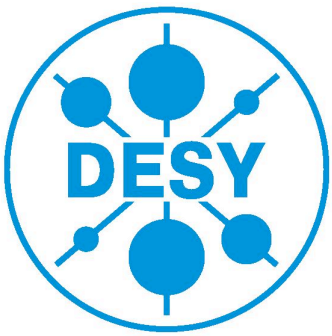


ATLAS EXPERIMENT

ATLAS FE-I4A Pixel Module as a Trigger Plane for the Beam Telescope

Tobias Bisanz, University of Göttingen, Germany

tobias.bisanz@stud.uni-goettingen.de

September 6, 2013

GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Abstract

The usage of the new ATLAS read-out front end FE-I4 as a triggering element in the EUDET pixel telescopes was investigated. The different steps, from tuning the individual sensors to the selection of a region of interest with the assisting DAQ hardware and control software, are described in this document. Furthermore, the HitOR signal inversion with different inverter models is described as well as the complete set-up including all the cabling. Finally some information will be given on the current status of this project as well as some ideas on what has to be improved.

Contents

1 Introduction

1.1 About this report

This report deals with the instrumentation of the DESY pixel telescopes with the FE-I4 as a trigger plane. After a brief description of the telescope and its uses as well as the used hardware, an introduction to the FE-I4 will be given.
The following sections are then a very detailed how-to on using the software and hardware for users who have never dealt with the FE-I4 and would like to use it as a trigger plane. Thus these sections are written as a step-by-step instruction, assisted by pictures and screenshots.

1.2 The Telescope

A beam telescope is a very useful tool for testing and determining properties of devices under test (DUTs). Such a DUT could be for instance a prototype part of a large high energy physics detector like ATLAS or CMS. A beam telescope at a testbeam offers the required environment to do various studies, for example the examination of the charge collection properties of parts of the ATLAS pixel detector before and after irradiation.

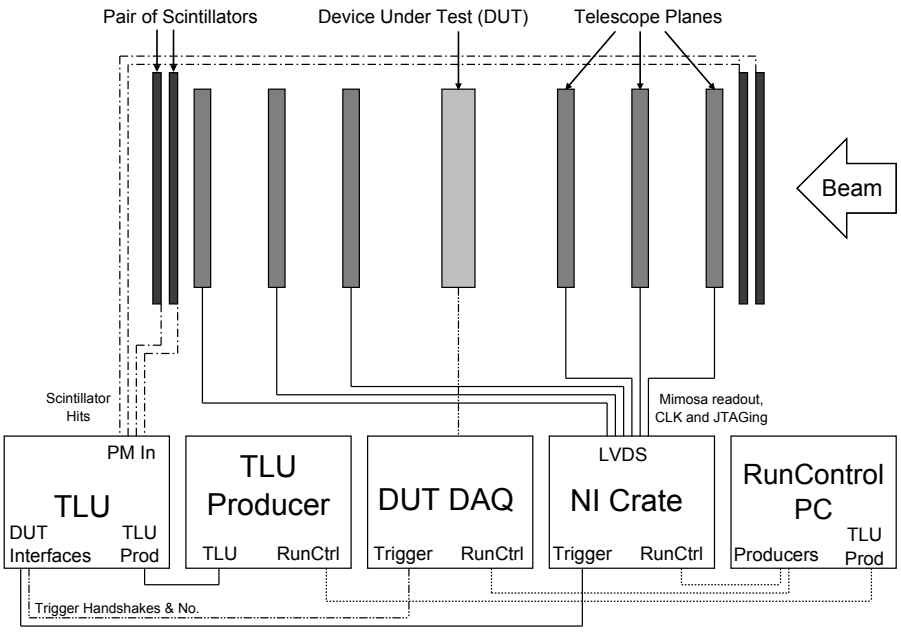


Figure 1: Schematic overview of the telescope.

The set-up of a beam telescope is schematically shown in Figure 1 and a picture of it is given in Figure 2.
The individual telescope planes house pixel detectors. In case of the EUDET pixel telescopes these are Mimosa26 sensors [1]. While in principle the number of detector

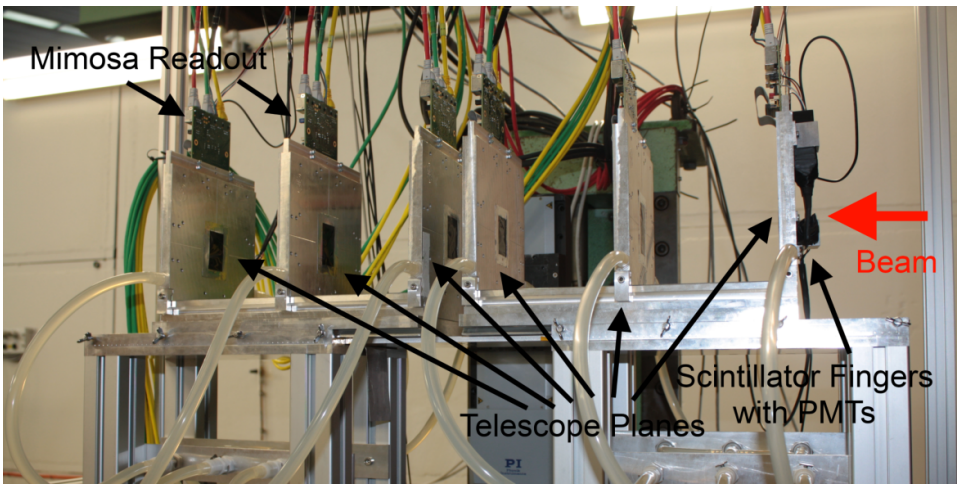


Figure 2: Picture of the telescope without DUT.

planes is arbitrary, usually six planes (three planes upstream as well as downstream from the DUT) are used. The more material there is in the beam's trajectory, the more multiple scattering occurs. Six planes are a good trade-off between this and the precision of track reconstruction.
Before the first and after the last telescope plane there is a pair of crossed scintillator fingers which are read out via photomultiplier tubes (PMTs). These scintillators are used to trigger a read-out of all the telescope sensors and the DUT. For this the signal from the PMTs is fed to the so-called Trigger Logic Unit (TLU) [2]. The TLU controls the read-out by assigning an event number to each triggered read-out. Usually for a read-out to be trigger, a coincidence of all four PMTs is required (in principle this requirement can be loosened by the user).
After collecting all this data, an offline analysis can be carried out. This is done with the aid of the EUTelescope software [3].

1.2.1 Triggering and the TLU

Triggering is a crucial component of data taking. Taking a photograph is a good analogy, you only want to take a picture when you actually have a nice scenery in your finder and press the trigger. When taking data with your telescope you only want to trigger a read-out when an electron (or any charged particle) passes through the detector. Therefore scintillators are placed in front and behind of the telescope. Since these scintillator fingers are, as the name suggests, elongated, a pair of crossed scintillators is used to limit the area on which you want to trigger (both scintillators have

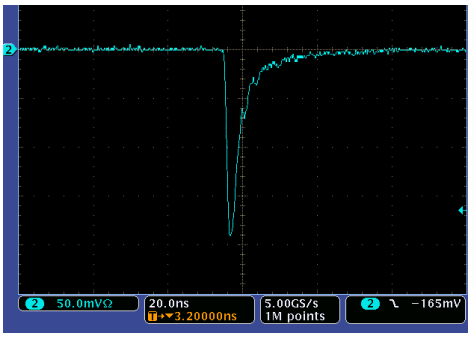


Figure 3: Signal of a typical PMT hit.

to trigger). The same set-up is used after the telescope, this allows the rejection of cosmics (or other noise) or any particles which exit the telescope before reaching the downstream scintillator pair.
The hardware to process the scintillator signals is the TLU, for this there are four discriminator stages in it. Figure 4 shows the schematic of the TLU discriminator stage.

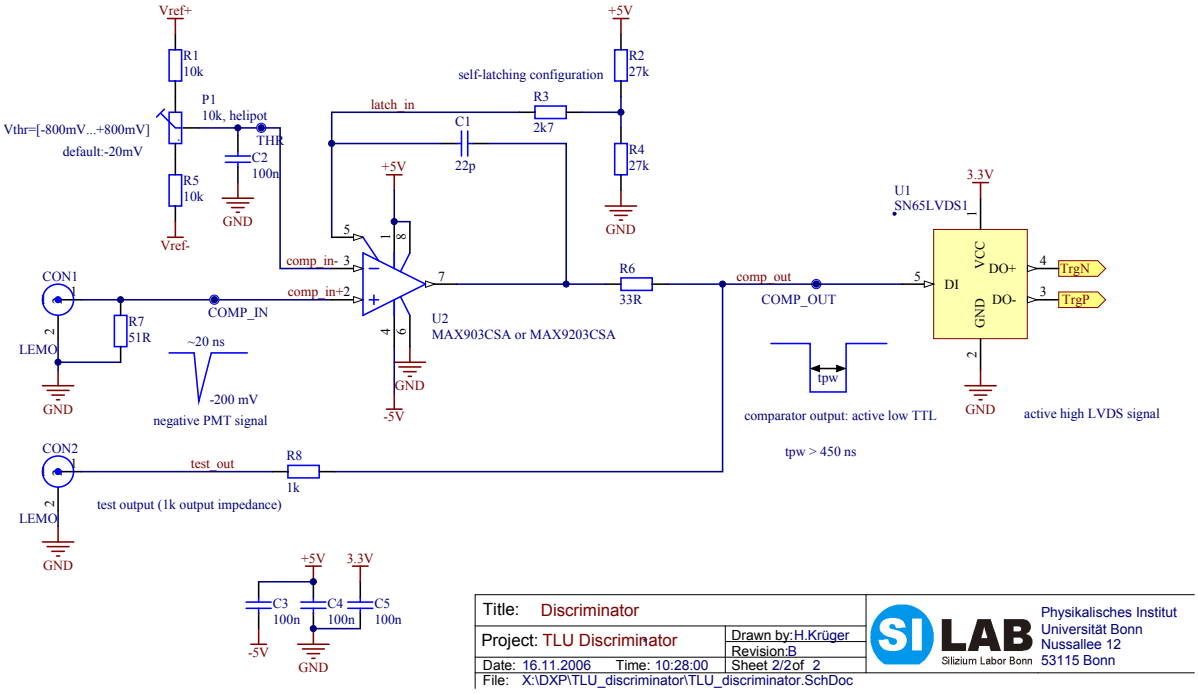


Figure 4: TLU discriminator board [4].

It is designed to process signals from PMTs, thus triggers on a falling slope. If it triggers on a positive pulse, the self-latching capability is lost. For further information on the TLU discriminator see [4].
Triggering on the scintillators is not the only task of the TLU, it also issues a handshake procedure between the TLU and the DAQ systems.

1.2.2 The Mimosa26 Sensor

The telescope planes are instrumented with Mimosa26 monolithic active pixel sensors (MAPS). They yield 576×1152 pixels of $18.4\,\mu$m pitch in both directions, having an intrinsic resolution of better than $4.0\,\mu$m. The size of the chip is $13.7 \times 21.5\,\mathrm{mm}^2$. Columns are read out in parallel, row by row with a read-out time of $115.2\,\mu$s. It features a correlated double sampling (CDS), i.e. the pixel's voltage after reset is additionally measured to the pixel's signal voltage. Subtracting the reset voltage then allows noise reduction. Each column is equipped with a discriminator.

1.3 The ATLAS FE-I4A

For the ongoing ATLAS upgrade a new Front-End (FE) has been developed for dealing with the high detector occupancy due to the LHC luminosity increase to its nominal value. Additionally a decrease in material and power consumption was set as a design goal. The FE-I4, as the new front-end is called, is used in the Insertable B-Layer (IBL) upgrade [5] and is a possible candidate for the outer layers of the ATLAS tracking detector after the Super-LHC upgrade.
Major changes compared to the old FE-I3 include the reduction of the pixel size from $50 \times 400\,\mu\text{m}^2$ to $50 \times 250\,\mu\text{m}^2$. This obviously improves the spatial resolution, but the main motivation for this reduction was the decrease in a pixel's hit rate due to the smaller cross section of it. The digital read-out had to be modified, to allow a fast data rate of 160 Mb/s instead of only 40 Mb/s. Due to these modifications also the active pixel area could be increased [6].

1.3.1 The Analogue Read-out

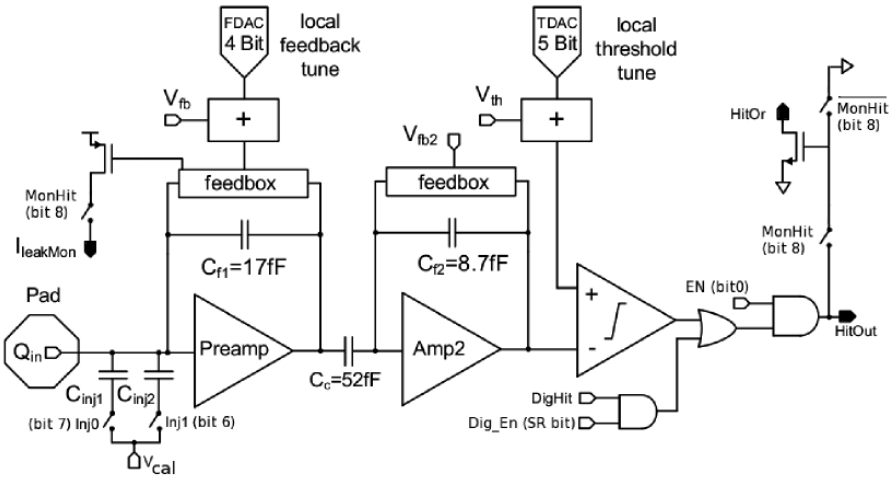


Figure 5: The analogue part of the FE-I4 read-out [7].

Figure 5 shows the analogue circuitry for the FE-I4 read-out. It is implemented in a two stage layout. The two stages allow better optimization, especially improving charge collection efficiency, signal rise time and power consumption.
The principles of such a charge sensitive amplifier (QSA) can be understood with the help of Figure 6. The sensor is modelled by a charge source and detector capacitance C_d. The collected charge causes an input voltage of U_{in} which results in an output voltage of $U_{\text{out}} = -AU_{\text{in}}$. In an

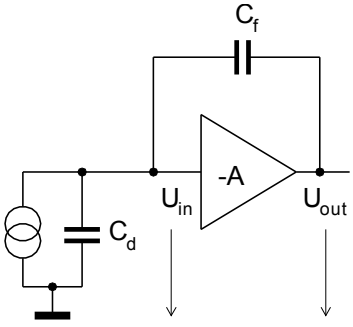


Figure 6: Priciples of a QSA [8].

ideal case no current flows into the amplifier, thus the collected charge equals the charge on the feedback capacitor and the remaining charge on the detector capacitance [8]:

$$\begin{aligned} Q_{\text{coll}} = Q_f + Q_d &= C_f \left(U_{\text{out}} - U_{\text{in}}\right) + U_{\text{in}} C_d \\ &= C_f \left(U_{\text{out}} - \frac{U_{\text{out}}}{A}\right) - C_d \frac{U_{\text{out}}}{A} \\ &= U_{\text{out}} \left(C_f \frac{A+1}{A} - C_d \frac{1}{A}\right) \end{aligned}$$

In the limit $A \to \infty$ one obtains the relation:

$$U_{\text{out}} = \frac{Q_{\text{coll}}}{C_f} \tag{1}$$

For a continuous reset either a resistor or a constant current source has to be placed parallel to the feedback capacitance C_f. This is realized by a NMOS feedback transistor, controlled by a DAC voltage which can be controlled via the FDAC value. After the two amplification stages a discriminator compares the amplifier output voltage with a reference voltage which can be set with the GDAC (global) and TDAC (individual for each pixel) values. Thus tuning this discriminator threshold will be of importance in our task for using the FE-I4 as a trigger plane.
Changing the FDAC value changes the time the signal is over threshold, since this value roughly corresponds to the amount of deposited charge in principle this time (digitised) gives you the amount of charge deposited. Since we are not interested in the total amount of charge, but only if the hit is above threshold we do not have to tune this. The effect of FDAC tuning on the G/TDAC values has not been investigated and is assumed to be negligible.
Simulations of this analogue stage have been performed and show an equivalent noise charge (ENC)[1] dependence on the detector capacitance of 70 e+0.15 e/fF and a timewalk of 20 ns for a 1.4 ke threshold with injected charges ranging from 2 ke to 52 ke [6].

1.3.2 The USBPix Board

For communication between the FE-I4 and the STControl software running on a standard PC a Multi-IO board with an adapter card for the FE-I4 is used [9]. The Multi-IO board features a Xilinx Spartan3 FPGA, a USB 2.0 interface, an 8051 microcontroller (μC) and some SRAM.
The Multi-IO board is designed to be used with a broad spectrum of different adapter cards, thus easy (re)programming of the FPGA is necessary. This is possible with the μC and slave parallel programming, i.e. no additional JTAG or similar is required.
Together with the adapter card for the FE-I4 and the STControl software this forms the DAQ system used in our set-up.

[1] "The equivalent noise charge (ENC) is the signal charge for which S/N=1."[10]

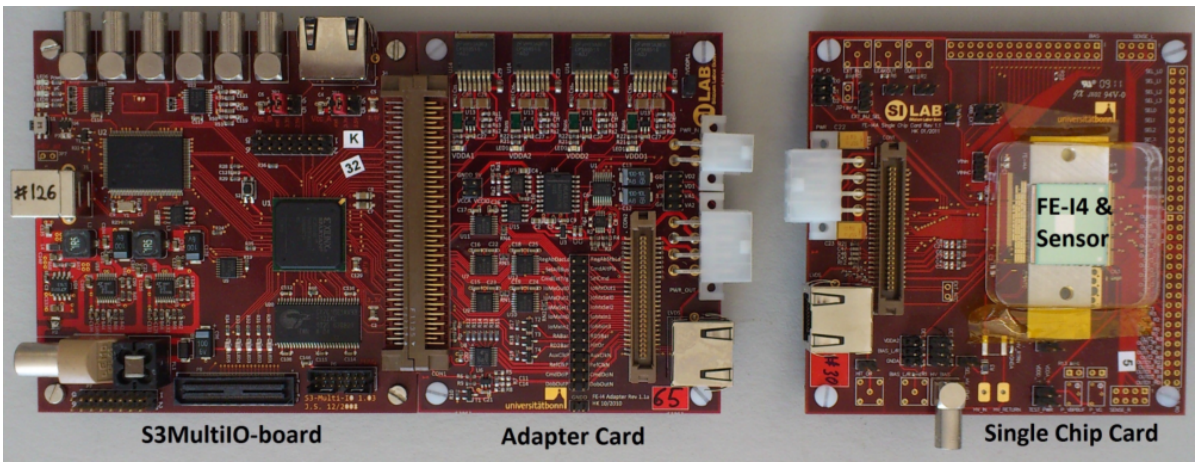


Figure 7: The Multi-IO board with the adapter card connected to it, as well as the single chip card (SCC) onto which the FE-I4 with sensor is mounted [11].

1.4 Motivating the FE-I4 as a Trigger Plane

This project dealt with the replacement of the PMTs as the triggering element with the FE-I4 based pixel module as a trigger plane. There are two major advantages of this approach: since the FE-I4 reads out a pixel detector and delivers a hit bus signal (i.e. a signal when a hit is registered) it can be used for a so-called region of interest (ROI) selection. Since each pixel can be selected to either contribute or not contribute to this hit bus signal this is possible.
A ROI selection can be useful if the DUT is significantly smaller than the Mimosa26 sensor and the trigger area. In that case, selecting only an area around the DUT increases the efficiency by reducing the amount of recorded events, where no tracks go through the DUT.

2 Tuning with the FE-I4

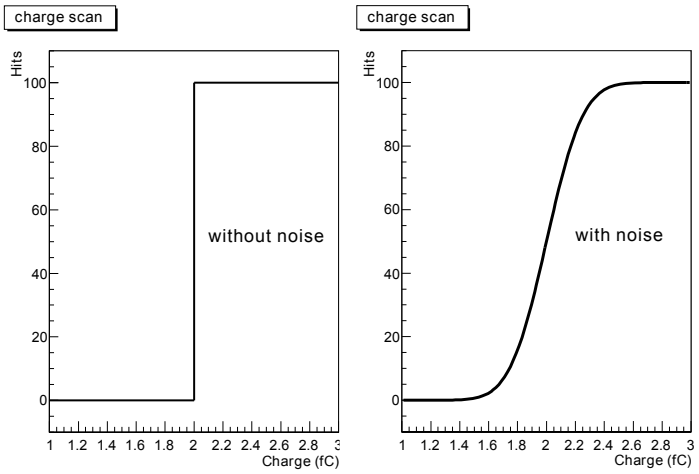


Figure 8: Hit probability with and without noise [8].

In Figure 5 one can see the charge injection capacitors C_{inj} which can be used for threshold tuning. By charging them with a well-defined voltage the amount of charge injected into the preamplifier stage is known. In theory one would expect the detector to report hits, as soon as a certain discriminator threshold (defined by GDAC and TDAC) is exceeded. In reality this is not the case and injected charges just below the threshold charge would fire sometimes due to noise as well as injected charges slightly above threshold would sometimes not, due to the same reason. Origins for this noise can be divided into three categories [8]: thermal noise due to thermal fluctuations in charge carrier distributions, shot noise which is caused by discrete charge carriers which stochastically pass a potential barrier and 1/f-noise for which there are several sources (e.g. charge trapping and release in semiconductors). Thus the ideal hit probability (the probability that a hit is registered depending on the injected charge) deviates from a step function $\Theta\left(Q - Q_{\text{thr}}\right)$ by being convoluted with a Gauss function [8]:

$$p_{\text{hit}} = \Theta\left(Q - Q_{\text{thr}}\right) \otimes \exp\left(-\frac{Q^2}{2\sigma^2}\right) \tag{2}$$

$$= -\frac{1}{2}\text{erfc}\left(\frac{Q_{\text{thr}} - Q}{\sqrt{2}\sigma}\right) \tag{3}$$

This behaviour is shown in Figure 8 for a hypothetical tuning to 2 fC. By doing such scans for every pixel and fitting the complementary error function as obtained in formula 3 (often referred to as S-curve) one not only obtains the charge threshold (Q_{thr}) but also the noise in ENC (σ).

2.1 STControl

STControl is the software used to communicate with the USBPix board, thus allowing to perform various functionality tests as well as performance scans. Ultimately STControl is also used to include the FE-I4 into the telescope datastream, allowing the USBPix to be connected to RunControl as a producer.
In the following sections a general introduction to STControl will be given, as well as a how-to on the threshold tuning of a FE-I4 with a detector and how to include the obtained configuration files into RunControl.

2.1.1 First steps

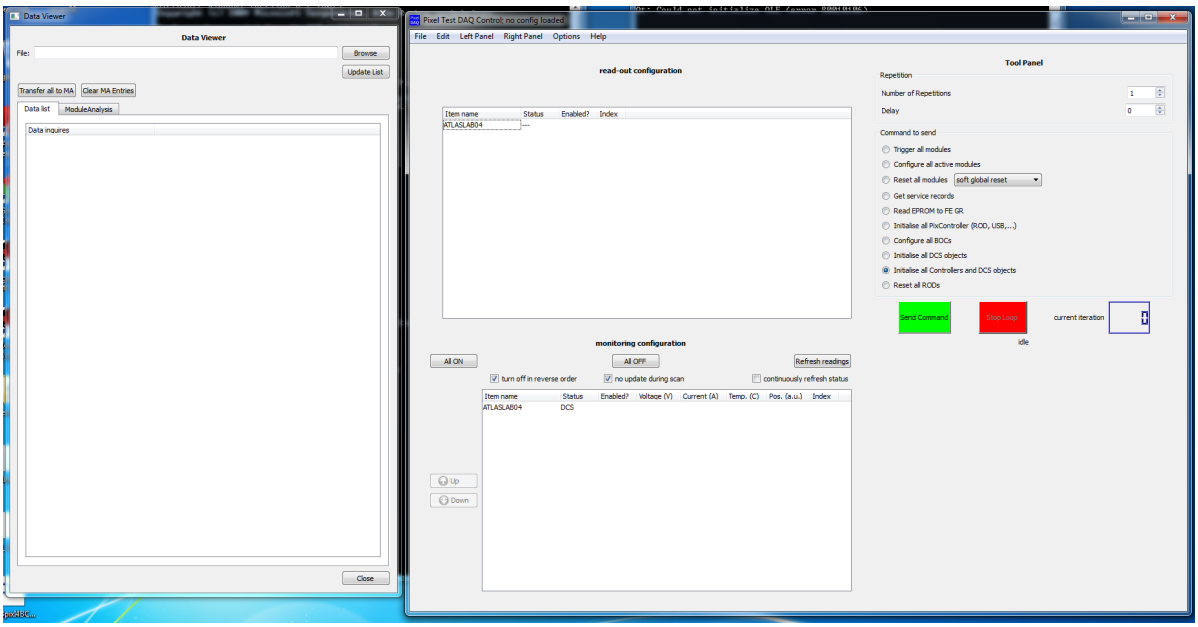


Figure 9: Starting up STControl and the DataViewer.

On the STControl PC open a command line interface, change into `C:\soft\USBpixI4\trunk\` via change directory (`cd`). Execute the batch file `setup.bat` and `cd` into the `bin\` directory. You can then open STControl via `start STControl.exe`. Additionally the Data Viewer can be opened via `start DataViewer.bat`.
In the STControl window, a new configuration file can be created by clicking on `File`→ `New cfg`. The FPGA FW file is the `usbpixi4.bit` and usually preselected. If dealing with only one connected USB board, selecting `single board cfg. (ID=-1)` is sufficient, otherwise the USB board has to be specified. Since we want to create a new configuration file, we select to `generate FE-I4 from scratch` and close the dialogue by pressing `Finish`.
After creating the new configuration file, it is necessary to initialise everything, to do that, one selects in the `Tool Panel` on the right `Initialise all Controllers and DCS`

`objects` and confirms by pressing `Send Command`.
After doing that, one can power up all the voltage regulators on the board by pressing `All ON` in the `monitoring configuration` on the lower left of STControl. On the USBPix four blue LEDs should light up and in the `monitoring configuration` the voltage should read around 1.2 V for the VDD and 1.5 V for the VDA regulators. If this is not the case, check the powering of the USBPix board.

2.1.2 Initial Scans & Masking

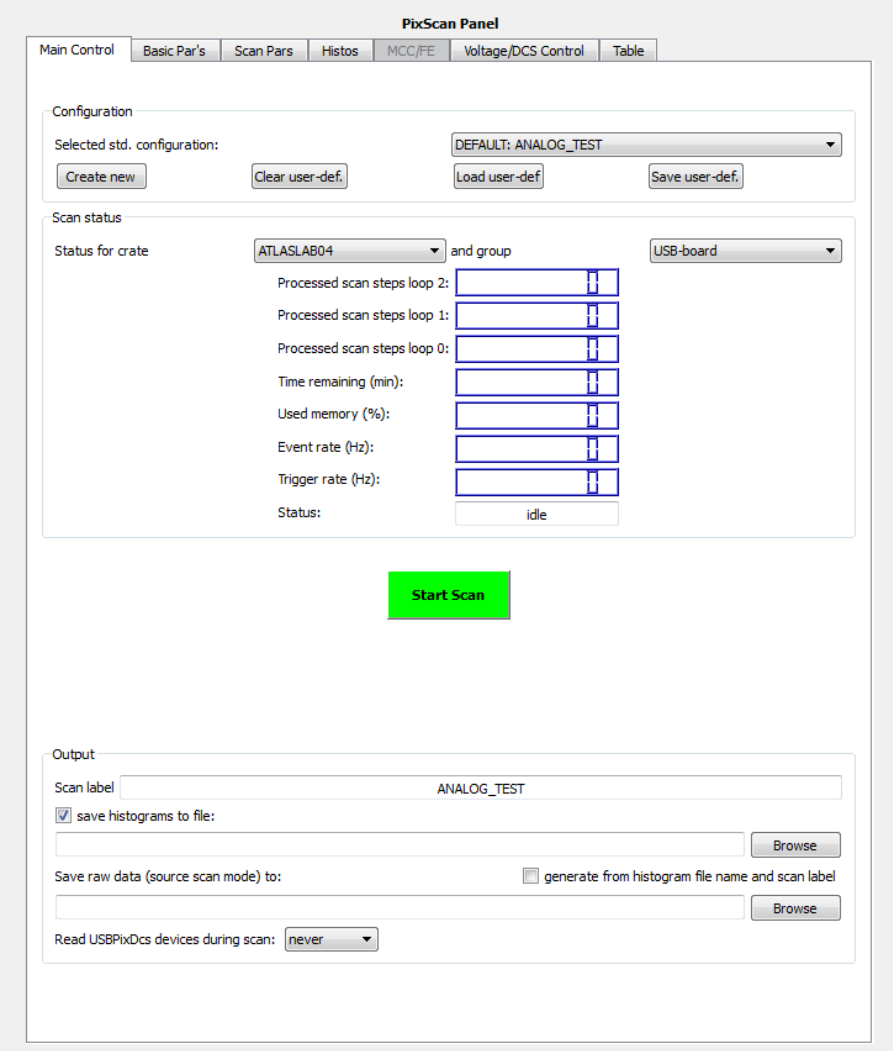


Figure 10: The PixScan panel.

After creating a new configuration file and powering up the voltage regulators, it is crucial to run a few scans. To do this choose `Right Panel`→`PixScan panel` in the upper toolbar (compare with Figure 10). Once this is done, you can select different scans in

the `Main Control` tab of the `PixScan panel`. You do this by selecting a standard configuration. Before running any scans it is advisable to specify to save the histogram to a file (you do that by checking the tick box in the `Output` section of the `PixScan Panel` and choosing the desired file path). Additionally the `Scan label` can be changed, if it is not changed and one scan is performed multiple times, an incrementing enumeration will be added to the label.

The first scan one should perform is the `DEFAULT: RX_DELAY_SCAN`, simply select it from the drop-down menu and press `START SCAN` in the `PixScan Panel`. It should take only a couple of seconds and the result can be checked via the DataViewer. To do that, simply open the `.root` file you specified in the `PixScan Panel` and press `Transfer all to MA`. In the `ModuleAnalysis` tab, an entry for the `RX_DELAY_SCAN` should appear and one can view the histogram by double clicking onto `Plot RAW_DATA_0` in the expanded view of the `RX_DELAY_SCAN` entry.
The histogram should look like the one shown in Figure 11.

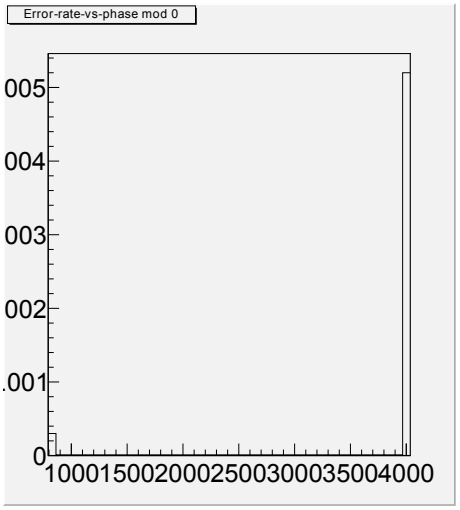


Figure 11: Result of the RX_DELAY_SCAN.

The next two scans which should be performed verify the analogue and digital read-out of the FE. The `DEFAULT: ANALOG_TEST` injects 200 times a fixed charge into the FE preamplifier, well above threshold. The amount of detected hits is recorded and stored in a histogram. Similarly the `DEFAULT: DIGITAL_TEST` sends a signal to the FE discriminator, surpassing the analogue circuitry and only testing the digital one. The results of these scans can be used to mask noisy and dead pixels. After running both scans, one can import them in the DataViewer, to do this press `Update List` and then `Transfer all to MA`. This step is necessary to import any new histograms from any new scans.
You can preview the occupancies for the scans by choosing `Plot OCCUPANCY` in the expanded views of these scans. By right-clicking onto the `Plot OCCUPANCY` entry of the Analog Scan it is possible to create a mask for noisy and dead pixel masking. By choosing $199 < \text{OCCUPANCY} < 201$ we create a mask which only enables pixels which recorded exactly 200 events.
To import the mask in STControl you select `Edit`→`set all masks from scan` in the upper toolbar. In the appearing open dialogue select the `.root` file you saved your data in. Then select the `ANALOG_TEST` as a scan and use the `HITOCC` histogram as a mask. Tick the check boxes to apply them to `ENABLE` and `ILEAK` as shown in Figure 13.

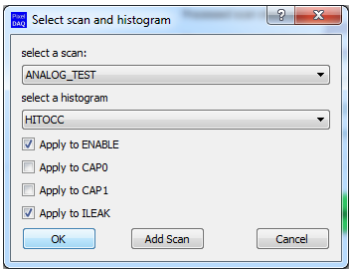
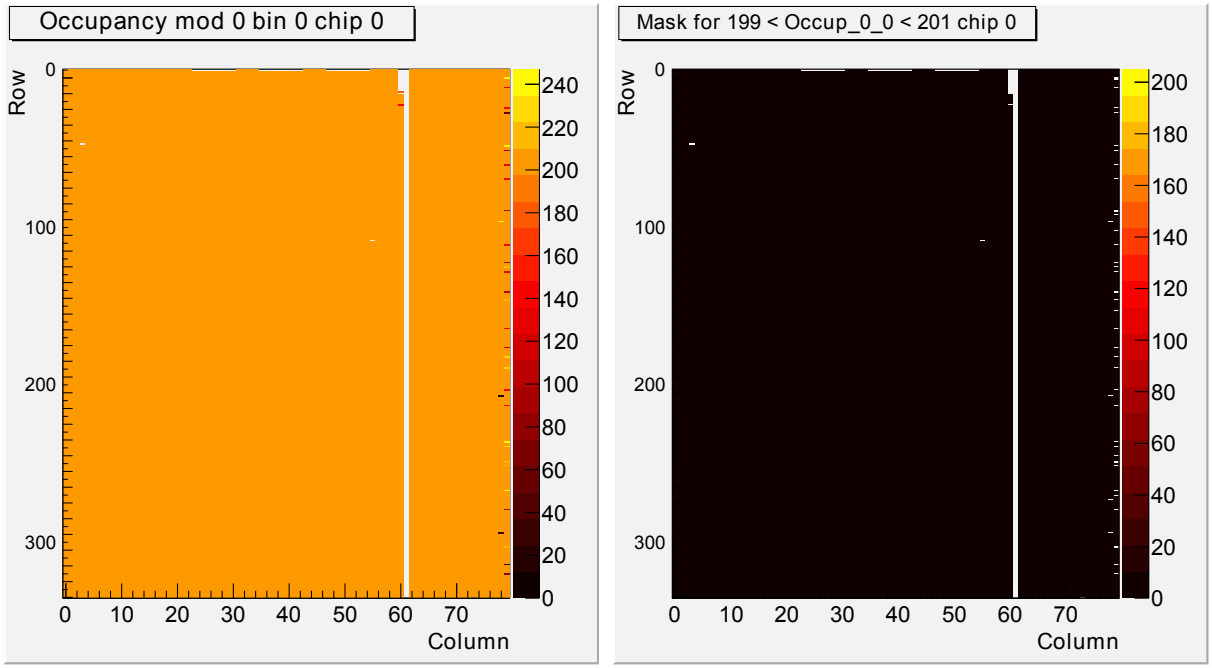


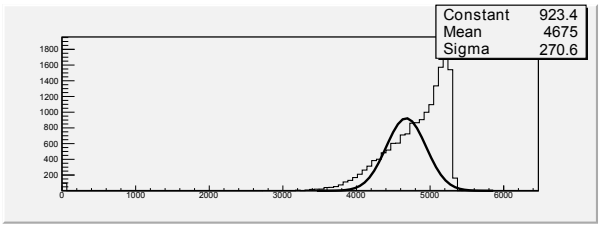
Figure 13: Importing a mask.



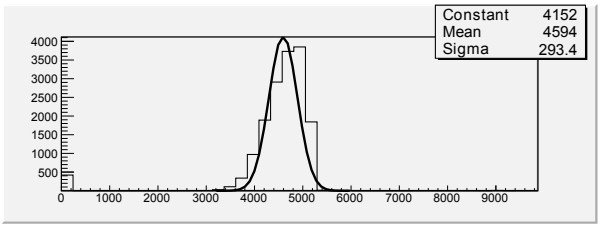
(a) Result of an ANALOG_SCAN. (b) A derived HITOCC mask.

Figure 12: ANALOG_SCAN and mask.

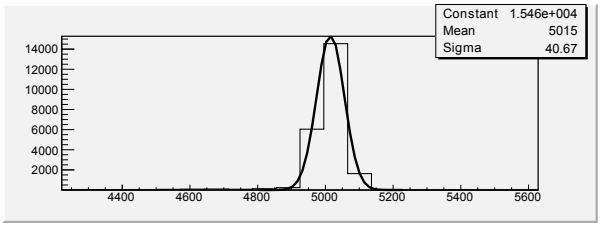
After doing that we can open the `Config Editor I4` by double left clicking onto the module name (by default this is SC1) in the `read-out configuration` on the left panel of STControl. The `Config Editor I4` window should pop up and you can select the `FE Masks, Pixel` tab. In the upper most section, we can diasble the outermost columns, since they are usually noisy. To do this you right click onto the entry below `DisableColumnCnfg` (there should be bunches of zeroes there), causing the `Column Pair Switch` window to appear. Click onto the first two (1,2) and last two (39, 40) buttons and confirm this by pressing `OK`. This should change the first and last two zeroes to ones.
The `ILEAK` mask has to be inverted, to do this you right click onto the `map` entry for `ILEAK` in the lowest section of that window and select `Invert`. By right clicking onto it and selecting `Display/Edit` you can also view the set mask.
After setting the mask it is advisable to save your configuration file (it is recommended to do that very often, in case STControl crashes). Simply do that via `File→Save cfg file` or by pressing `Ctrl+S`.
After disabling noisy pixels it is useful to do the `DEFAULT: NOISE_OCC` scan. With the default settings for the scan it may take a few minuites. After the scan is complete one can `Update List` and `Transfer all to MA` in the DataViewer. The `Plot OCCUPANCY` for this scan should not show any noisy pixels anymore.



(a) Before tuning.



(b) After GDAC tuning.



(c) After GDAC and TDAC tuning (zoomed in).

Figure 14: Threshold tuning.

2.1.3 Threshold Tuning

As discussed before (see Section 1.3) the discriminator level has to be tuned to a given value for an injected charge. This is controlled by a global DAC value (GDAC) and one specific for each pixel (TDAC). To tune them we can fortunately use predefined scans. Before doing so we can look at the current threshold dispersion by doing a `DEFAULT: THRESHOLD_SCAN` scan. If you select this scan in the drop down menu, you can change to the `Scan Pars` tab in the `PixScan Panel`. In the `Loop 0` section you can choose to Scan from 0 to 100 (default would be 0 to 200) and specify the number of steps to be only 21 instead of 201. For quick scans these low values are recommended, they greatly increase the speed of the scans. For the final threshold dispersions, these values should be left unchanged to get a precise result. It must be noted, that as soon as you select any different scan from the drop down menu, all values are reset to their default ones. Thus when doing more quick scans, this has to be changed every time.
After the scan one can again use the DataViewer to view the results. Import the scan and look at the `SCURVE_MEAN`. Sometimes it can happen, that there is still a noisy pixel, the uppermost histogram will then range far above a threshold of 10 000. With the help of the DataViewer the values from these pixels can be ignored for plotting. In the `ModuleAnalysis` tab choose `Options`→`Show Panel`. In the window choose the `PixLib plotting` tab and search for the `SCURVE_MEAN` histogram line. Change the Min value to 0 and the Max to 10 000 and set Enal to 1 to enable this plotting range. Confirm via pressing `Save+Close`.
The `SCURVE_MEAN` plot should now look much better. A gaussian is fitted to the data and gives the mean threshold as well as its dispersion. Obviously if no tuning has been done, the distribution might not be gaussian, but this will change after tuning and therefore should cause no worry (see also Figure 14).

To tune the sensor first run the `DEFAULT: GDAC_FAST_TUNE` scan, the desired threshold can be specified once again in the `Scan Pars` tab. For that, change the threshold target value for G/TDAC tuning in the `Tuning parameters` section. Once again, this value gets overwritten by its default one of 3000 whenever you change the scan, also when changing from GDAC to TDAC tuning. After doing this tuning you can check the results via a `THRESHOLD_SCAN` or just continue doing the `DEFAULT: TDAC_FAST_TUNE`. Obviously the same target value as before should be set. After this save your configuration file and do a (long) `THRESHOLD_SCAN` to verify that the tuning has worked. If no more noisy pixels show up and the threshold is at the targeted value with a low dispersion, your tuning is done and the configuration file can be used for testbeam operation.

2.1.4 ROI selection

To select a ROI open the ILEAK mask editor as described in Section 2.1.2 (choose to `Display/Edit` the ILEAK mask). In the editor it is possible to select a rectangular ROI with the help of the `ROI` section (marked in green in Figure 16). You simply do that by selecting the lower and upper row and column and clicking on ROI. This sets the desired ILEAK values for the chosen ROI to 0 which is what we want.
If any noisy pixels get marked in the ILEAK mask (set to 0), this is not a problem. Since we have disabled them via the ENABLE mask they do not contribute anyway (this also means, in principle one could simply set the whole ILEAK mask to 0 if no ROI is chosen instead of inverting the applied ENABLE mask). After changing the mask, do not forget to save your configuration file. If you want to immediately use the changes without reloading the configuration file, it is necessary to `Configure all active modules` in the `Tool Panel` (see Figure 9 and 15).

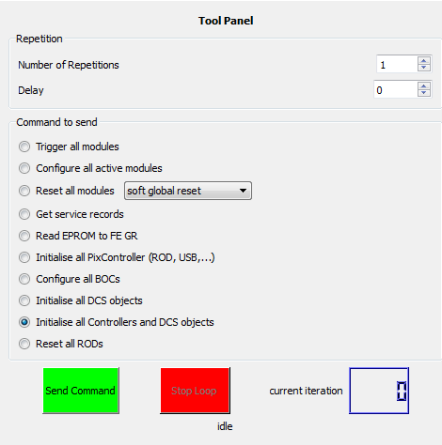


Figure 15: Tool Panel in STControl.

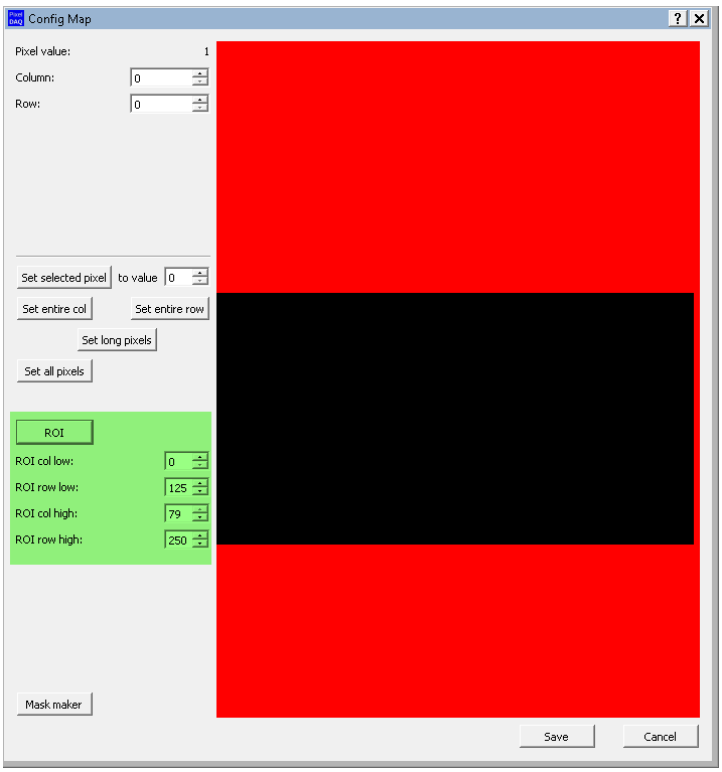


Figure 16: ROI selection.

2.2 Integration in RunControl

As previously mentioned the USBPix can be easily integrated into the telescope datastream. To add it as a producer, one must add a corresponding producer section in the desired configuration file. Since we are using two USBPix boards (given that we are reading out two FE-I4s) we must either add a dual board STControl configuration file or two single board ones.

Two Single Board Configuration Files

```
[Producer.USBpixI4]
boards = 9,15
config_file[15] = C:\path\to\config\file\config_for_board_15.cfg.root
config_file[9] = C:\path\to\config\file\config_for_board_9.cfg.root
lvl1_delay = 26
SRAM_READOUT_AT =  10
UseSingleBoardConfig = Yes
fpga_firmware = C:\soft\USBpixI4\trunk\config\usbpixi4.bit
```

In this case, the boards with the ID 9 and 15 are used (these identifiers are hard coded and are most probably written on the board, otherwise they can be seen when creating

a new configuration file in STControl in the drop down menu where you can select the board). The path of the configuration files is the path on the PC on which STControl will be running. Generally this is not the PC on which RunControl is running.
The `SRAM_READOUT_AT` controls after what amount of collected data the data is sent to RunControl. Together with the `lvl1_delay` value these two parameters might need to be adjusted by the user once at the beginning of operation. For that the `lvl1_delay` needs to be verified by looking at the lvl1 distributions of the USBPix boards in the OnlineMonitor and might need some slight adjustments.

One Dual Board Configuration Files

```
[Producer.USBpixI4]
config_file = C:\path\to\config\file\config_for_dual_board.cfg.root
lvl1_delay = 26
SRAM_READOUT_AT =  10
```

In this case only one dual board configuration file is used.

Connecting STControl with EUDAQ

After creating the configuration file, and starting RunControl on the RunControl PC you have to start STControl on the PC to which the USBPix boards are connected. You do that in a similar way as described in 2.1.1 but with minor changes. `cd` into `C:\soft\USBpixI4\trunk\` and run `setup_with_eudaq.bat`, `cd` into the `bin\` directory and execute `STControl_eudaq.exe -r 192.168.1.1` where the `-r` flag tells STControl to connect to RunControl at `192.168.1.1` as a producer. Needless to say, you need to use the IP of the PC in your network which runs RunControl.

3 The Signal Inverter

3.1 The HitOR Signal

The FE-I4 provides a hit bus signal, i.e. a signal when one of the pixels records a hit above threshold. For this the HitOut from Figure 5 for all pixels are OR'ed together to form the HitOR signal. The ILEAK mask we set during the tuning procedure is crucial for this, setting the ILEAK bit for a pixel to 0 enables this pixel to contribute to the hit bus. This can also be seen in Figure 5 where the MonHit bit turns on the leakage current output as well as the HitOR output.

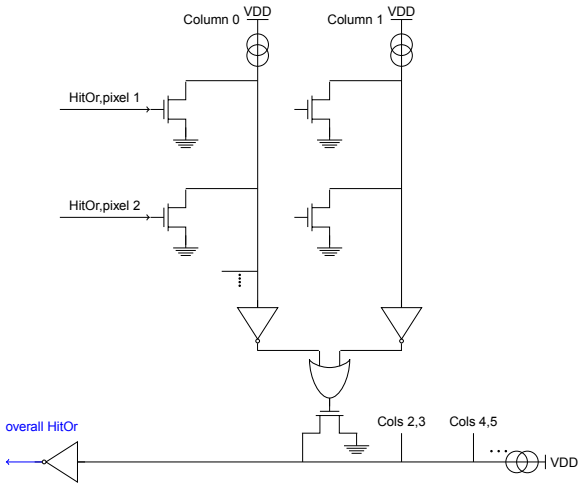
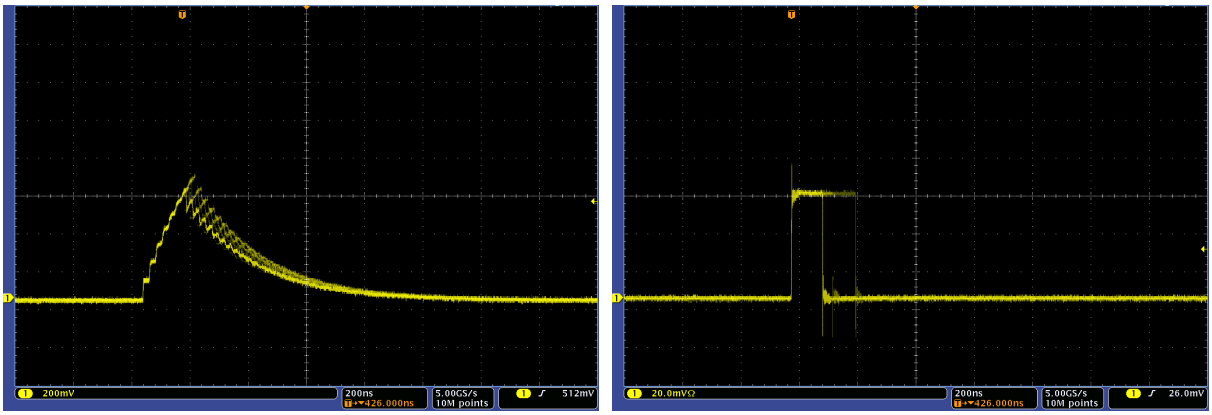


Figure 17: OR'ing together all the HitOR signals from all the pixels [11].

To form the bus the individual HitOR pixel signals are acting as a gate of a NMOS-transitor as shown in Figure 17. As soon as there is a HitOR from any of the pixels in a column, the transistor will pull down the line from this column. Thus to get a positive signal in an event of a hit, the signal gets inverted. After that, a pair of columns is OR'ed together. This output then again acts as a NMOS-gate, pulling down the overall HitOR line in case of a hit. An inverter at the end causes the positive signal in case of any hit of a contributing pixel.

Since the TLU triggers on a falling slope (see Section 1.2.1) it is a necessity to invert the signal. For this we built, tested and characterized two different inverter models which will be described in the next sections.



(a) HitOR into high impedance (200 mV/div). (b) HitOR with 50 Ω termination (20 mV/div).

Figure 18: HitOR into different terminations, both images 200 ns/div.

Figure 18 shows the HitOR signal monitored with a scope while the sensor was exposed to beam. The length of the HitOR signal, when terminated with 50 Ω ranges from roughly 100 ns to about 200 ns. Initial plans were to use an unterminated high impedance inverter, motivated by experience from other groups. Obviously when using a non terminated cable one gets reflections which can also be seen in Figure 18a. These reflections get less of a problem when very short cables are used, therefore mounting the inverter directly onto the FE-I4 would be the best solution. Unfortunately since the set-up had to be operated in a magnetic field of a little above 1 T and the inverters contained some slight magnetic components this was not possible, instead it had to be tied via cable ties to the framing and connected via a short LEMO cable.

3.2 The First Inverter

The first inverter uses an Analog Devices AD8131, which is a low cost, high speed differential driver. It features a -3 dB bandwidth of 400 MHz and a slew rate of 2000 V/μs. The circuit has a fixed gain of 2 without any external components, thus the layout is relatively simple. Figure 21 shows a picture of one of the inverters. A schematic of it is given in Figure 19. While the schematic suggests that the circuit is some strange open loop circuit, one has to actually consider the functional diagram of the AD8131 as given in Figure 20 which features a standard closed loop circuit with a fixed gain of 2.

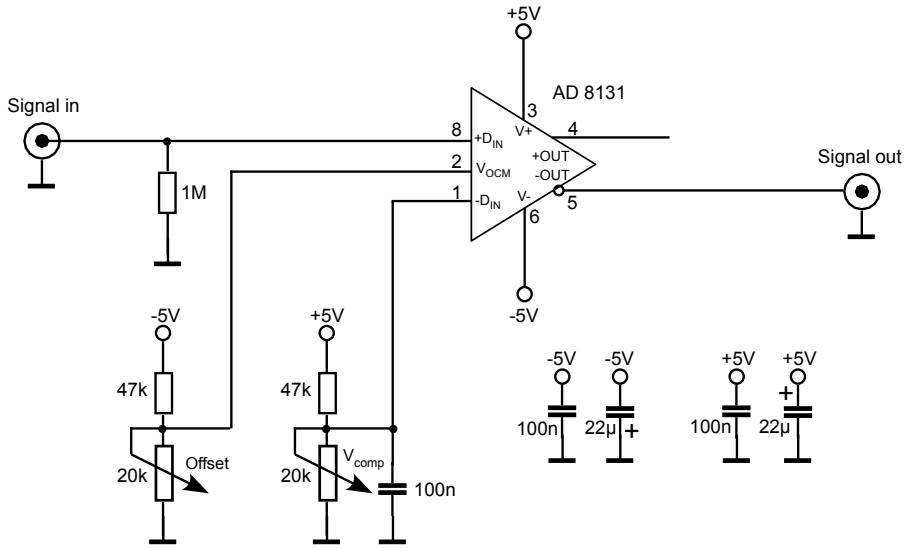


Figure 19: Schematic of the first inverter.

The two potentiometers allow to adjust the offset voltage as well as the comparator threshold. As can be seen from the choice of termination as well as the resistor/potentiometer values the design was planned to be used with the unterminated, relatively large HitOR signal.

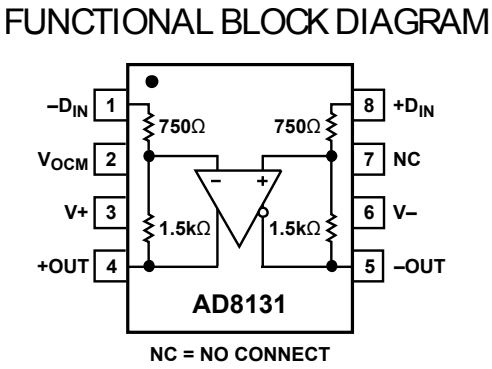


Figure 20: Functional block diagram of the AD8131 [12].

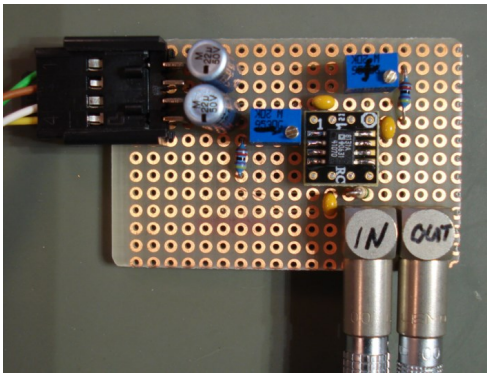


Figure 21: Picture of the first inverter.

3.3 The Second Inverter

The design goal for the second inverter featured a very high input impedance and real inversion without the need of any bias or offset voltage. Thus the most simple circuit, featuring a unity gain stage (for high impedance) as well as an inversion stage was built. The design was mostly copied from the datasheet, leaving away unnecessary parts of it. The schematic is given in Figure 22.

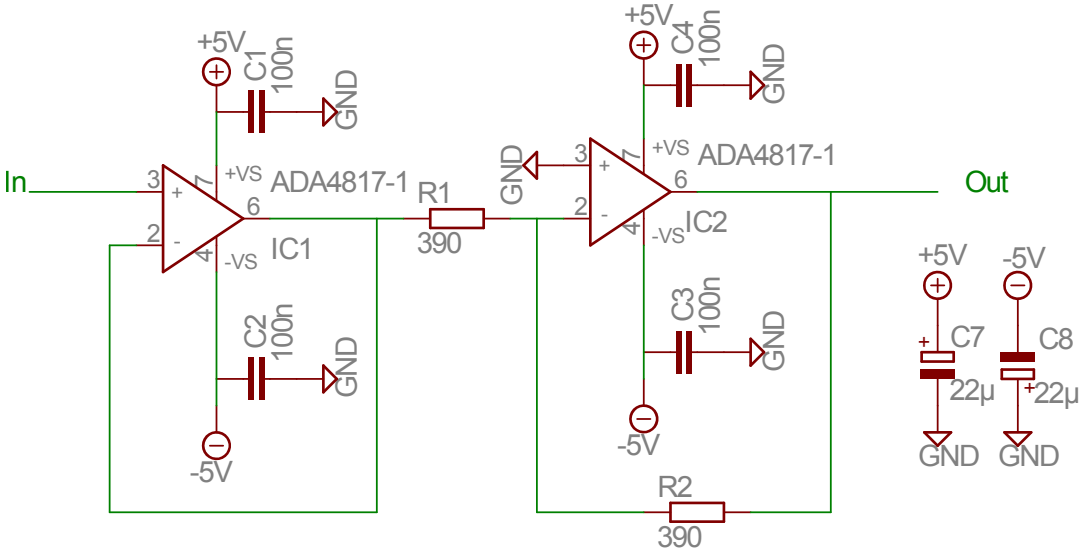
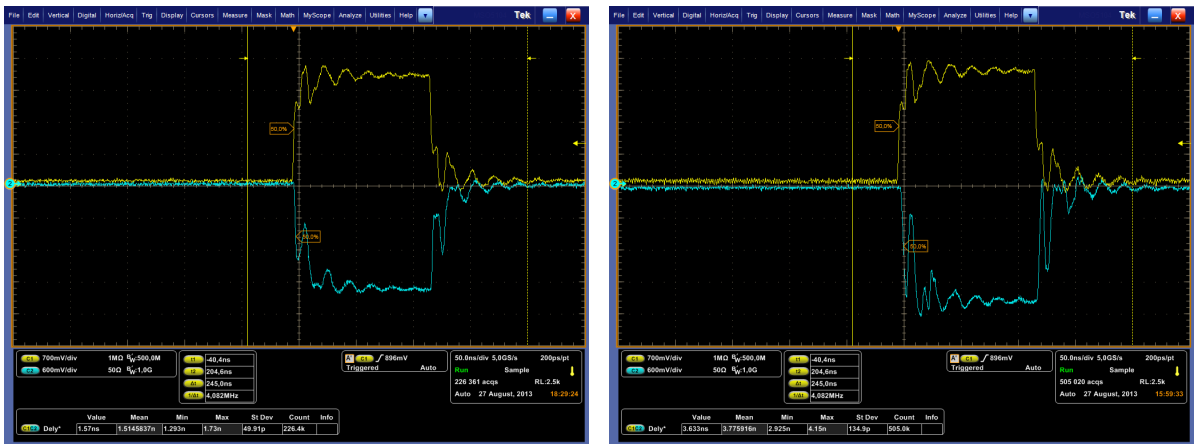


Figure 22: Schematic of the second inverter.

The initial prototype, built with through-hole technology on drill hole prototyping boards had a ripple of roughly 100 mV (p-p) due to the circuit going into oscillation. Thus a second version was built with mainly surface mounted technology and appropriate components as surface-mount devices, but still on a drill hole board. This could decrease the ripple to below 20 mV (p-p) but not nullify it.



(a) First inverter. (b) Second inverter.

Figure 23: Delay measurements for the two inverters, yellow shows the signal from the signal generator and cyan the inverted one.

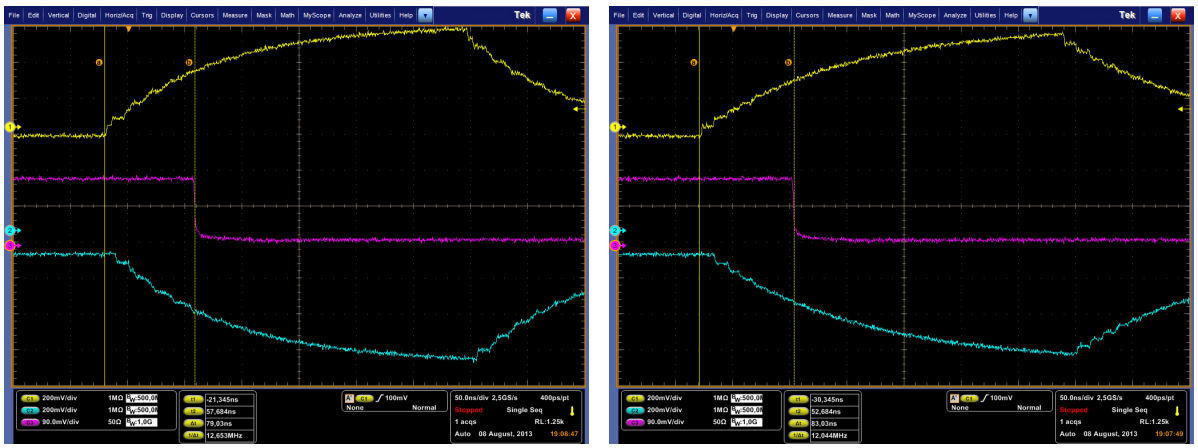
3.4 Performance of the Inverters

Using a signal generator, pulses with a very short rise and fall time of only 1 ns and a duration of 120 ns were fed into the different amplifiers. The standard deviation of the delay was measured with the help of a scope. Snapshots of these measurements can be seen in Figure 23a for the first inverter and in Figure 23b for the second one, respectively. The overall delay is biased by the used cables (the same cables were used in both measurements), the one connecting the output of the signal generator directly to the scope was 20 cm shorter than the cable going from the signal generator to the inverter and then to the scope. Thus an additional delay of approximately 1 ns due to the extra cable has to be accounted for (assuming a signal propagation at $\frac{2}{3}$c).
The first inverter had an overall delay of 1.6 ns with a standard deviation of 50 ps, whereas the second one had a delay of 3.6 ns with a standard deviation of 135 ps.
Given, that the second inverter features two stages, a larger delay and also a larger standard deviation is what one would expect. Additionally the ripple of the second inverter will decrease the stability of the delay.
These low standard deviations[2] are of no concern for the application of the inverters in the trigger plane set up.

Impact of the cable length

As can be seen in the Figures 24a and 24b the cable length has some impact on the signal shape, if it is not terminated. Due to the slower rise time (because of the low-pass properties of a real cable) a time walk occurs, before a trigger is issued by the TLU. Additionally one can again see unwanted reflections in not correctly terminated cable.

[2]Given a sampling rate of 5 GS/s of the scope, which corresponds to timesteps of 200 ps, a resolution of $\frac{200\,\text{ps}}{\sqrt{12}} \sim 60\,\text{ps}$ can be assumed to be reasonable.



(a) Signal with a 40 cm cable.

(b) Signal with a 100 cm cable.

Figure 24: Yellow shows the HitOR, cyan the inverted signal and magenta the TLU test output.

3.5 Outlook considering the Inverters

It is strongly encouraged to use a 50 Ω termination and an amplifier with an inverting circuitry featuring some minor amplification (a too high gain would limit the bandwidth). This was tried with the two given inverters, terminating the LEMO cable via a T-piece and a 50 Ω termination resistor. The problem with the first inverter was, that the comparator voltage had to be adjusted in the small window of 50 mV. This is not an issue when using a real inversion with the second inverter. Unfortunately the ripple of the second inverter became an issue, given that the terminated signal is only roughly 50 mV (p-p). Therefore also the second inverter could not be reliably used in the set-up with a 50 Ω termination.
The proposal for the future is therefore to use a single stage inverter with a gain of somewhat around 2-4 and a 50 Ω input impedance.

4 The Set-Up

In this section the experimental set-up will be described with emphasis on the correct cabling and connection of all the involved components.

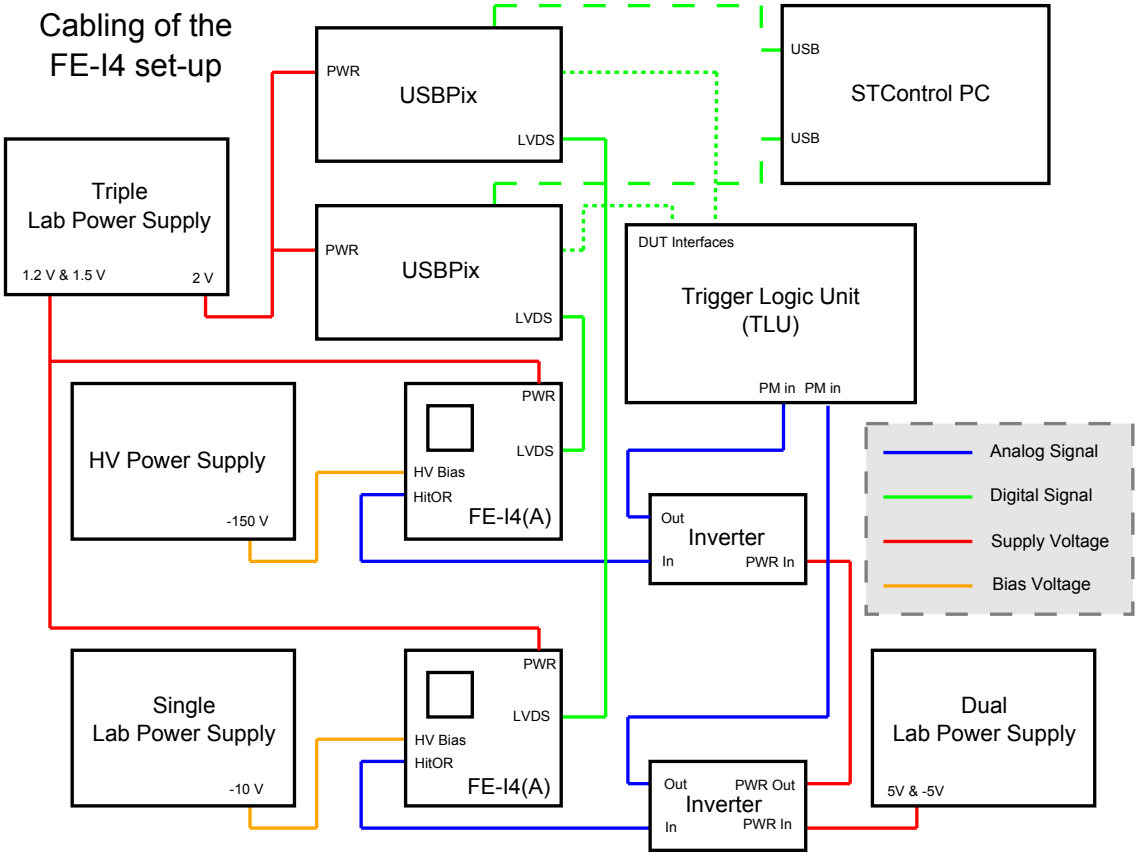


Figure 25: Schematic cabling of the complete set-up.

Figure 25 shows a general overview of the complete cabling.

4.0.1 The FE-I4

The Single Chip Card (SCC) which holds the FE-I4 (see Figure 7) is mounted via an aluminium framing on the most downstream and upstream telescope plane. The Molex connector (labelled PWR on the PCB) on the SCC is the connector for the power supply of the FE-I4. It uses +1.2 V for the digital part and +1.5 V for the analogue one. If long power cables are used, it is recommended to use sense leads.
The connection between the SCC and adapter card is via a standard CAT-5/6 cable with RJ-45 connectors (the connectors on both the SCC and adapter card are labelled LVDS on the PCB).
One of the two LEMO sockets is the HitOR signal (the outer one, it's labelled HIT_OR

on the PCB) and the other one is the bias voltage (label HV_BIAS). Depending on which sensor you use you apply the desired negative bias voltage to the HV_BIAS LEMO. The signal from the Hit_OR goes to the inverter input.

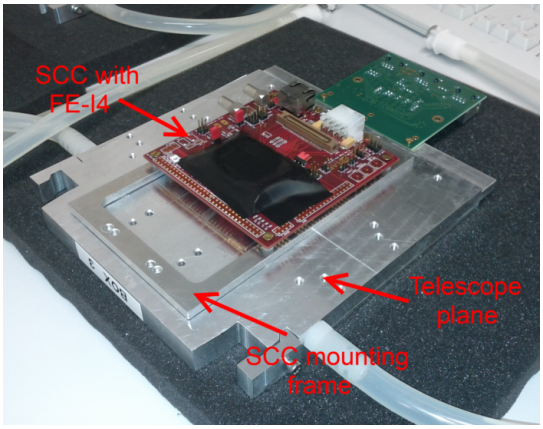


Figure 26: Mounting frame for the SCC.

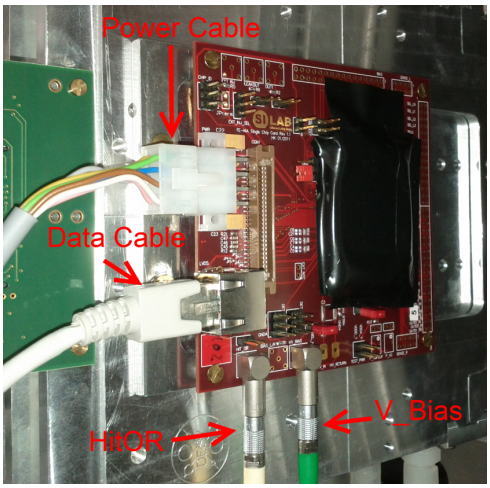


Figure 27: Cabling of the SCC.

Depending on which sensor is mounted onto the FE-I4 different bias voltages are used. Planar sensors are biased with approximately -150 V whereas only -10 V are applied to 3D sensors.

4.0.2 The USBPix board

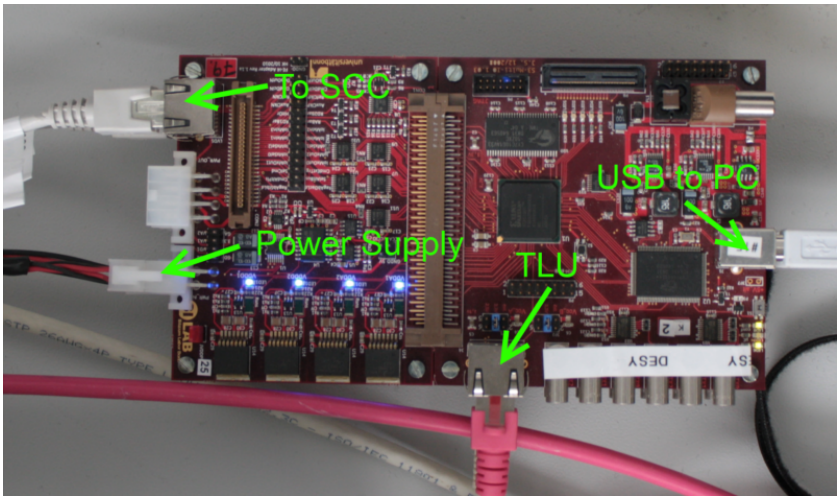


Figure 28: Cabling of the USBPix board.

The cabling of the USBPix boards is pretty straight forward. The one RJ-45 connector (with the LVDS label on the PCB) is the connector to the SCC. The small (4-pin) Molex

connector (label PWR_IN) is the power supply for the board. The other 8-pin Molex (label PWR_OUT) could be used to supply the SCC with the desired supply voltage, but the possibility of sense would be lost. The board uses a single supply voltage of +2 V. The other RJ-45 connector is for connecting the USBPix board to the TLU and the USB connector is for the connection to the PC running STControl.

4.0.3 The Inverters

Both inverter models are equipped with the same power connector. Figure 29 shows the layout, it can be easily verified by looking at the polarity of the electrolytic capacitors used for buffering. Their negative electrode is usually marked in some way.
In principle one can daisy chain the power from the first inverter to the second one, in this case it might be useful to check the voltage at the second inverter, since there might be some voltage drop over the multiple connectors.
The inverters feature two LEMO connectors, one input and one output. If no 50 Ω termination is used, it is recommended to keep the cable between the HitOR output from the SCC to the inverter input as short as possible. The output of the inverter is fed directly into the TLU with a LEMO cable.

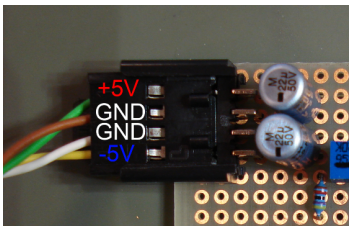


Figure 29: Pin layout of the power connector of the inverters.

5 Results, Outlook and Conclusion

5.1 Running with the Trigger Plane

5.1.1 HitMaps and Correlation Plots

Two FE-I4s, both with the first inverter using no 50 Ω termination, were used during a few runs in testbeam 24/1 here at DESY. The DESY SCC-CNM-102 (with a 3D sensor), biased with −10 V was mounted on the most upstream telescope plane (USBPIXI40) whereas the DESY SCC-PPS-169 (planar sensor) with −150 V bias voltage operated as the most downstream trigger plane (USBPIXI41). A coincidence was required on both of them.

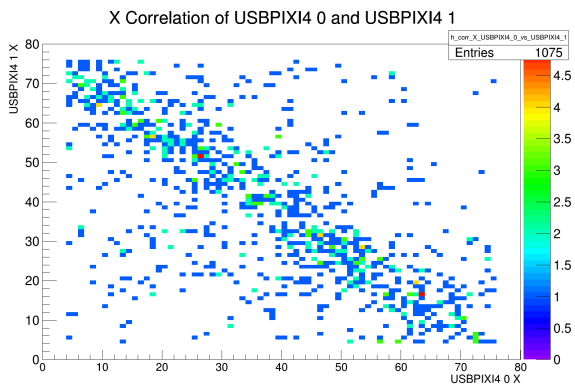


Figure 30: Correlation plot of the two FE-I4 tigger planes in x-direction.

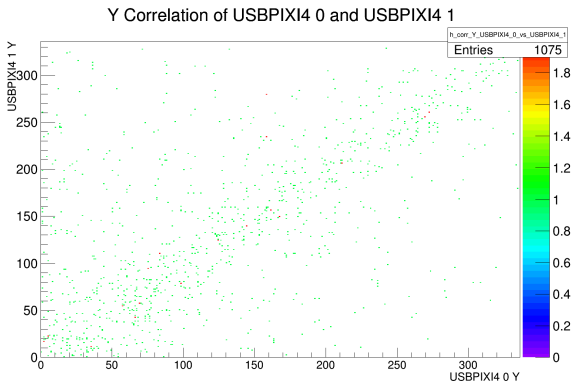


Figure 31: Correlation plot of the two FE-I4 tigger planes in y-direction.

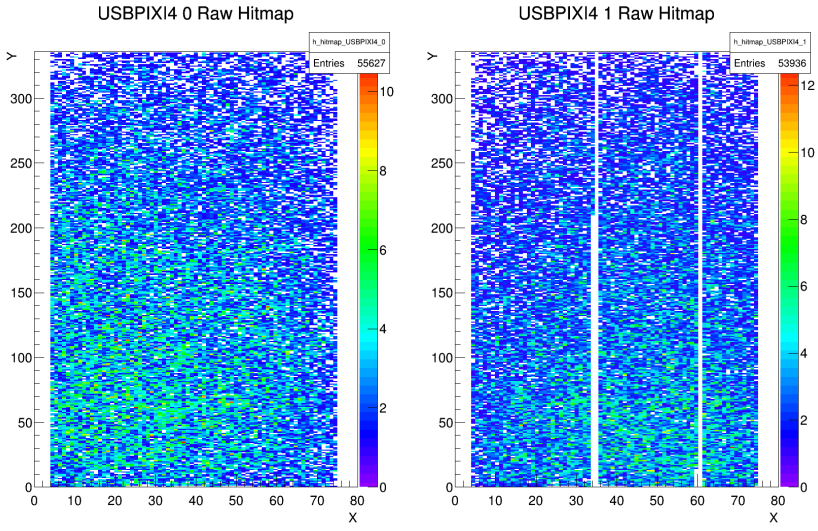


Figure 32: HitMap for the CNM-102 (left) and PPS-169 (right).

Figure 30 and 31 show the correlation plots for the set-up described above. The x-direction is anti correlated because of the way the planes are mounted onto the telescope. Due to the fact that the FE-I4 features a lot more pixels in y-direction the y-correlation is a little more difficult to see on the plots. What can be seen, is that for the USBPIXI41 the column slightly above 60 and around 35 does not correlate with any pixels on the USBPIXI40. This is due to the fact, that on the SCC-PPS-169 (USBPIXI41) which was used for this test set-up had dead columns at those positions. This can also be verified by looking at the the hitmap plots, see Figure 32. The outer two columns on both sides were disabled to reduce noise, thus yielding no hits nor correlations.

5.1.2 Region of Interest selection

Additionally a ROI was selected. The corresponding ILEAK mask can be viewed in Figure 33. Six telescope planes (called MIMOSA26 0-5) were used in the set-up. The ROI was selected for the upstream trigger plane (just before MIMOSA26 0). The resulting hitmaps can be seen in Figure 34.
It can be clearly seen, that for the further downstream planes, the sharp edge of the ROI diffuses. This is what one expects, especially if using a relatively large collimator (which was the case) and there being a lot of additional material in the path. This is the case in testbeam 24/1, as the telescope is placed inside a large solenoid magnet, the beam thus gets scattered while passing through it.
Additionally a lot of tracks do not go through the ROI, since during a read-out, one usually not only obtains one track but multiple. Thus if any particle going through the ROI triggers, there might be two (or more) tracks, not all of them necessarily going through the ROI.

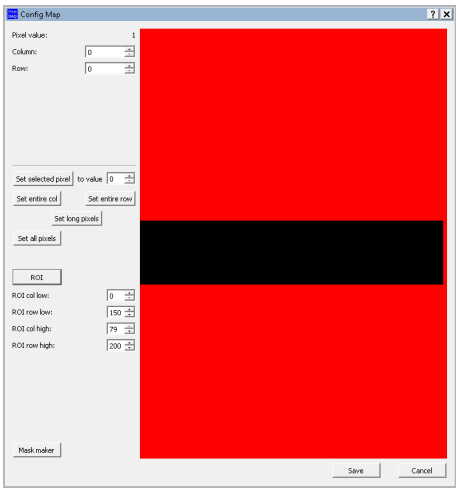


Figure 33: ROI selection.

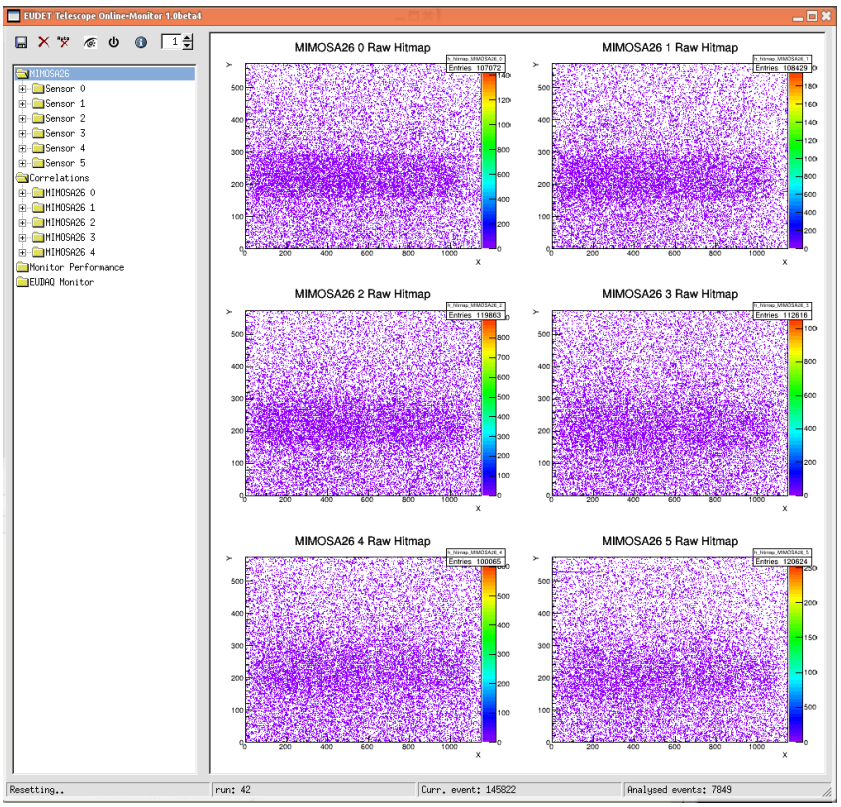


Figure 34: HitMap for the MIMOSA26 sensors.

5.2 Outlook

While triggering with the FE-I4 works in principle, the way it was implemented using a high impedance inverter and some short but not neglectable cable, was very unstable. This does not guarantee real precise timing, not because of the inverter performance, but due to the distortion of the signal and cable reflections (see Section 3.5 for more detail on this).
The ROI selection worked out of the box, the STControl software allows to easily prepare and tune a sensor to a desired threshold as well as to select a ROI. The planar sensor that was used unfortunately had two dead (double) columns, but in the final set-up will be replaced by a new one.
No work has been done yet on the timing of individual tracks in the EUTelescope analysis software.

5.3 Acknowledgement

I would like to express my gratitude to Igor Rubinsky, who not only supervised me during my stay here at DESY during the 2013 DESY Summer School but also had to bear up against a lot of my questions. Thank you for showing me this hands-on part of

particle physics which was entirely new to me and I learned to love!
I must also thank the DESY telescope and testbeam team and all people related to this. In particular Ingrid-Maria Gregor, Marcel Stanitzki, Ulrich Kötz, Phillip Hamnett and Richard Peschke who assisted me at least at some point during my work.
Additionally I got of lot of help from Torsten Külper concerning electronics, for that I am very grateful.
A great deal of this wonderful experience was due to the summer students who also worked at the testbeam and made this summer as perfect as it could be. I would like to name Riccard Andersson, Anne Schütz and Paul Schütze, thank you ◡̈!
I must thank the DESY ATLAS group for their warm welcome and the opportunity to work with them as well as all the organisers of the DESY 2013 Summer School for making all this happen.

6 References

[1] J. Baudot, *First test results Of MIMOSA-26, a fast CMOS sensor with integrated zero suppression and digitized output*, Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE, **2009**.

[2] D. Cussans, *Description of the JRA1 Trigger Logic Unit (TLU), v0.2*, EUDET-Memo-2008-50, **2008**.

[3] `eutelescope.desy.de`

[4] H. Krüger, *Discriminator Board (TLU_DB) for the JRA1 Trigger Logic Unit (TLU)*, **2006**.
`https://webpp.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Docs/TLU_Discriminator_Board_Documentation.pdf`

[5] M. Capeans *et al.*, *ATLAS Insertable B-Layer Technical Design Report*, CERN-LHCC-2010-013; ATLAS-TDR-19, **2010**.

[6] M. Karagounis *et al.*, *Development of the ATLAS FE-I4 pixel readout IC for b-layer upgrade and Super-LHC*, CERN-2008-008, **2008**.

[7] J. Albert *et al.*, *Prototype ATLAS IBL Modules using the FE-I4A Front-End Readout Chip*, JINST **7**, P11010, **2012**.

[8] J. Große-Knetter, *Vertex measurement at a hadron collider - the ATLAS Pixel Detector*, BONN-IR-2008-04, **2008**.

[9] J. Schneider, H. Krüger, *S3 Multi IO System - S3 Multi IO USB Card*, V1.03, **2010**.
`http://icwiki.physik.uni-bonn.de/svn/USBpixI4/doc/MultiIOboard_V1.03_Manual.pdf`

[10] C. Grupen, B. A. Shwartz, *Particle Detectors*, 2$^{\text{nd}}$edition, Cambridge University Press, **2008**.

[11] T. Obermann, *Development of a test beam telescope based on the ATLAS front end ASIC FE-I4*, BONN-IB-2012-14, **2012**.

[12] Analog Devices, Inc., *Low Cost, High Speed Differential Driver AD8131 Datasheet*, Rev. B, **2005**.
`http://www.analog.com/static/imported-files/data_sheets/AD8131.pdf`