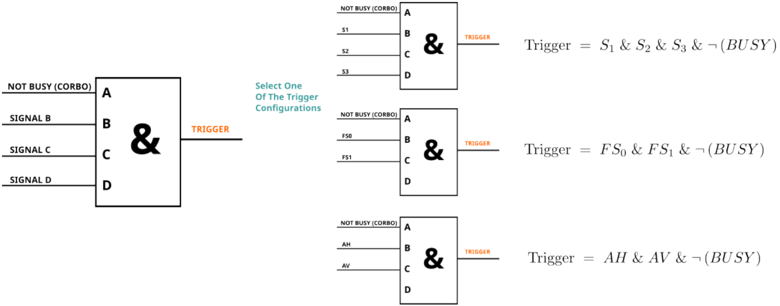
- Data acquisition is a process with more than one step:
  - Transformation of signals (delay, discrimination, ...)
  - Digitizing the signals themselves (aka: "The Measurement")
  - Transfer of the data to where we need it
- Form trigger & busy logic from signals in our system.
- Prepared triggers in our setup:



- Which trigger was used for which measurement?

Synchronous DAQ & our setup

- Data acquisition is a process with more than one step:
  - Transformation of signals (delay, discrimination, ...)
  - Digitizing the signals themselves (aka: "The Measurement")
  - Transfer of the data to where we need it
- Form trigger & busy logic from signals in our system.
- Prepared triggers in our setup:



- Which trigger was used for which measurement? ⇒ Log Book!

Why are we spending so much time on this?

1. It may be interesting?
2. We want to compare data captured by two DAQ systems
3. The oscilloscope requires some adaptations

Why are we spending so much time on this?

1. It may be interesting?
2. We want to compare data captured by two DAQ systems
3. The oscilloscope requires some adaptations

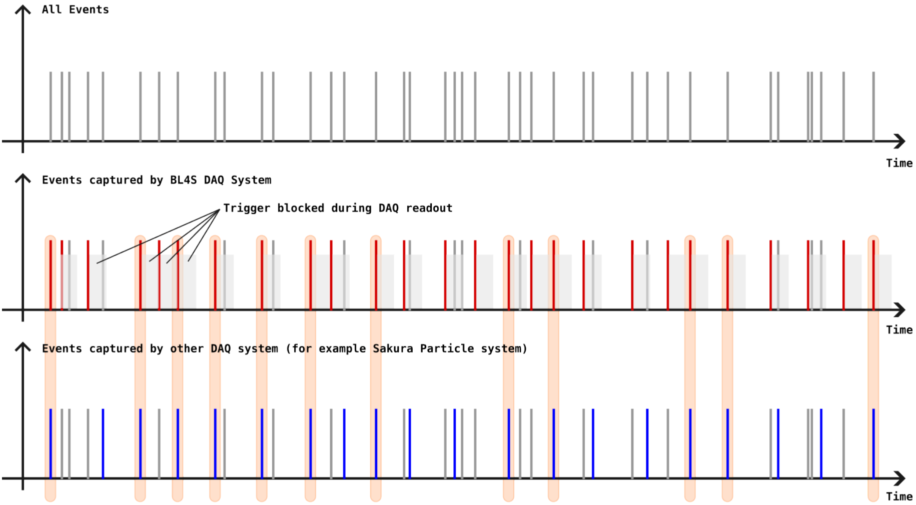
Why are we spending so much time on this?

1. It may be interesting?
2. We want to compare data captured by two DAQ systems
3. The oscilloscope requires some adaptations

Why are we spending so much time on this?

1. It may be interesting?
2. We want to compare data captured by two DAQ systems
3. The oscilloscope requires some adaptations

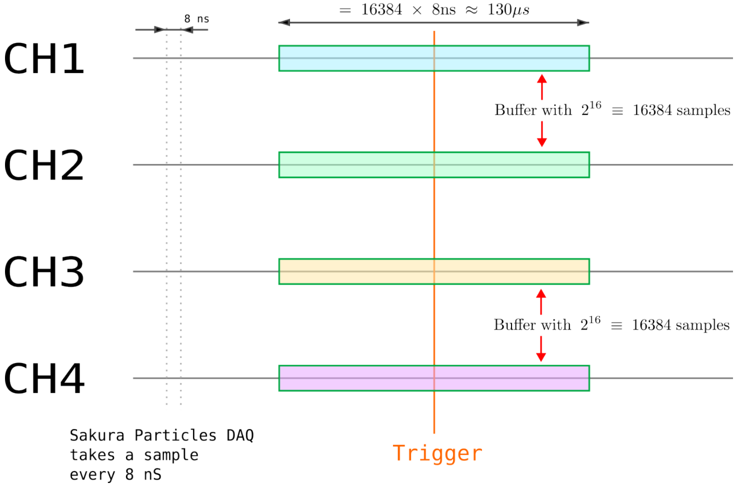
#2 Compare data captured across two DAQ systems



It is virtually guaranteed that only some of the particles will be captured by both systems without synchronisation (even if the data rates are very similar). It is not predictable which events are commonly recorded

#2 Compare data captured across two DAQ systems

Red Pitaya System (as we understand it)



- **Idea**: Share trigger, then both systems should "see" same particles

#2 Compare data captured across two DAQ systems

- We have prepared two TTL level trigger signals, latency should be within $\sim 100ns$
- Capture window width of $\sim 130\mu s$ on the Red Pitaya side should in principle work with our typical data rates (1-3 kHz).
- **But:** we do not currently share a **NOT BUSY** signal from the Sakura particles DAQ system. This may be a problem if the readout / writeout part takes signifcantly longer than $130\mu s$. To be discussed if this is something we should address.
- Alternative in case we run into issues with the data rate: increase the spill length by approx a factor of 2.
- $\Rightarrow$ As discussed yesterday by Maarten, we could in principle do this. It would be good if we could understand if we need this by Tuesday (needs preparation on the PS machine control side).

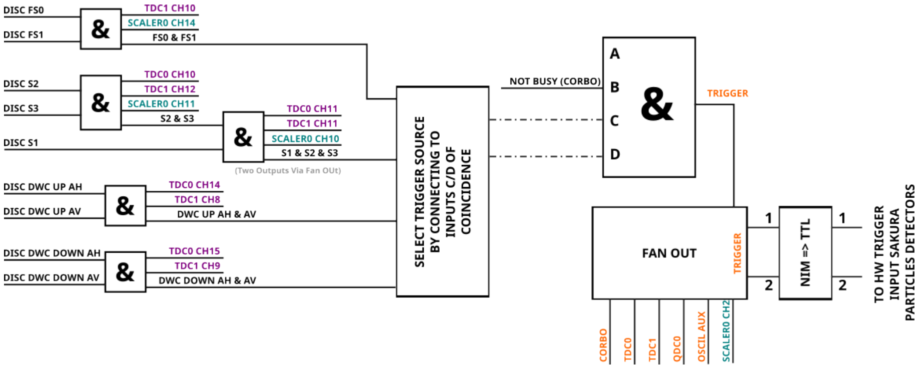
#2 Compare data captured across two DAQ systems

- We have prepared two TTL level trigger signals, latency should be within $\sim 100ns$
- Capture window width of $\sim 130\mu s$ on the Red Pitaya side should in principle work with our typical data rates (1-3 kHz).
- **But:** we do not currently share a **NOT BUSY** signal from the Sakura particles DAQ system. This may be a problem if the readout / writeout part takes signifcantly longer than $130\mu s$. To be discussed if this is something we should address.
- Alternative in case we run into issues with the data rate: increase the spill length by approx a factor of 2.
- $\Rightarrow$ As discussed yesterday by Maarten, we could in principle do this. It would be good if we could understand if we need this by Tuesday (needs preparation on the PS machine control side).

#2 Compare data captured across two DAQ systems

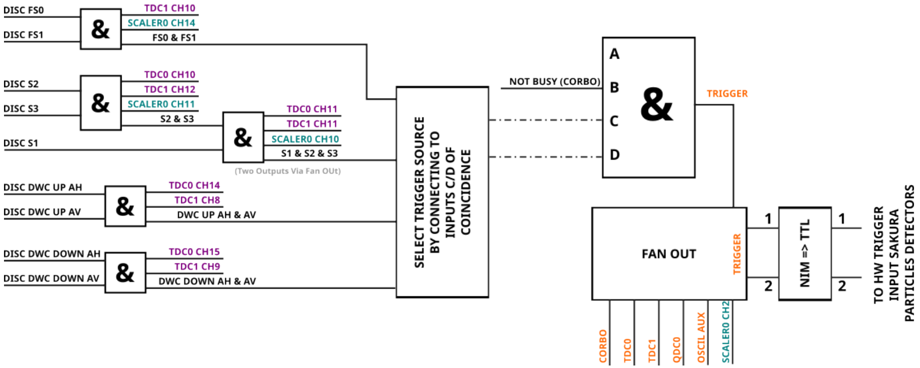
- We have prepared two TTL level trigger signals, latency should be within $\sim 100ns$
- Capture window width of $\sim 130\mu s$ on the Red Pitaya side should in principle work with our typical data rates (1-3 kHz).
- **But:** we do not currently share a **NOT BUSY** signal from the Sakura particles DAQ system. This may be a problem if the readout / writeout part takes signifcantly longer than $130\mu s$. To be discussed if this is something we should address.
- Alternative in case we run into issues with the data rate: increase the spill length by approx a factor of 2.
- $\Rightarrow$ As discussed yesterday by Maarten, we could in principle do this. It would be good if we could understand if we need this by Tuesday (needs preparation on the PS machine control side).

Full picture of the trigger configuration?



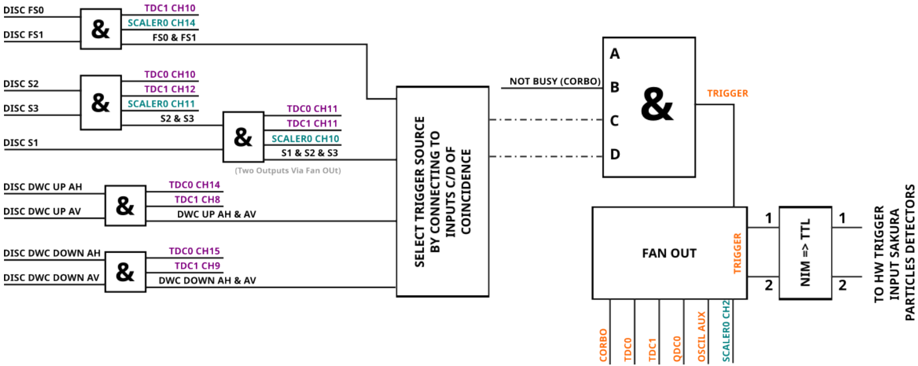
- Where can we find this picture?
- Is this really the full picture regarding the trigger?

Full picture of the trigger configuration?



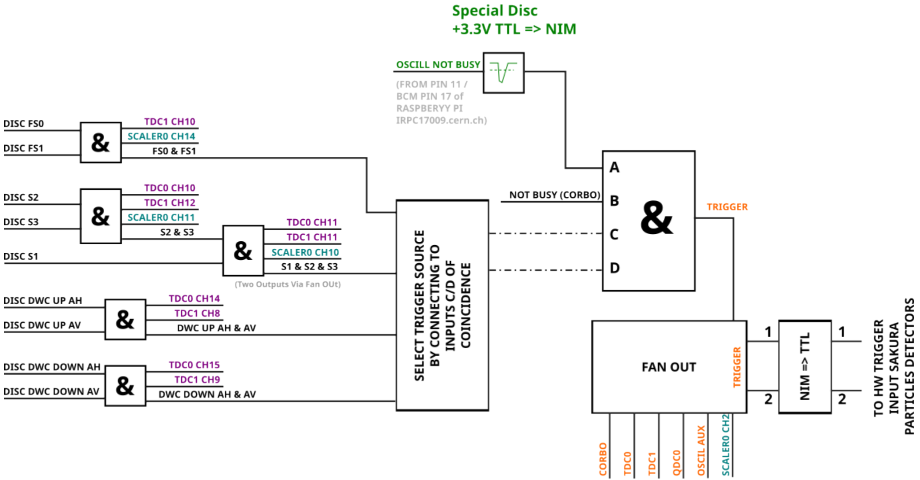
- Where can we find this picture? ⇒ In the log book!
- Is this really the full picture regarding the trigger?

Full picture of the trigger configuration?



- Where can we find this picture? ⇒ In the log book!
- Is this really the full picture regarding the trigger?

Full picture of the trigger configuration!



- We have not spoken about the Oscilloscope yet

Working with the Oscilloscope (Tektronix MSO58LP)

First: The Good News

- Very, very capable device & **we managed to rent one!**
- Sample rate of up to 6.25 GS/s
- Bandwidth up to 1 GHz
- 8 independent channels
- Supports data acquisition with shared external trigger signal
- Can be remote controlled via a browser!

Working with the Oscilloscope (Tektronix MSO58LP)

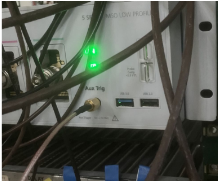
First: The Good News

- Very, very capable device & **we managed to rent one!**
- Sample rate of up to 6.25 GS/s
- Bandwidth up to 1 GHz
- 8 independent channels
- Supports data acquisition with shared external trigger signal
- Can be remote controlled via a browser!

Working with the Oscilloscope (Tektronix MSO58LP)

First: The Good News

- Very, very capable device & **we managed to rent one!**
- Sample rate of up to 6.25 GS/s
- Bandwidth up to 1 GHz
- 8 independent channels
- Supports data acquisition with shared external trigger signal

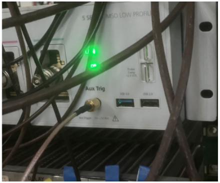


- Can be remote controlled via a browser!

Working with the Oscilloscope (Tektronix MSO58LP)

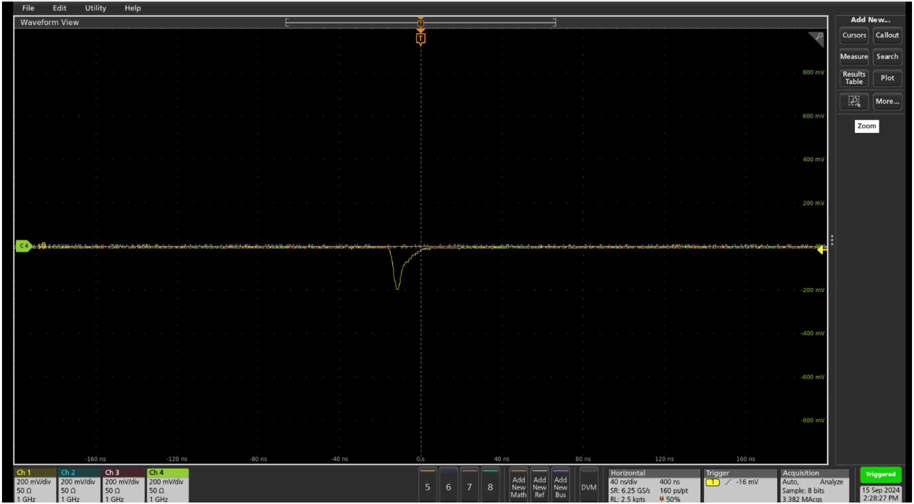
First: The Good News

- Very, very capable device & **we managed to rent one!**
- Sample rate of up to 6.25 GS/s
- Bandwidth up to 1 GHz
- 8 independent channels
- Supports data acquisition with shared external trigger signal



- Can be remote controlled via a browser!

Oscilloscope Remote Control



Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

- Oscilloscope can record particles into an internal buffer that is surprisingly large
- $\Rightarrow$ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal (($\Rightarrow$ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \ \times \ 8\,\text{channels} \ \times \ 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

- Oscilloscope can record particles into an internal buffer that is surprisingly large
- $\Rightarrow$ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal (($\Rightarrow$ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \ \times \ 8 \, \text{channels} \ \times \ 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

- Oscilloscope can record particles into an internal buffer that is surprisingly large
- ⇒ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal ((⇒ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \ \times \ 8\,\text{channels} \ \times \ 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

- Oscilloscope can record particles into an internal buffer that is surprisingly large
- $\Rightarrow$ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal (($\Rightarrow$ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \;\times\; 8\,\text{channels} \;\times\; 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

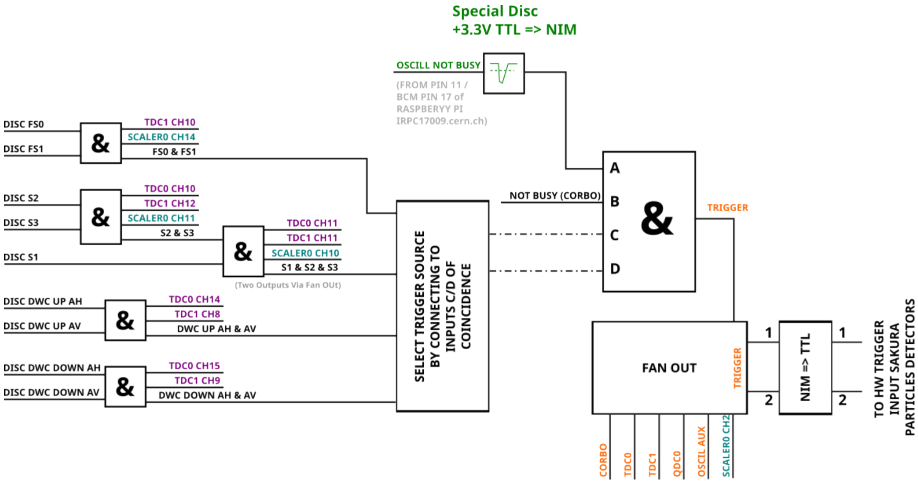
- Oscilloscope can record particles into an internal buffer that is surprisingly large
- $\Rightarrow$ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal (($\Rightarrow$ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \; \times \; 8\,\text{channels} \; \times \; 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Working with the Oscilloscope (Tektronix MSO58LP)

The Great, But Sometimes-Also-Not-So-Great News

- Oscilloscope can record particles into an internal buffer that is surprisingly large
- ⇒ we can collect 10k to 30k particles before we have to read out the data from the scope!
- Fully programmable via a protocol called VISA
- **But:** Reading out the buffer particle by particle, signal by signal ((⇒ Waveform by waveform) is **SLOOOOOOOOW**!!
- $\sim 0.3s$ per channel and per waveform
  $\Rightarrow \quad 30 \cdot 10 \ \times \ 8\,\text{channels} \ \times \ 0.3s$ is about 20 hours!
- **Does not work for us!**
- Good news again: There is a way to readout all $30k$ particles for each channel into one big file, that is a lot faster (approx 20 min for all 8 channels with 30k particles)
- But what about the event building? What about the synchronisation with the "main" DAQ?

Now: full picture of the trigger configuration, understandable!



Next Steps

- Continue calibration of our detectors (Scintillators, DWC)
- Timing in of signals
- Integrate one detector of Sakura Particles and Mavericks online
- Check data rates and trigger concepts
- HAVE FUN!!!

Thank you! - Any Questions?