Update on the double injection threshold scan

Thanks to Maurice Garcia-Sciveres, Timon Heim and Magne Lauritzen

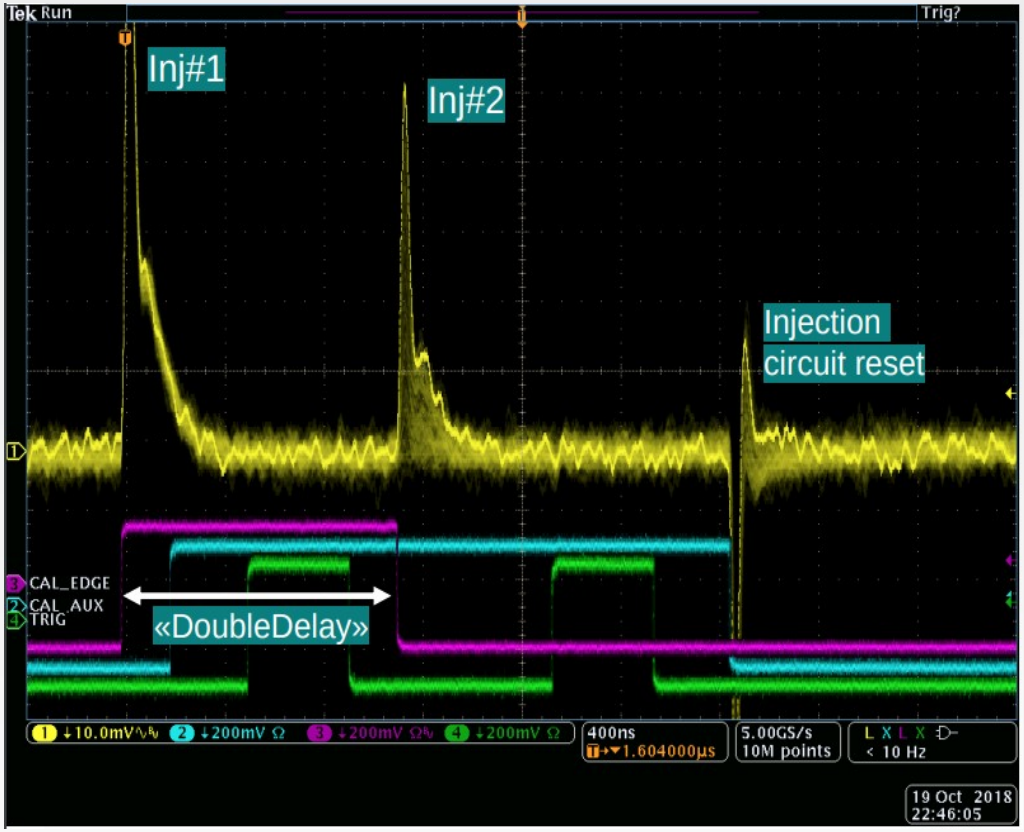
Presented by Simon K. Huiberts

ATLAS EXPERIMENT

UNIVERSITY OF BERGEN

Introduction

- The purpose of the **double injection scan** is to study the behaviour of the Front-end (FE) during charge injections and data readout
- I.e. to study how the **readout signal** is **affected** by a **preceding injection** -> simulate charge deposition from particles hitting close in time
- **A double injection scan** injects **two consecutive charge injections** into each pixel
  - Done via Cal commands which control the capacitor injections for a selected pixel
- **How it's done:**
  - First injecting a constant charge into the selected pixel (Inj#1)
  - Wait a set period (DoubleDelay)
  - Injecting a second charge of varying magnitude into the same pixel (Inj#2)

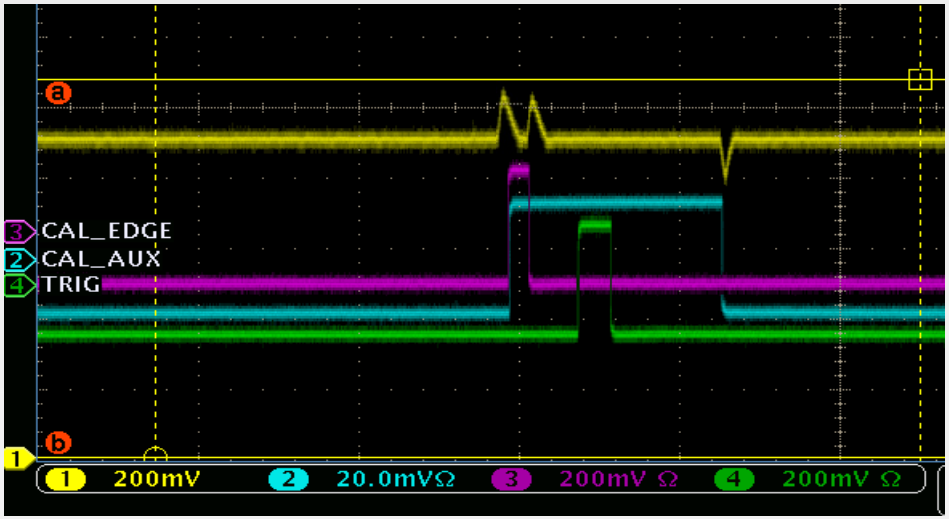


Double injection scan taken by an oscilloscope. Figure by Magne Lauritzen

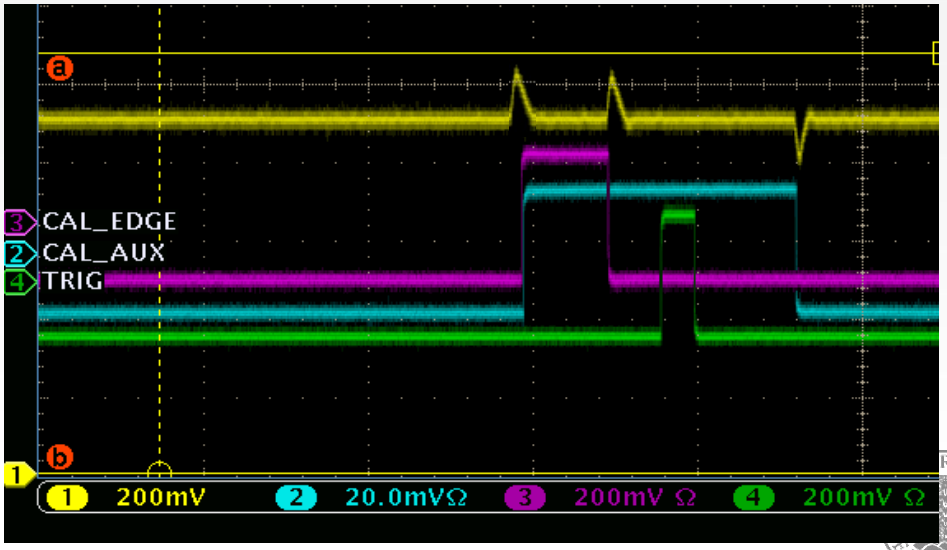
Double delay values

- **Double delay** parameter in the scan is used to set the **period** between the **first injection** and the **second injection**
- This is done by changing the **duration** of the **CAL_edge** control signal
  - Top figure shows a scan with double delay value = 5 [BX] and the corresponding small CAL_edge duration
  - Bottom figure shows a scan with double delay value = 16 [BX] which increases the CAL_edge duration
- For double delay values **< 16** only one Cal command is used which allows for **quarter values** to be used (14, 14.25, 14.75 [BX] etc.)
- For double delay values **> 16** two cal commands are used and only **whole numbers** can be set (16, 17, 18 [BX] etc.)

Low value of the double delay parameter

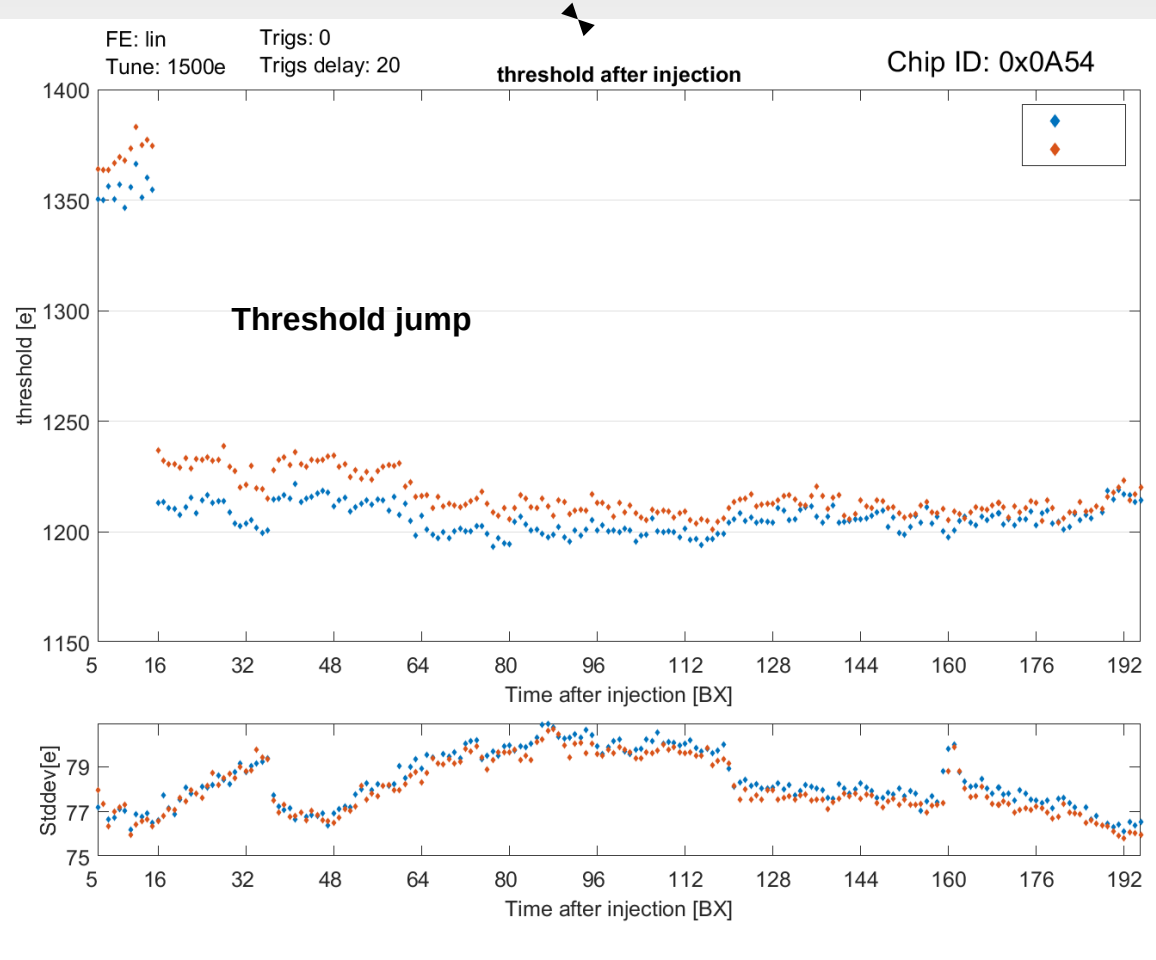


Higher value of the double delay parameter



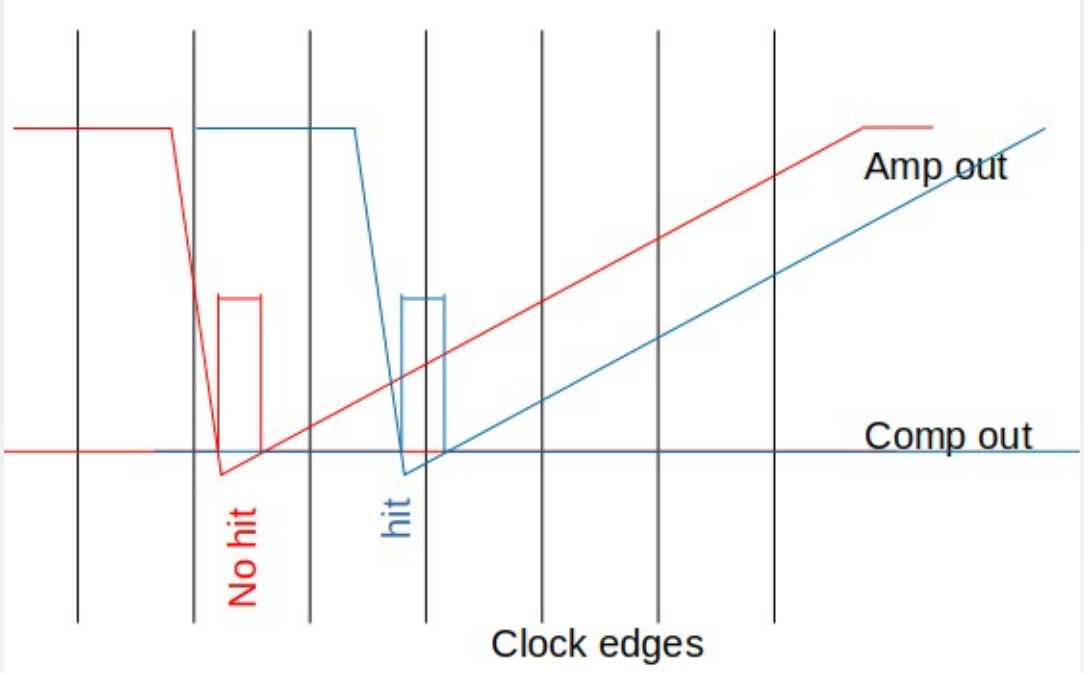
Threshold mean vs. double delay (Linear FE)

- Blue points: Baseline (One have one injection)
- Red points: **Inject 2000e** (Two consecutive injections)
- X-axis shows the time between the two injections (double delay)
- **Observed issue**: High threshold jump going from double delay value of **15 [BX]** to **16 [BX]**
  - Two different injection commands are used here
  - As the **single Cal command** allowed for **quarters bunch crossing delays** to be used
  - Suspected that this threshold jump came from the injected **signals being out of phase** relativity to the clock edge



Phase shift and Comparator output

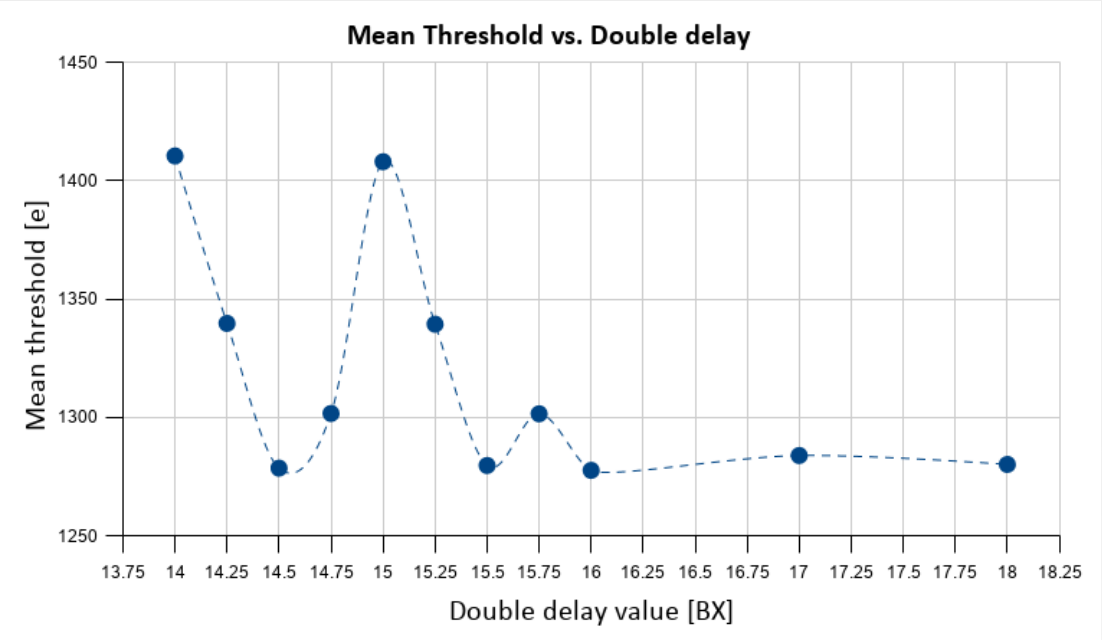
- In RD53A, a signal is only **recorded** as a hit if the **output** of the **comparator** is **high during** a rising **clock edge**
- Comparator is high when the injected signal is above the analogue threshold
- Clock edge has a period of one bunch crossing (BX = 25 ns units)
- This means that depending on the phase of the injection the **output pulse** of the comparator **may or may not** be **recorded**
- E.g. the red line in figure shows an injection that reaches above threshold but as the injection is not in the same phase as the clock edge, the hit is not recorded
- While injecting the same amount of charge in the correct phase (blue) will result in a recorded hit as the comparator output matches the rising clock edge



Comparator output for injections at different phases. Figure by Maurice Garcia-Sciveres

Threshold mean vs. double delay quarter values

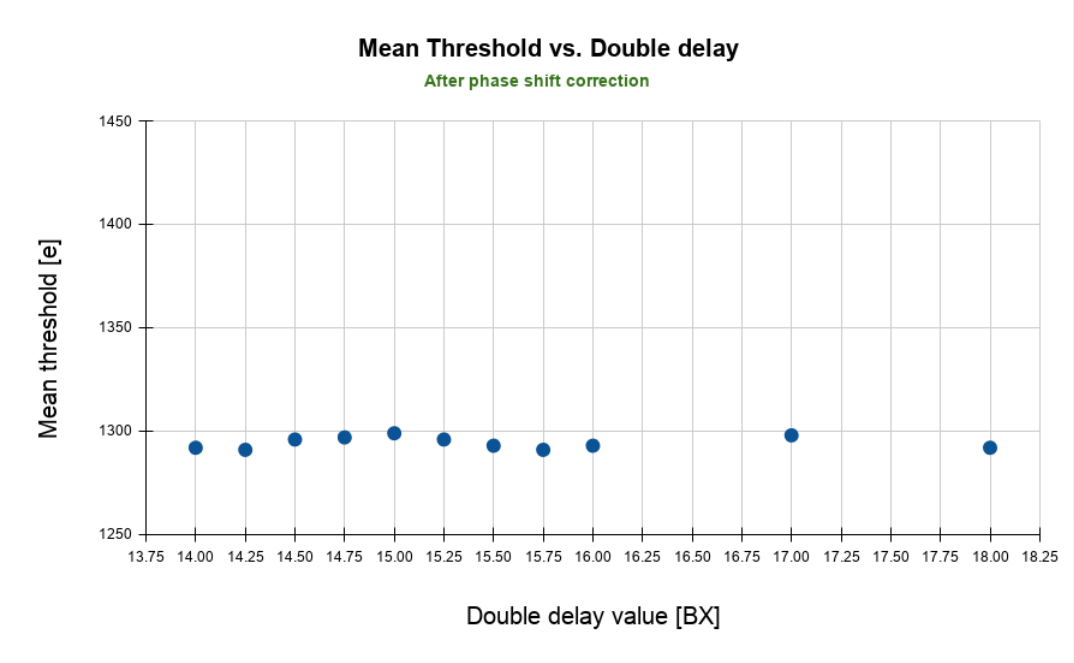
- Blue points: is the mean of the threshold distribution on the linear FE
- X-axis shows the double delay values in 0.25 [BX] steps
- Only **quarters values** can be set for double delay **< 16** [BX]
- Observe that the mean of the **threshold** distribution **increases for** ¾ of the points **when < 16** (15, 15.25 and 15.75 etc.)
- However, for **half values** of the double delay (14.5 and 15.5) the mean of the **threshold is correct** as its roughly the same for the double delay > 16 points
  - **Conclusion** is that the injection is in the **correct phase** for the **14.5** and **15.5 values**



Threshold mean vs. double delay (after phase shift correction)

- After fix: The mean of the threshold distribution is now roughly the same for all double delay values

- **Conclusion is that the injections are now in the same correct phase**



Conclusion

- A double injection scan sends out two consecutive charge injections into a single pixel
- Observed an issue where the injections were sent out of phase w.r.t the clock edge resulting in an artificially high measured threshold
- This has now been fixed by implementing a phase shift such that all injections happen in the same phase
- Future work is to study the differential and linear FE with the working double delay scan (present in three weeks)

Thank you for your attention!

ATLAS EXPERIMENT

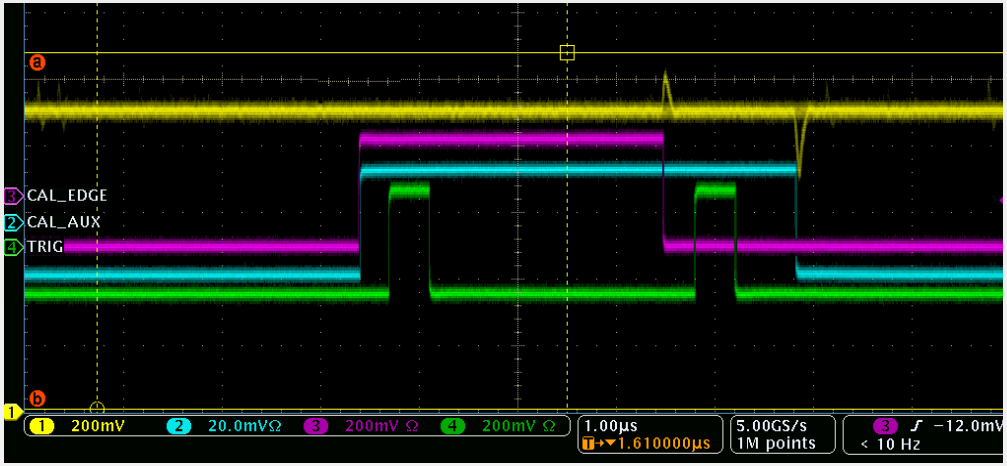
UNIVERSITY OF BERGEN

Backup

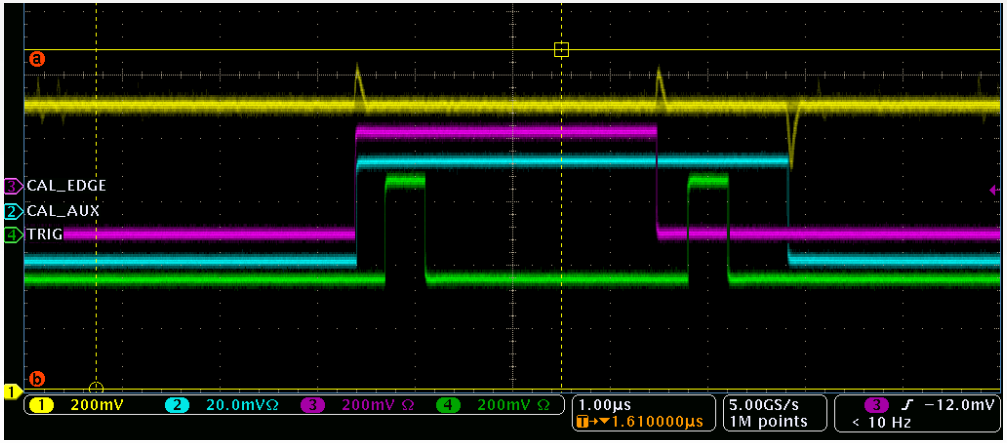
Method

- For each allowable value of the double delay, perform:
  - A double injection threshold scan with **#Inj1 set to 0e** (Upper figure)
    - Gives a **baseline** use to compare the effect of the next scan:
  - A double injection threshold scan with **#Inj1 = 2000e** (Bottom figure)
    - Injection 1 crosses the pixel threshold
- **Probe** the effect that the #Inj1 = 2000e has on the threshold distribution

Baseline (Only have the second charge of varying magnitude)

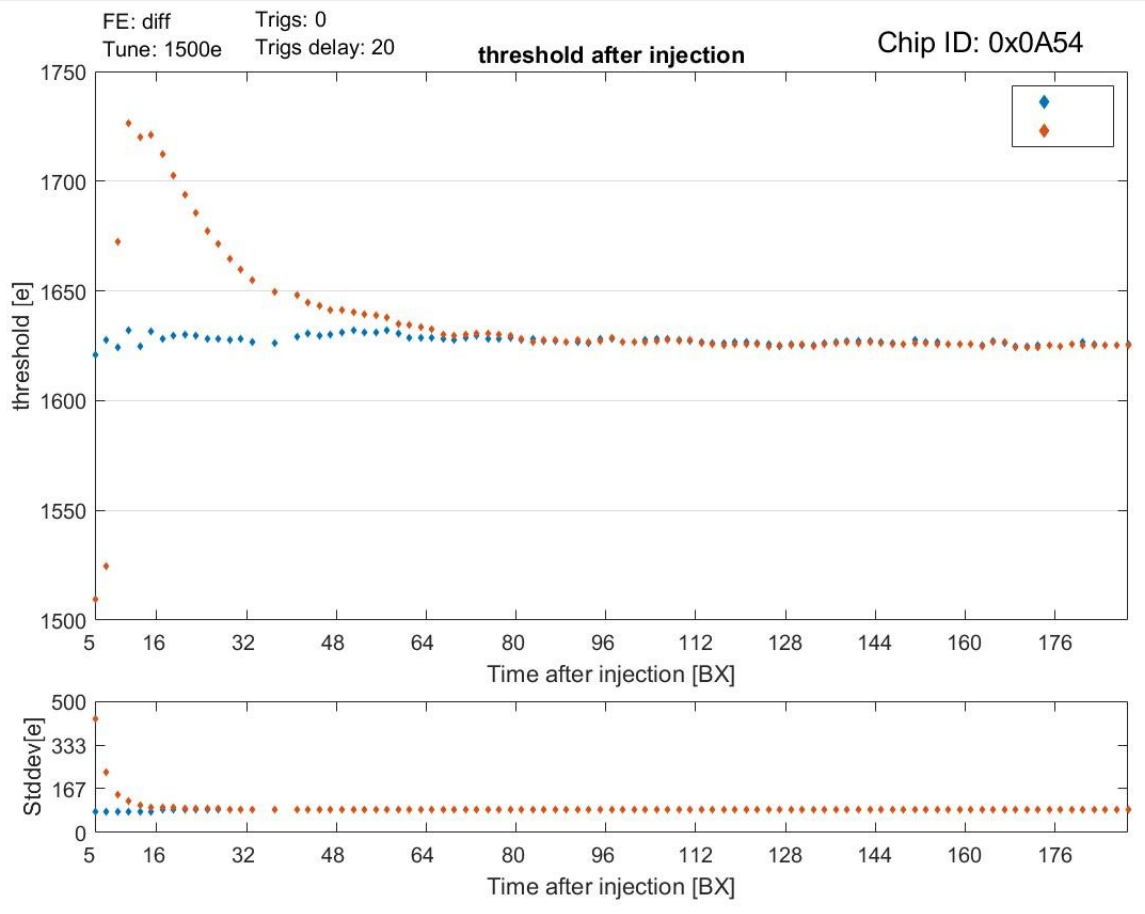


Inject 2000e and then inject a second charge of varying magnitude



Threshold mean vs. Double delay (Differential FE)

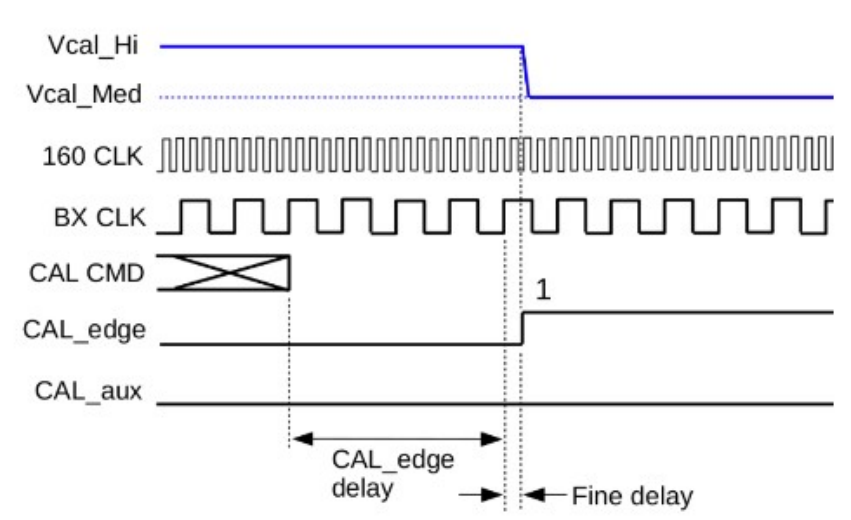
- Diff FE: Tuned to 1500e
- Blue points: **Baseline** (Only have the second charge of varying magnitude)
- Red points: I**nject 2000e** and then inject a second charge of varying magnitude
- X-axis shows the time between the two injections (double delay)
- When the double delay is small - > the mean of threshold increases when having a first injection of 2000e (Red points)
- Most likely caused by the disturbance of the first analog injection or the readout



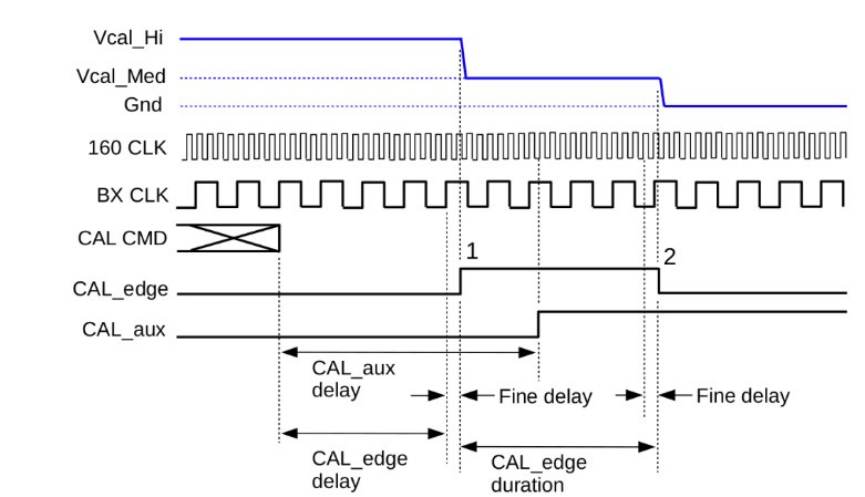
Cal command

- The **Cal command controls** the generation of two internal signals **CAL_edge** and **CAL_aux**
- Injecting charge into the pixel in done when these internal signals are changed
  - Top figure: The CAL command changes CAL_edge from **low to high**
  - Inject charge from **Vcal_Hi to Vcal_Med**
- CAL_edge can either be set to a **single step mode** (top figure) or a **pulse mode** (bottom figure)
  - In **step mode** CAL_edge it will **stay up**
  - In **pulse mod**e it will **stay up** only for a given time and **then go low** again
    - Inject **twice** with only one **CAL command**

Step mode with **one** injection



Pulse mode with **two** injections



Scan sequence in original code (now fixed)

- Due to a bug in the RD53A chip, the cal edge would go low (if high) after receiving a CAL CMD.
- This would cause a second injection to interfere with the threshold scan injection as show in the figure

UNIVERSITY OF BERGEN