xPIN Diode + Sample and Hold + NI-6002 Manual

August 13, 2020

Contents

1 Introduction

This manual describes the usage of the PIN diode + Sample and Hold () + NI-6002 setup in ELI Beamlines, which is used for shot to shot x-ray photon pulse detection at the PXS experiment in E1. This section will provide a brief overview of the theory around it. In the next section all the separate parts will be described in more detail and instructions on how to connect them together will follow.

The advantages of the PIN diode in those scenarios is that it has a sufficient time resolution and that it is mainly sensitive to photons up to a certain energy. That reduces the overall photon count which in turn increases the accuracy. To get the full experimental setup we also need an oscilloscope and a trigger signal synchronised with the pulses. To use the S&H we also need a way to delay the trigger by an adjustable amount, for that we use a signal generator in the "triggered" mode. The last thing we need is the NI-6002 device and a computer with the software for it instaled.

In short a PIN diode is a P-N junction but with a section of undoped (Intrinsic) semiconductor between the P and N regions. The diode is reverse biased and when a photon hits the intrinsic zone it knocks an electron out of its place. The electron-hole pair that was created is then split by the voltage over the region and creates a current. This current however is usually very small and needs to be amplified.

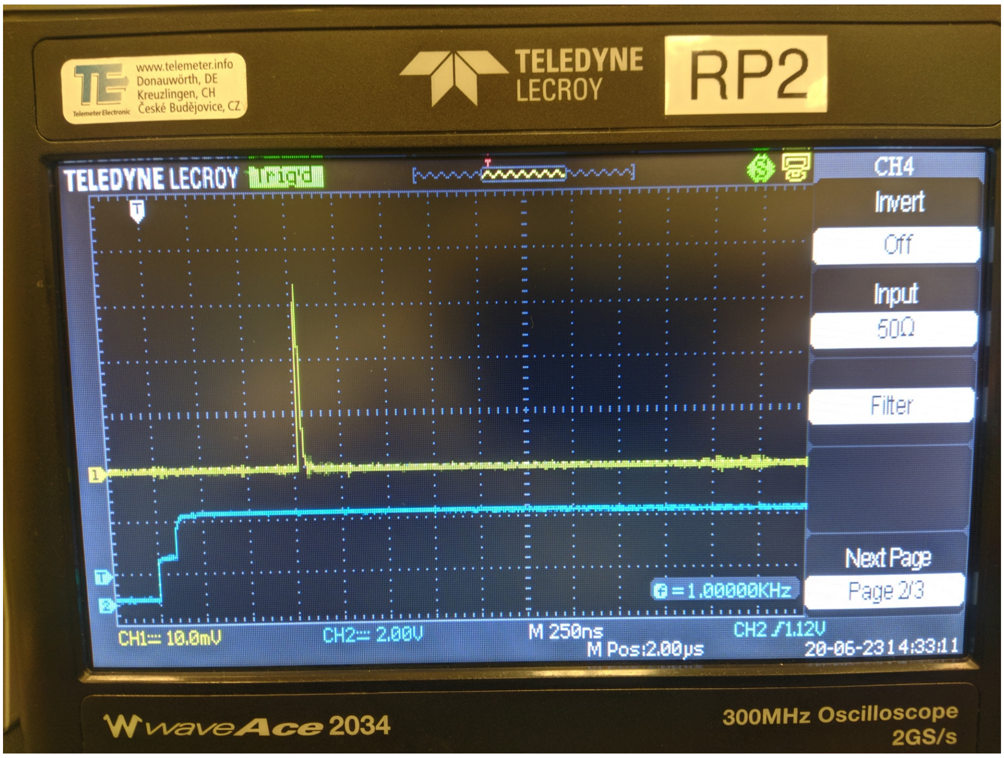


Figure 1: The shape of the PIN output signal (yellow line) viewed by an oscilloscope

It was mentioned that the time resolution of the PIN diode is very good, however that means that the measured signal can also get very short. In our lab the photon pulse is around 20 fs and the pulse that comes from the diode can be about 70 ns. In order to estimate the number of photons per pulse we need to get the area of the pulse. Since the shape of the measured pulse (shown in fig. 1) is quite constant, we decided that instead of using an expensive data acquisition device that could capture the nanosecond range pulse accurately we would use the S&H. The S&H is a device which captures the signal it gets and holds it for a specified amount of time. This lets us get the height of the pulse peak and hold it until the next pulse, which makes it much easier to capture. Specifically in our lab the pulse frequency is 1 kHz, that means that we can hold the peak hight for half a microsecond, this way even a 4 kHz data acquisition device should provide accurate data. This signal is then captured by the NI-6002 and analysed by the computer connected to it.

2 Parts

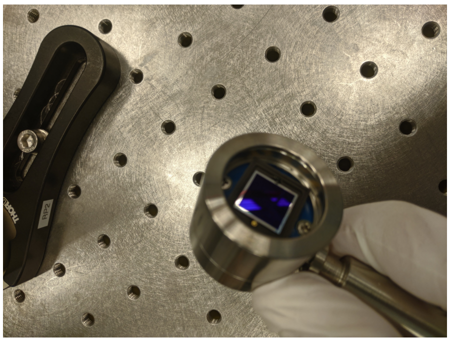
This parts introduces the specific parts we are using.

2.1 PIN Diode

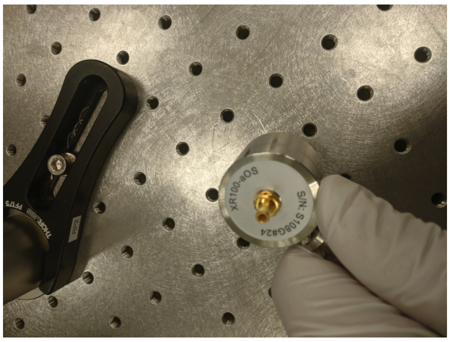
We are using an xPIN diode from RIGAKU, the product page on their website can be found here [1]. ~~However the bias box shown there is different from the one we are currently using~~. Its spectral range is 5 keV (T = 10%) – 27 keV (above 10% of peak sensitivity). The bias box ~~we have~~ provides a reverse bias of 50 V and amplifies the signal from the diode.

The device has 3 parts: the PIN diode itself, the bias box and a cable connecting the two (fig. 2). The bias box has two more connections: a micro USB power cable and a BNC connection for the output signal. For the micro USB power source, we used a small battery as it provides a more stable output then other sources we had available. A picture of the diode setup can be seen in fig. 3.

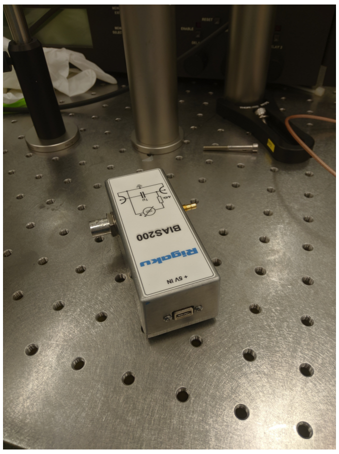
The PIN diode is simple to setup as it has no adjustable settings, just connect all the parts together and get the output.



(a) The PIN diode, front view without VIS protection



(b) The PIN diode, back view



(c) The bias box

(d) The cable to connect them

Figure 2: PIN diode parts

[1] https://www.rigaku.com/products/detectors/xpin

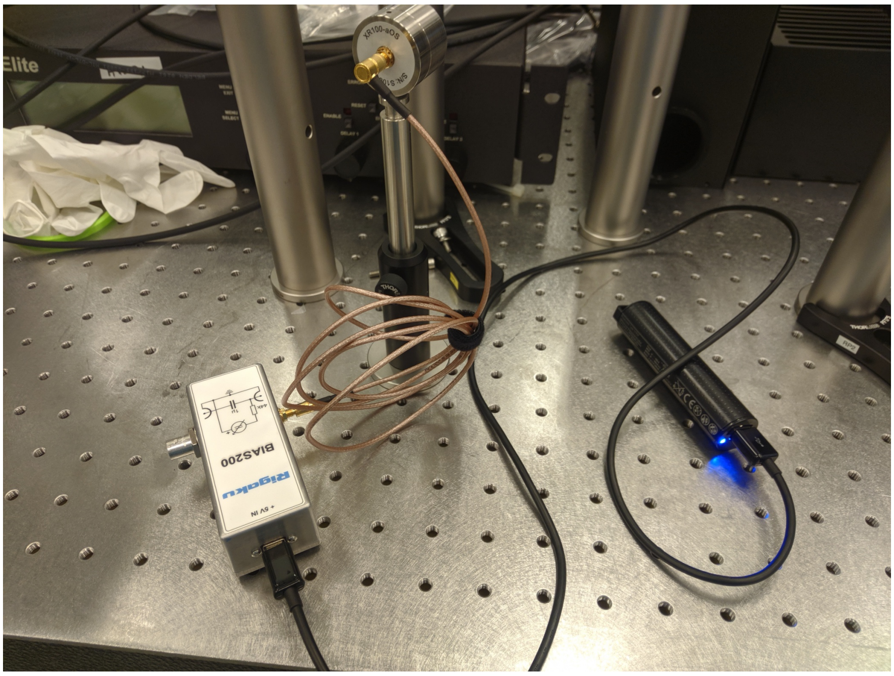
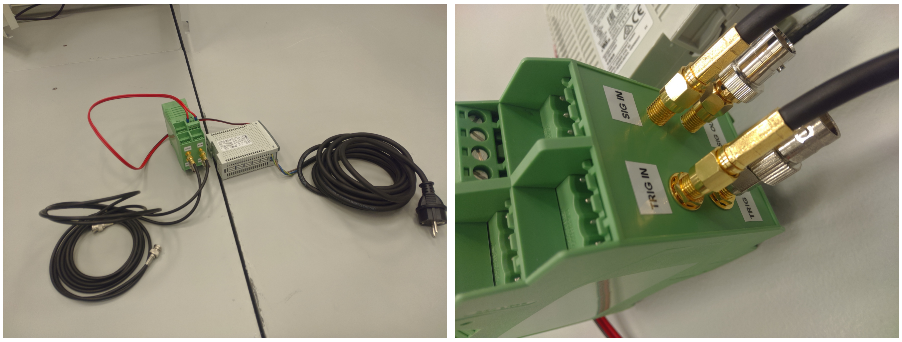


Figure 3: The PIN diode setup

2.2 Sample and Hold

The device referred to when written S&H is a device built in ELI (shown in fig. 4) which contains a Sample and Hold amplifier (its schematics can be seen in appendix A). The Sample and Hold amplifier is made by DATEL and the product page can be found here[2].

The device consists of two parts, the gray box contains the power supply and the green one the Sample and Hold amplifier. The green box has 4 ports: TRIG IN, TRIG OUT, SIG IN and SIG OUT. It works as follows: When the voltage on TRIG IN crosses a certain threshold, the device "saves" the value of the voltage on SIG IN and then continues to output that voltage on SIG OUT until the voltage in TRIG IN drops back down. TRIG OUT just outputs whatever came in on TRIG IN and is not essential for the function. An image from the S&H in action is in fig. 5.



(a) Sample and Hold (b) Detail of the ports

Figure 4: The Sample and Hold device

[2]http://catalog.datel.com/item/data-acquisition-product-line/sample-and-hold-amplifiers/shm-43mc

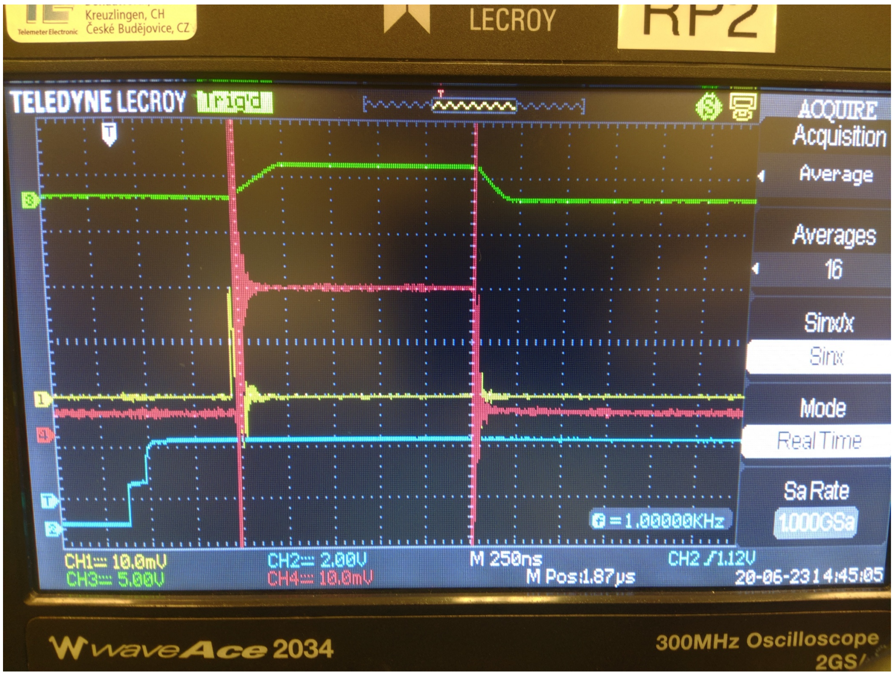
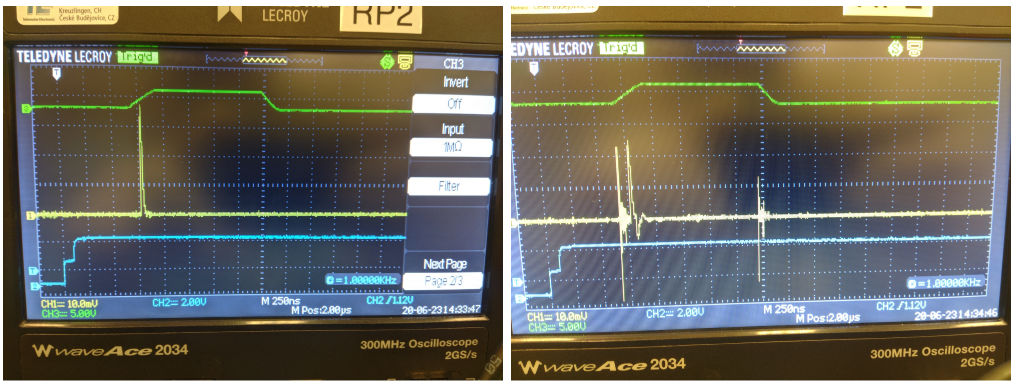


Figure 5: Example of the S&H in action. Green line is TRIG OUT, yellow line is SIG IN, red line is SIG OUT and the blue line is just used as a trigger for the oscilloscope

~~There's some more information that needs to be mentioned however.~~ ~~The first thing is that~~ the S&H needs the TRIG IN signal to reach at least ~~about~~ 5 V (and it's safe to go up to 9) otherwise it won't trigger. This is the threshold mentioned before, however it does not actually trigger when TRIG IN reaches the threshold, it triggers when it reaches a smaller value (this is best illustrated in fig. 5 where you can see the red line triggering when the green line has barely just started going up).

Another thing is that the S&H changes the input signal. By this we don't mean that the signal on SIG OUT is distorted, but that the signal connected to SIG IN gets distorted by connecting the S&H to it. When tested using another signal generator instead of the xPIN, the measured voltage was a linear function of the generated voltage with coefficient a being about 2 and b being -5 mV. This can be seen in fig. 6 but this is the xPIN again so the function isn't quite the same. The S&H also adds noise to the signal (SIG IN again), especially around the time the trigger goes up (again visible in fig. 6).



(a) PIN diode output when the S&H is disconnected.

(b) PIN diode output when the S&H is connected.

Figure 6: This figure shows how the S&H distorts its input signal. The connections are again the same as in fig. 5.

The S&H does also increases the signal on SIG OUT by a bit compared to the voltage on SIG IN. This is visible in fig. 7 and it seemed to be about 4 - 5 mV during our testing.

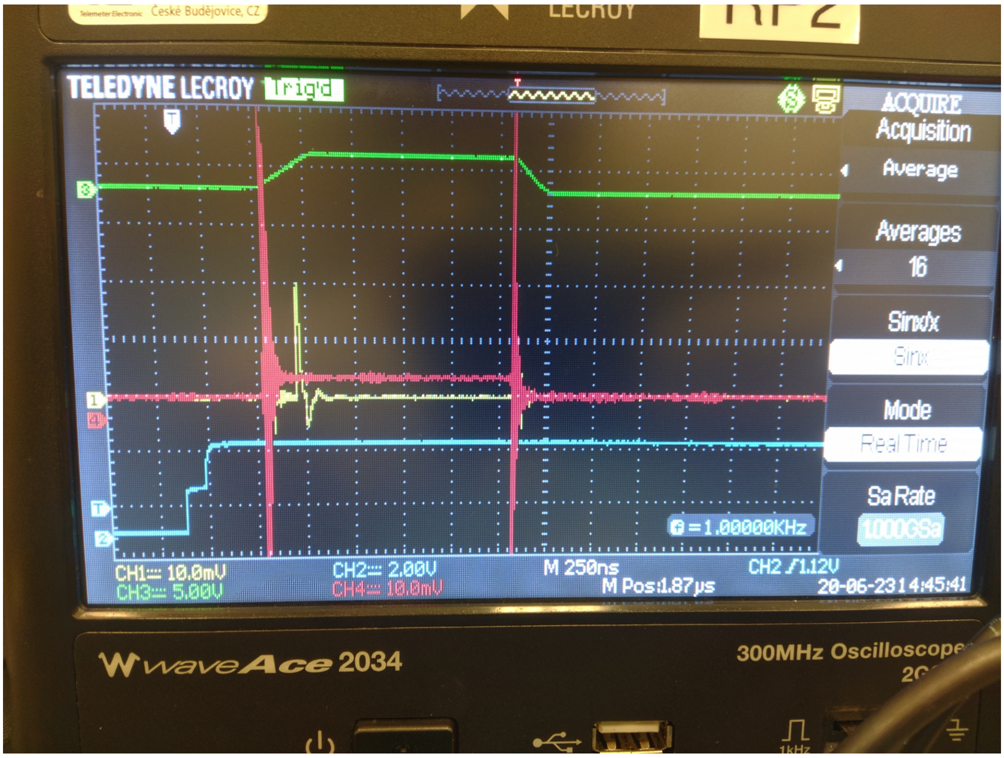


Figure 7: The connections are the same as in fig. 5 but this time the S&H trigger is moved a bit in front of the peak. It is possible to see that the yellow line dips a bit just as the S&H triggers, this is just an artifact of the S&H it is not a part of the actual signal. It can also be seen that the red line is about 4 mV above the yellow line. This is another artifact of the S&H.

Finally while the S&H can hold the signal, the voltage it outputs slowly decays. It can be seen in figure fig. 8 where it decays by about 4 mV in just under 500 µs. ~~This can be very relevant when working with a very low voltage! Because if the voltage is low,~~ the decay can go below the baseline of the signal (fig. 9) and this may make data analysis of the data more difficult (especially if the decay ends on the baseline). In particular, the provided software will not be able to analyse such data.

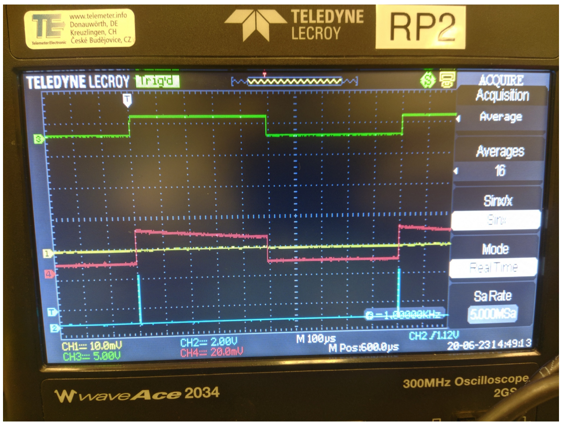


Figure 8: The mentioned decay of the SIG OUT voltage during the trigger. Connections are the same as in fig. 5.

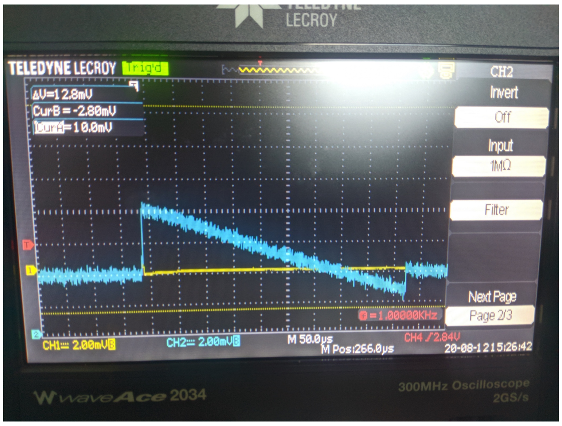


Figure 9: An image of how the held voltage can decay under the baseline, in this image the xPIN output is connected to yellow and the SIG OUT is connected to blue.

2.3 NI-6002

This is a data acquisition device made by National Instruments, the product page can be found here[3]. This device can do a variety of things but the only thing we need it for is the analog input port (it can capture at frequencies up to 50 kHz). We are using it in the single-ended mode, more specifically we have the S&H output connected to the AI0 and AI GND ports (AI0 is important as otherwise the software won't work with it). Along with this you will need a computer with the custom software installed on it (along with the nidaqmx drivers).

2.4 Notes on other equipment

To use the S&H it is necessary to get a trigger signal that is synchronised with the photon pulses and both it's delay and width are adjustable. For that we use a signal generator in the "triggered" and "pulse delay" modes. The one we use can be seen in fig. 10

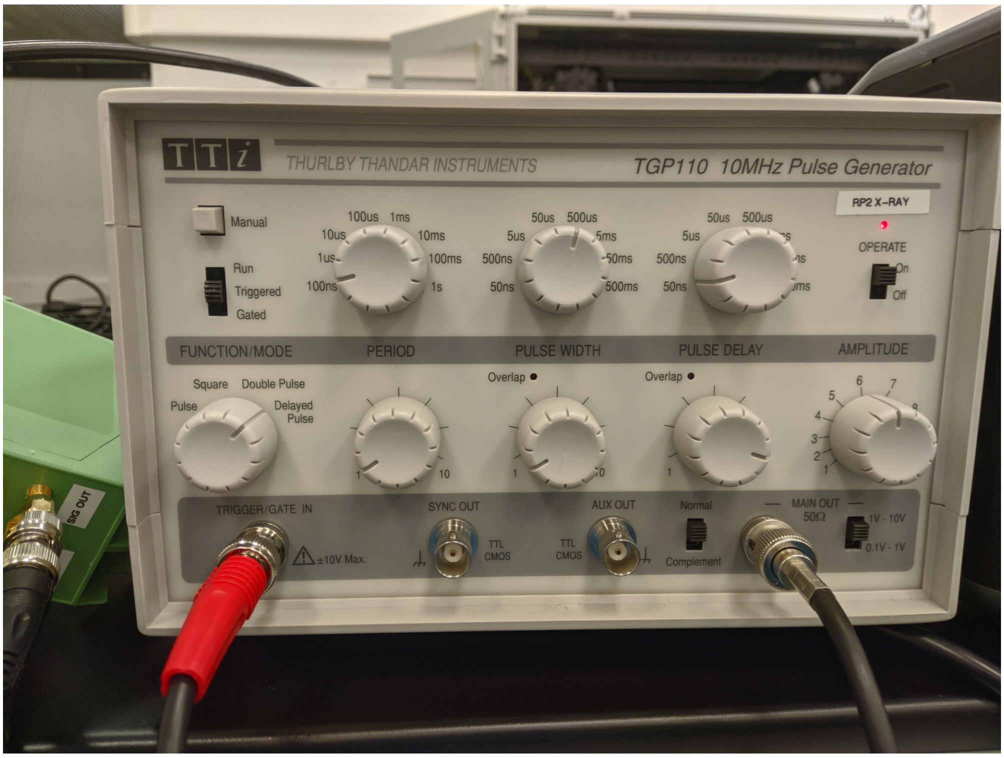


Figure 10: The signal generator we use as the S&H trigger.

[3]https://www.ni.com/cs-cz/support/model.usb-6002.html

3 Connecting it together

After you secure the photon source and the trigger signal, set up the diode and check it's output using an oscilloscope (using the trigger signal as a trigger for the oscilloscope). Once you get stable data there you can start setting up the S&H.

First get the signal generator (SG) and set it in "triggered" mode, use the ~~photon~~ trigger signal as the SG trigger. Also set the SG to "pulse delay" or equivalent mode, set the delay to 0 and the pulse width to something reasonable. Once again you may check on the oscilloscope if it works as expected. Then connect the SG output to TRIG IN on the S&H and if your oscilloscope has enough ports you may also connect the TRIG OUT to it (which might help get the delay right). Then connect the PIN output to SIG IN on the S&H and once again you may use a BNC splitter to also get the raw PIN output on the oscilloscope. The last step is to connect the SIG OUT to the oscilloscope.

Then you need to make some adjustments. Firstly, in section 2.3 it was mentioned how the S&H adds a fixed amount of voltage during the trigger. If you are observing the data directly on the oscilloscope you may want to adjust for that now. Just move the delay somewhere where the input is on its zero and then adjust the line height of the SIG OUT line (on the oscilloscope) so that the line during the trigger is at some baseline. Later you can get the voltage with respect to this baseline which will give you the correct data. After this step the oscilloscope should look like in fig. 11.

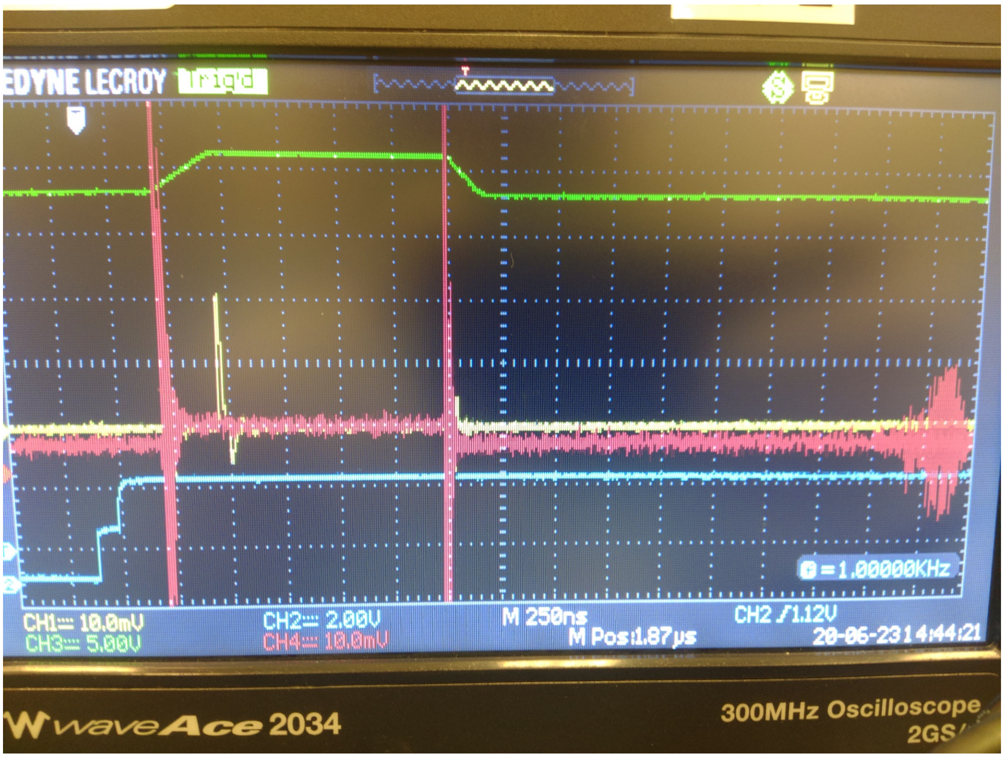


Figure 11: This shows how the oscilloscope should look after adjusting the SIG OUT line height. Once again connections are the same as in fig. 5.

Now you need to adjust the trigger delay. Simply move the pulse delay so that the S&H trigger aligns with the PIN output peak. Keep in mind that the S&H delays the pulse slightly so you will need the trigger to happen slightly after the peak. Once it has been set correctly the voltage during the trigger should rise to be the same as the voltage of the peak on the PIN.

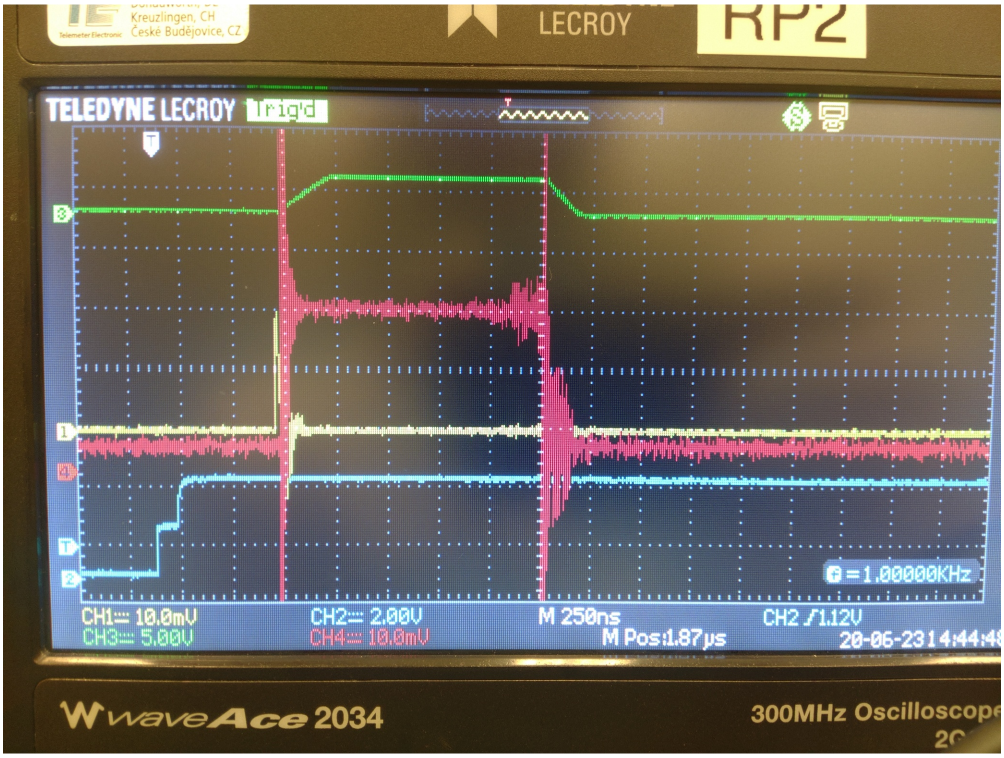
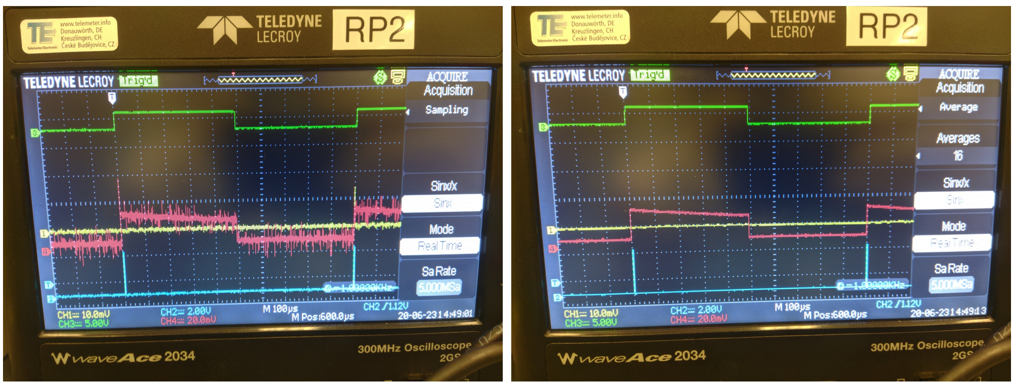


Figure 12: This is how the oscilloscope looks after the delay has been aligned. Once again connections are the same as in fig. 5.

Now the xPIN + S&H setup is just about complete, the only thing remaining is to adjust the pulse width and capture the data. Unless the xPIN voltage is really low it is probably best to make the pulse width half the photon pulse period as that way it will be the easiest for the acquisition device to capture. However if the S&H decay causes the "peak" to cross the baseline (or come close to it) you should make the pulse width shorter. This is because the provided sofware can only analyse the data correctly if the "peak" is a peak. Also the drop after the peak needs to be at least a certain value (depends on the processing parameter called Edge Detection Threshold).

In fig. 13 you can see how the data looks in the big picture. As mentioned you can see that the voltage during the trigger of the S&H decays slowly but still significantly. Now you need to connect the S&H SIG OUT to the NI-6002 port AI0 and AI GND. Then just connect the device to a the computer and start up the software.



(a) Oscilloscope set on sampling to show the noise.

(b) Oscilloscope on averaging shows the decay nicely.

Figure 13: Here you can see the oscilloscope as the pulse delay has been adjusted. Once again connections are the same as in fig. 5.

4 Software

Instructions on how to install and run the software are in the README file. The software is mostly web based and written in Python. Once you start the server you can visit the web page on port 8050 (you type '<ip>:8050' in the url bar in the browser where <ip> is the ip address of the computer). You should get a website with 4 tabs, the two middle ones may be disabled and a warning may pop up on load. The rightmost tab is the help tab, there you can find some information on how to use the software interface. The technical documentation of the code itself is distributed with the code itself and here there will be some notes about the data analysis itself.

4.1 Data Analysis Description

Once the data acquisition starts a thread is created which continuously gathers data from the NI-6002 as fast as possible. The data here is the output of the S&H but it looks differently then it does in the oscilloscope, visible in fig. 14. Here a superposition with a sine wave is clearly visible (or maybe a different function, depending on the lab, in our case the wave is a 50 Hz sine so it's quite likely it's due to the plugs).

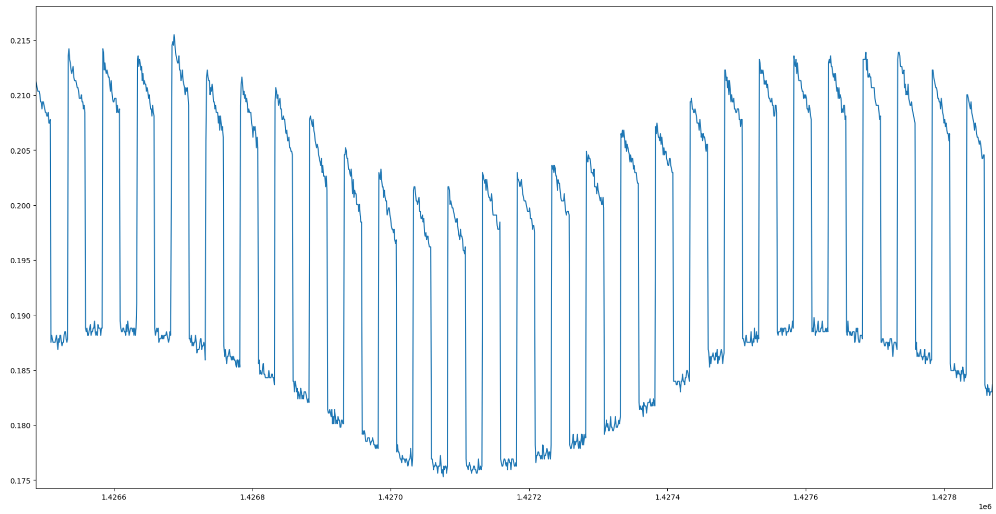


Figure 14: Raw data captured by the NI-6002.

~~First of all~~ when section is written it refers to a group of datapoints between two sharp edges, it corresponds to when the TRIG IN signal on the S&H is either on or off, but not currently changing. Sections alternate between up and down sections, up sections also having a slope. The software always remembers data about the last two sections.

When the software gets a new datapoint, first of all it calculates a value called 'diff' in the code, this is the difference between the new datapoint and the last one. Then it is checked if the absolute value of 'diff' exceeds a parameter caled Edgde Detection Threshold (EDT).

If so, then it checks if 'diff' is positive or negative. If it is negative, that means that that an up section must have just ended and a down section should have been before it. To verify that it checks whether the average of the last section is greater than the average of the previous last section. If that is the case then it proceeds to get a voltage difference between the two sections and saves it.

It gets the voltage difference by first removing the outliers from each section (this is important for when some of the datapoints land on the edge between sections). Then it gets the averages of each section and also the average slope of the up section. It then moves the up section average by the average slope to get the actual peak. Then it gets the difference between the peak and the average of the down section.

Now going back, if the averages of the two last sections do not match up, it prints a warning message to let the user know that there must have been some anomaly, this feedback can be used to adjust various parameters. Going another step back, if 'diff' is positive, nothing is done.

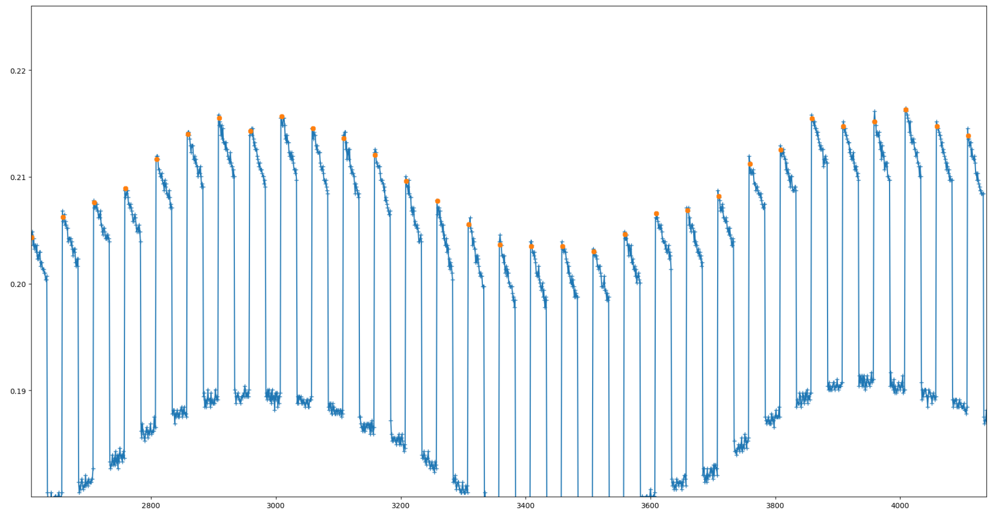


Figure 15: Showcase of the precision of peak estimation.

However there is another condition, if the absolute value of 'diff' is bigger than Edge Detection Threshold and there are some values gathered in the last section, then it forgets the current previous last section and moves the current last section in its place. It also resets the values for the last section.

Lastly only if the absolute value of 'diff' is lower than EDT, it adds the datapoint to the current section. This is also important when datapoints land on the edge between sections.

In case a programmer wants to look at the code instead, all of this is in the 'handle_processing' method of 'DataAnalyser'

4.2 Setting the correct Edge Detection Threshold

~~This is probably the most unpredictable and complicated step~~. The most important tools for adjusting EDT is the graphing option on the host computer (you can't do this through the web interface) and the peak voltages per second (counts per second) in the live graph in the web interface. ~~I should also mention that~~ while the raw data graph is in focus, if you press p, you can stop the graph at the current image (and resume pressing p again). If the counts per second don't add up with the expected pulse frequency, something is wrong. You can start by observing the approximate spread of the raw data on the raw data graph, EDT can't be too close to that or random noise can be branded as an edge. If your peak voltages will be relatively high, this should be all you need, give a sufficient overestimate and all should be fine.

However if you need to detect very small voltages you run into issues as the EDT becomes comparable to your peak voltages. Firstly, one of the S&H artifacts actually helps us, the S&H always increases the measured voltage by a fixed amount during the trigger, this makes the signal more noticeable. Now, again you will need to look at the raw data graph and look at the edges there, you need EDT to be lower than those (you should mostly look at the edges from up to down sections as due to decay those are always smaller). Also from experience, the data you see is usually the normal data, every now and then the peak voltage is significantly more or less, so if you want to capture those you need EDT to be lower than it seems you would need.

~~This was mentioned in section 3 but it will gone over it again as it can be very important.~~ If your peak voltages are really small, you also need to account for the decay. The software can only analyse data that looks approximately like the data in the images here. It can happen that the held voltage decays below or on the baseline, you would easily notice this in the raw data graph. If this happens you can try to adjust the pulse delay so that the peak is shorter but doesn't decay under the baseline, that way the software can still work with it.

4.3 Data Analysis Artifacts

~~It was mentioned that~~ the captured data is a superposition of the data we want and an underlying function (in our case a sine wave). This underlying function does affect the processed data (peak voltages). Whenever the gradient of the function is positive, the peak voltages are increased by some magnitude and the other way around. This is caused by how the processing works and can be seen in fig. 16.

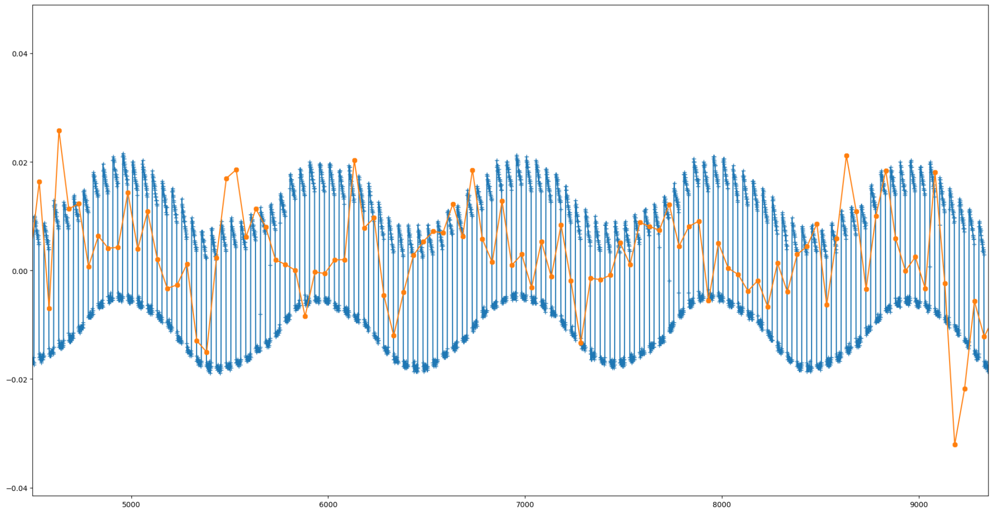


Figure 16: A comparison of raw data peak voltages, both normalized and adjusted so that they can be easily compared.

4.4 Working with saved data

When you run the data acquisition you can save the data, you can save in a csv, but it is highly recommended that you save in a hdf5 file as it can save much more data. When you save the data in a hdf5 file you can choose what you save, including the raw data on which you can do your own analysis. The debug markers are meant to be used by someone who is editing the code, currently they contain data on each detected spike (which is useful when trying to find errors in data analysis).

If you want to analyse the data in python I highly recommend reading up a bit on the h5py [4] library and to use matplotlib for the visualisation. If you choose to do that, you may find some useful tools in the 'util' file/submodule of the PINSoftware module, more information is in the technical documentation.

[4] https://docs.h5py.org/en/stable/

A S&H Schematic