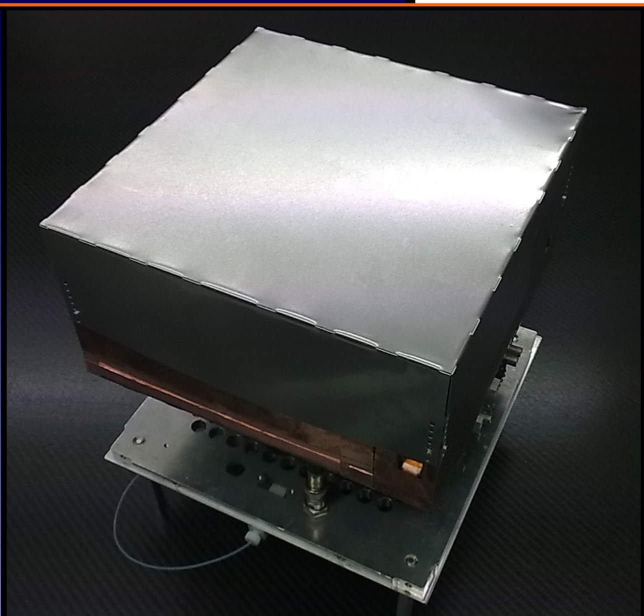
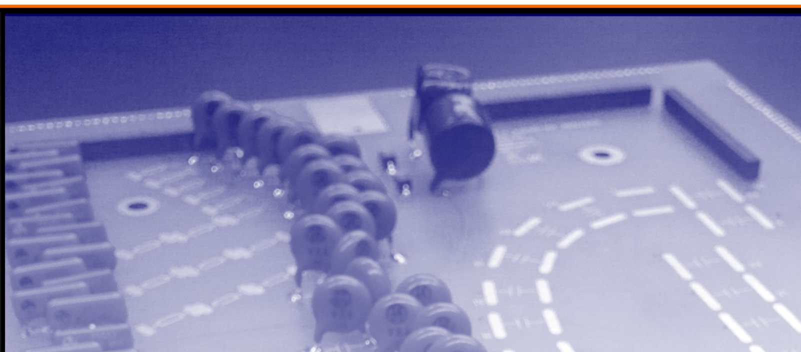
CDT Detector Control

Reference Manual





Notice

This document contains CDT GmbH proprietary information. CDT provides the information and data included in this document for your benefit, but it is not possible to entirely verify and test all information, in all circumstances, particularly information relating to non-CDT manufactured products. CDT makes neither warranty nor represents relating to the quality, content, or adequacy of this information. The information in this document is subject to change without notice. Although every effort has been made to ensure the accuracy of this manual, CDT shall not be liable for any errors, incidental, or consequential damages in connection with the furnishing, performance, or use of this manual or examples herein. CDT assumes no responsibility for damage or loss resulting from the use of this manual, for loss or claims by third parties, which may arise through the use of *CDT Detector Control* or for any damage as a result of malfunction or repair.

March 6, 2015

CDT CASCADE Detector Technologies GmbH
Hans-Bunte-Str. 8-10, 69123 Heidelberg, Germany
Phone: +49 (0) 6221 796 745 1
Fax: +49 (0) 6221 796 745 3
martin.klein@n-cdt.com
www.n-cdt.com

Contents

1 Introduction **1**

1.1 Installation . . . 2

1.2 Getting Started . . . 2

2 Available Measurement Types, Data Display and Data Files **3**

2.1 1D Measurement . . . 3

2.2 2D Measurement . . . 3

2.3 TOF Spectrum, Single Sweep Spectrum . . . 4

2.3.1 MIEZE measurement using TOF Spectrum . . . 4

2.4 Pulse Height Spectrum . . . 7

2.5 CIPix Scan . . . 7

2.6 HV Scan . . . 8

2.7 Data Display and Data Files . . . 10

2.7.1 Scale . . . 10

2.7.2 Status Information within the Data Display . . . 10

2.7.3 Event filter and measurement statistics . . . 10

2.7.4 Data Files . . . 12

3 The Menus **13**

3.1 File and View . . . 13

3.2 Measurement . . . 14

3.2.1 Start . . . 14

3.2.2 Stop . . . 14

3.2.3 Comments... . . . 14

3.2.4 Batch-Job... . . . 15

3.3 Control . . . 17

3.4 Setup . . . 18

3.4.1 Software Setup... . . . 18

3.4.2 Detector Hardware Setup... . . . 20

3.4.3 Re-Initialize Detector Hardware 21
3.5 Window 21
3.6 ? (About) 22

4 The Slow Control 23
4.1 Gas Control 23
4.2 HV Control 24

5 The CASCADE DAQ Systems 27
5.1 CDR / DAQBox Control 28
5.1.1 CDR / DAQBox Control 1 28
5.1.2 CDR / DAQBox Control 2 29
5.1.3 Simulator 30
5.1.4 MIEZE 32
5.2 CIPix Control 33
5.3 IF15 / SU706 Control 35

6 The Toolbars 39
6.1 Standard Bar, Measurement Bar and Config Bar 39
6.1.1 Config Toolbar - Configuration of the Measurement 40
6.2 TOF Control 41
6.3 PHA Control 41
6.4 CIPix Control 41
6.5 HV Scan Control 42

A Detailed Description of the CASCADE Event-Filter Algorithm 43
A.1 The 2D-Pixel Event-Filter Algorithm 44

Chapter 1

Introduction

This manual describes in detail the software program *CDT Detector Control*. This program allows stand alone operation of CDT's neutron detectors from a PC as well as any type of CDT's DAQ systems. The Windows[1] based software package supports easy configuration of the DAQ system, starting and ending data acquisition as well as data download and variable display. It gets the user up and running the system immediately. The program itself is held in the typical Windows-style. Various types of measurements (e.g. 2D-readout, TOF-spectra or MIEZE-spectra) can be configured individually in a self explanatory way. The software provides for all necessary control functions, system health-check as well as configuration tools. Finally, the software can control and monitor the high voltage system, if devices of type NHQ from Iseg (Germany) will be used.

CDT Detector Control itself is consisting on two parts: the graphical user front-end (GUI) and the *CASCADE Hardware Library*. The *CASCADE Hardware Library* supplies all routines for configuration and measurement of the DAQ system as well as all the basic routines for communication. The library is available for Windows and Linux, which supports high level programming under C++ and includes software drivers to integrate the DAQ hardware into already existing instrument control under Windows (XP/7) and Linux (2.4.x-2.6.3). A detailed manual [2] describes the different parts of the *CASCADE Hardware Library*, the C++ class structure and their usage. Additionally, sample programs will demonstrate the use of the library and its functionality. Customer can use the *CASCADE Hardware Library* as base for its own further developments by inheritance of functionality of the library as usually done under C++.

[1] Windows is a registered trademark of Microsoft Corporation.
[2] *CASCADE Hardware Library*, Reference Manual, CDT GmbH

1.1 Installation

An installation of *CDT Detector Control* is not necessary. The software can be executed from any destination, hence it can be used having limited account permissions. Additionally, when running the first time as an administrator, the file type extensions will be assigned to *CDT Detector Control* (DDE[3] mechanism for Windows)
The program consists of a main file DetectorControlxx.exe[4] and additional .dll files, which provide functionalities for controlling the different hardware components. For communication with the detector, the driver for the PCI interface SIS1100 (Struck company), as well as the Cypress USB 2.0 interface driver have to be installed.

1.2 Getting Started

After starting, *CDT Detector Control* tries to get the last saved configuration from the Registry[5]. The information contains settings of the program itself as well as configurations of the connected additional hardware.

> **Caution:** Starting *CDT Detector Control* reads only the last configuration and does not change any settings. Most of the configuration is taken automatically, but certain settings, like for example the values for the high voltage system, have to be set/confirmed manually due to safety reasons.

Then, the pop-up window **New** (see fig. 2.1), which gives an overview about the available measurement types, can be canceld, if no specific choice is already intended to be done. The following steps we propose, to get a running system and to check its functionality:

- Configuration of the software setup (see 3.4.1): mainly the intended directory for the data output files.
- Configuration of the detector/DAQ hardware setup (see 3.4.2): initialize the DAQ system to be used. Please assure, that the hardware has been connected to the PC and that the relevant driver has been installed (e. g. the CASCADE DAQBox driver for communication via USB).
- Test of communication can be done as described under section 5.1, when opening the **Control** dialogues. The displays for **Firmware Version and Setup** as well as **Status Register** will show the actual values read out from the hardware. A more intensive test can performed through testing the memory sitting on the DAQ system. For this purpose the button **Test SRAM** can be used (to be found on the second tab as described under 5.1.2).
- Test of the entire Detector-DAQ-System by performing a scan of the preamplifiers discrimination thresholds and therefore one of the detectors most relevant parameters. Detailed description can be found under 2.5.

[3] Dynamic Data Exchange
[4] xx represents the version number: e. g. 96 = Version 9.6
[5] All configuration files are saved in the user section of the Registry at: HKEY _CURRENT_USER\Software\CDT\CDT Detector Control\ Version xx .

Chapter 2

Available Measurement Types, Data Display and Data Files

After executing *CDT Detector Control* the pop-up window **New** (see fig. 2.1) gives an overview about the available measurement types. By selecting a certain item and confirming by pressing the 'OK' button, a new document specific to the chosen measurement type is opened.

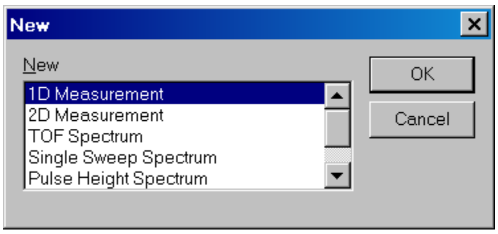


Figure 2.1: Dialog **New**

2.1 1D Measurement

1D Measurement is not applicable to CASCADE 2D detectors.

2.2 2D Measurement

2D Measurement allows taking a two-dimensional image of the signals accumulated within a chosen measurement time. These data will be shown in the data display as a 2D matrix of N x N pixels. The resolution can be set from 1 x 1 pixels up to 128 x 128 pixels, whereas lower resolutions than physically given show the cumulated values of the combined pixels. The coordiante system starts at the lower left corner with pixel (0,0) (see fig. 2.2).
The display options can be set by the **scale** menu (see 2.7.1), which is available by right-clicking on the display. It is possible to choose between a linear and a logarithmic scale with either a gray scaled or a color scheme. The minimum and maximum values can be set manually as well. The default display shows the measured absolute count values in a linear, automatically adjusted scale.

Information of the current settings and actual signal shape is presented in the right part of the display window (see 2.7.2). The header shows filename, total counts, the actual measurement time and the count rate, dividing both values.

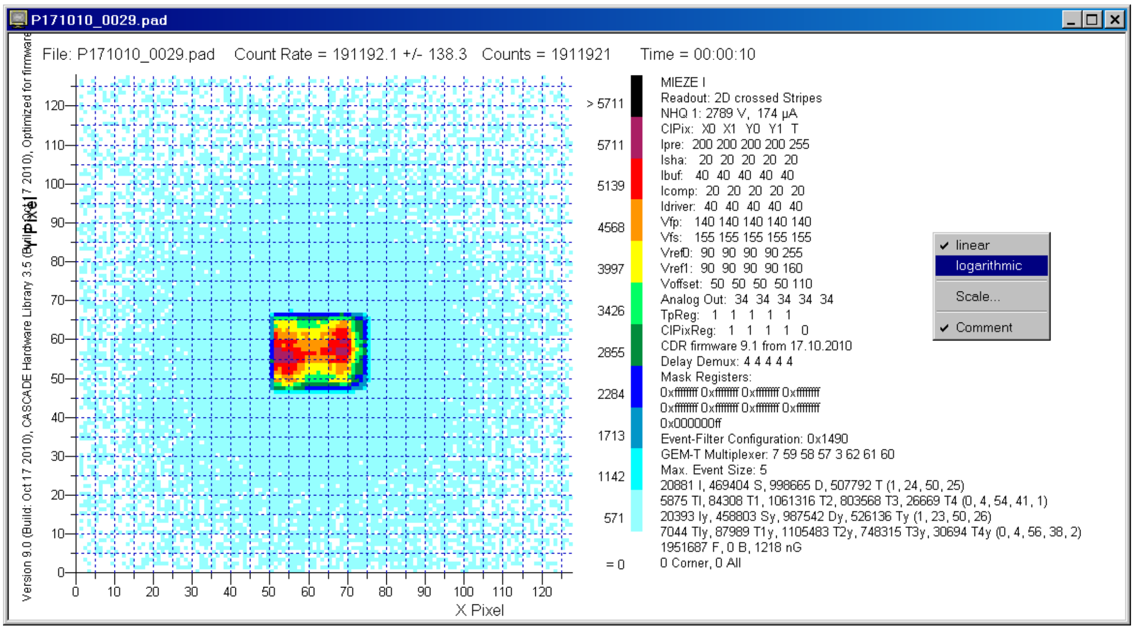


Figure 2.2: **2D Measurement**: Data display in rainbow color scheme (left), status and settings and scale menu after right-click (right).

2.3 TOF Spectrum, Single Sweep Spectrum

TOF Spectrum allows taking a time resolved measurement (see fig. 2.3). It shows the number of counts integrated over each bin and **Sweep** cycle as a function of the time of flight channels. Each bin has a length configured by the **Dwell Time**, which is set in units of 100 ns (see 6.2). Accordingly, the time base is set to zero for every Sweep cycle by a start signal. It is possible to measure for a fixed number of Sweep cycles or a fixed time. Fixed sweeps will always override fixed time.
When pressing the **Stop** button the measurement will not stop until the actual sweep is finished. This assures to get a complete data set. However, to abort the actual cycle there is an additional **Stop** button in the **TOF Control** Toolbar (see 6.2).

Single Sweep Spectrum is a subtype of **TOF Spectrum** in which, instead of integrating over the cycles, every single sweep will be recorded individually. The whole set will be shown in a large TOF display. Every single sweep will be saved as well as a sum of all.

Information of the current settings and actual signal shape is presented in the right part of the display window (see 2.7.2). The header shows filename, sweep cycles, total counts, the actual measurement time and the count rate, dividing both values.

2.3.1 MIEZE measurement using TOF Spectrum

For MIEZE experiments it is essential to measure the fast oscillating neutron intensity. This requires a high time of flight resolution of the detector system used. In addition, the neutron intensity is not only modulated in time but also in space along the propagation direction of the neutron beam. Therefore the point of neutron absorption needs to be defined very precisely in order to avoid contrast damping due to averaging.

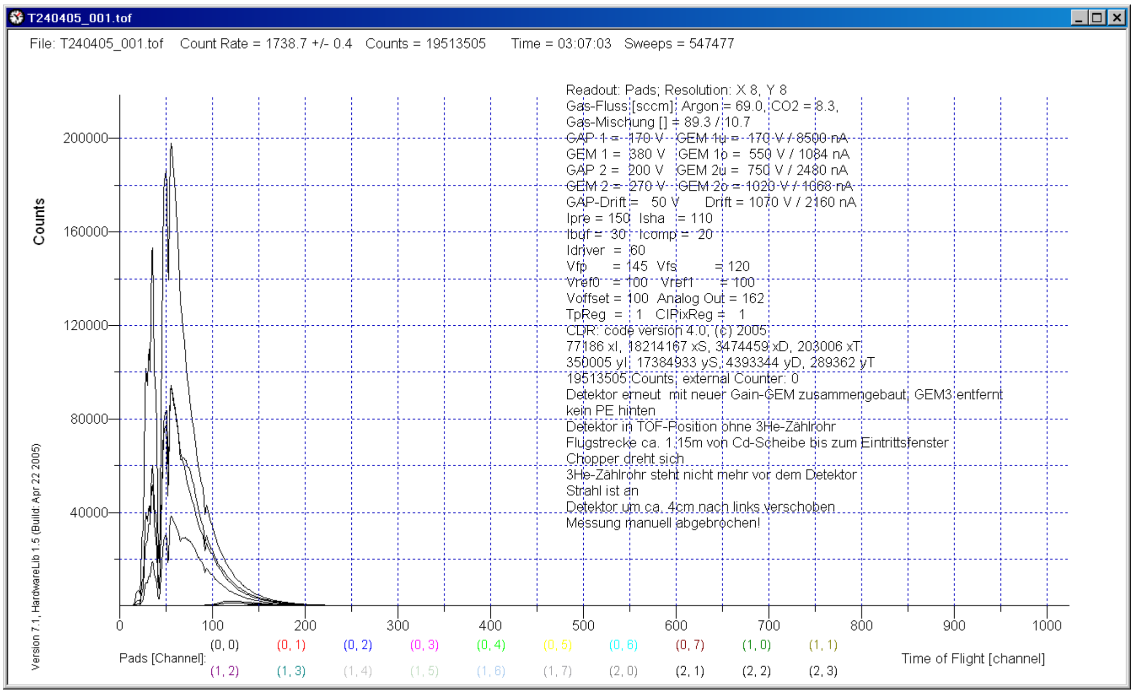


Figure 2.3: **TOF Spectrum**: Data display of counts as function of time for selected channels (legend at the bottom) and all channels summed up

As a consequence for the CASCADE-MIEZE detectors it is necessary to determine the particular boron layer in which the neutron was absorbed. For this purpose the detector is equipped with additional readout channels (CIPix T) sensing the mirror charge induced on each GEM if the cloud of charges passes through its holes. Its data are evaluated in parallel to identify in which boron layer **n** the absorption of the neutron took place. So finally complete TOF spectra (**x, y, t**) for each boron layer **n** and each pixel (**x, y**) of the readout structure are generated and stored within the local memory of the DAQ system (**x, y, t, n**).
When using **TOF Spectrum** to measure a MIEZE spectrum, the MIEZE frequency f_{MIEZE} is used to trigger inside the FPGA the internal sweep refresh similar to a chopper pulse. But the time stamp **t** is not derived as usual from the DAQ's internal clock oscillator. To guarantee, that the entire detector electronic runs phase synchronous to the neutron oscillation frequency, the time stamp **t** is now derived from a phase-locked-loop (PLL) circuit controlled by the MIEZE frequency f_{MIEZE}. With the 16-times higher frequency output of the PLL the internal time bin counter is triggered (needs option **External Time Bin Mode**, see fig. 5.5).

Figure 2.4 shows a measured MIEZE spectrum from a CASCADE-MIEZE detector equipped with four boron layers. For each boron layer a complete TOF spectrum has been measured with 16 TOF channels. The sorting order of the boron layers (as marked in fig. 2.4, listed in 5.5 and shown in fig. 5.6) starts with the bottom half of the detector with GEM I, which is directly located beneath the readout structure, followed by GEM II, and so on. The second half of the TOF spectra (right side) represents the top half of the detector starting with GEM 1, which is directly located above the readout structure, followed by GEM 2, and so on up to the neutron entrance window of the detector. Due to the actual design of the PLL, which is used to drive the time stamp generation inside the FPGA from external and therefore phase synchronous, in reality 17 TOF channels are filled up with counts. This design allows a jitter-free measurement and the first and the 17th channel have to be summed up after measurement to get finally the jitter-free counts of the first channel.

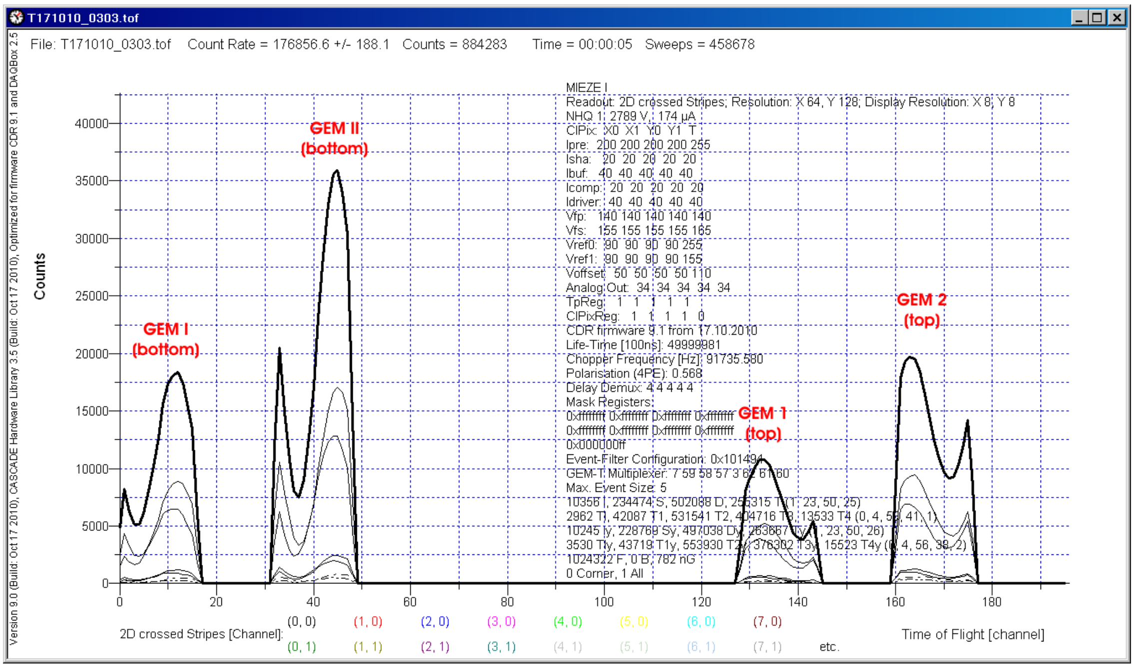


Figure 2.4: MIEZE signal of a detector with four layers measured using measurement type **TOF Spectrum**

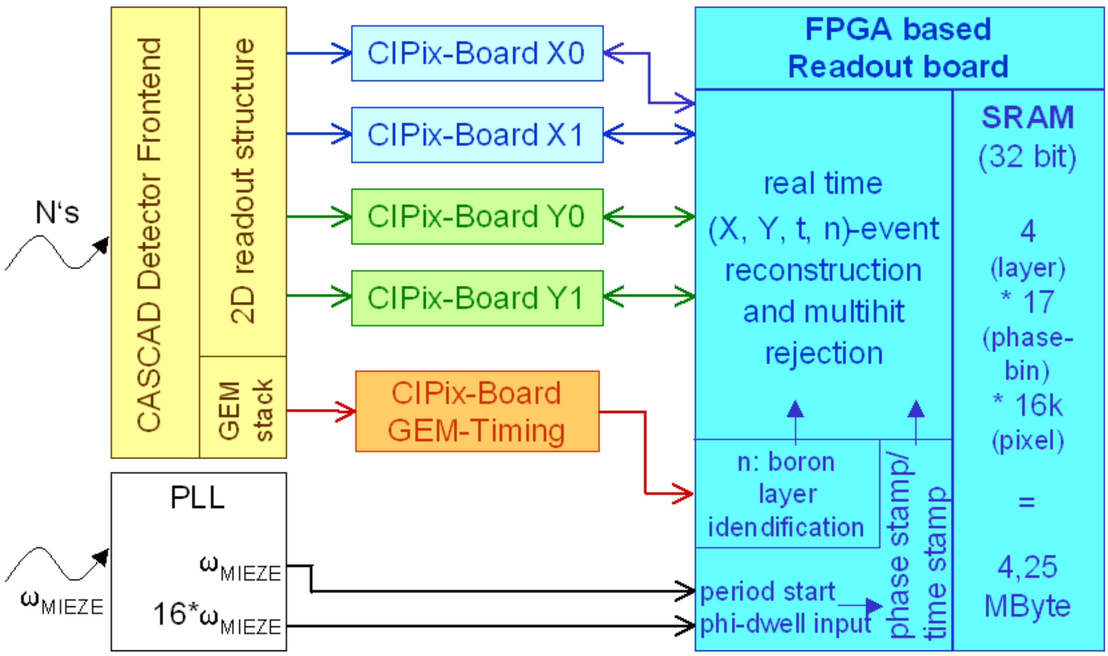


Figure 2.5: Schematic of the entire DAQ system of a CASCADE-MIEZE detector

2.4 Pulse Height Spectrum

By using the **Pulse Height Spectrum** mode it is possible to measure the intensity of signals within the detector determined by the internal ADC. It shows the histogram of peak heights.
It is possible to change the resolution (number of channels) by the **PHA Control** in the drop-down-menu **Resolution of ADC** (see 6.3). The histogram will always show the full range of 0 V up to the maximum possible voltage of the ADC, so it is only possible to change the binning.

Information of the current settings and actual signal shape is presented within the display window (see 2.7.2). The header shows filename, total counts, the actual measurement time and the count rate, dividing both values. 'Under' and 'Over' give the number of measured signals below and beyond the displayed range.

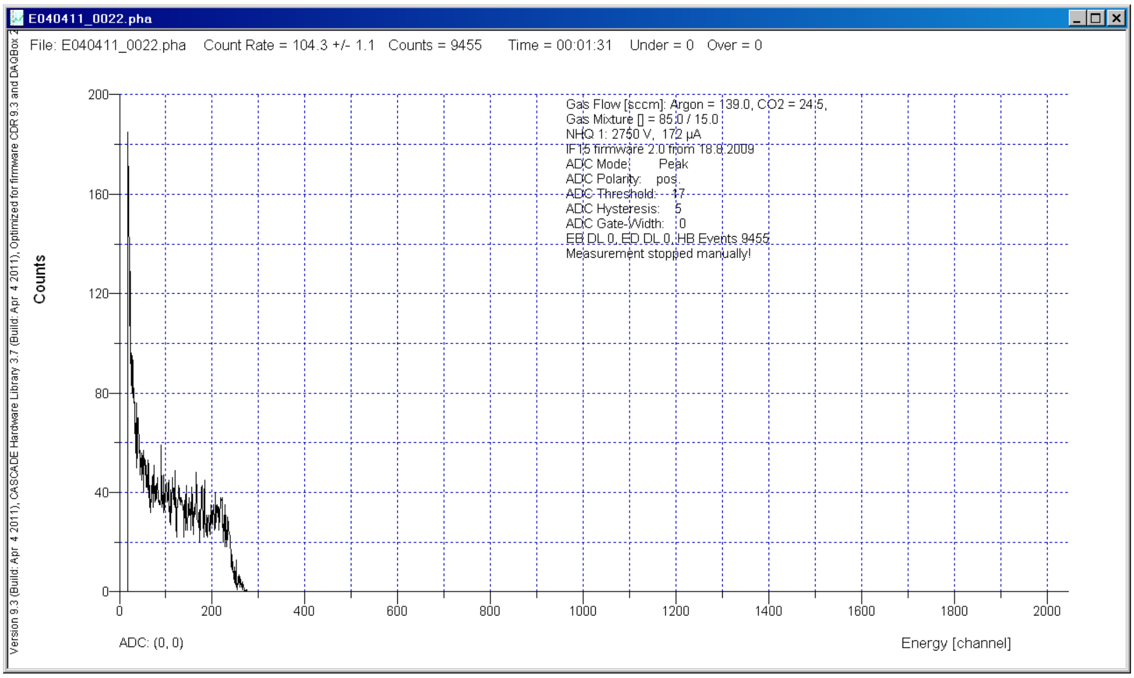


Figure 2.6: **Pulse Height Spectrum**

2.5 CIPix Scan

CIPix Scan can be used to determine the CIPIX operation parameters of the detector, like for example the discrimination threshold (*Vref0* and *Vref1*), which is the most relevant parameter. It shows the counts integrated over a given time as a function of the selected parameter (DAC). After setting the measurement time for a single DAC value, the software will perform a run counting the signals for each value within the range. To set up a CIPix Scan run, the **CIPix Control** (see 6.4) is used. Therefore the following parameters have to be declared: CIPIX board (*X0, X1, Y0, Y1, T, All*), lower and upper DAC value, DAC incrementation step width and name of the DAC (for possible DACs see 5.2). Note: the DAC settings will not keep their initial values, but stay at the value at which the CIPix Scan ended or stopped.

Information of the current settings and actual signal shape is presented within the display window (see 2.7.2). The header shows filename, total counts, the actual measurement time and the count rate, dividing both values.

Example: Threshold Scan (see fig. 2.7)

To operate the detector with its maximum sensitivity it is necessary to set the DAC value for the discrimination threshold (Vref1) just above the level of electronic noise. This 8-bit DAC can be adjusted from -200 mV to +200 mV corresponding to values of 0 to 255 with 0 mV being equal to 128. The run in figure 2.7 shows the noise level starting between 112 and 115. Note: for 2D detectors with readout stripes signals will be measured by a correlation of the X and Y lines, which suppresses the single channel noise.

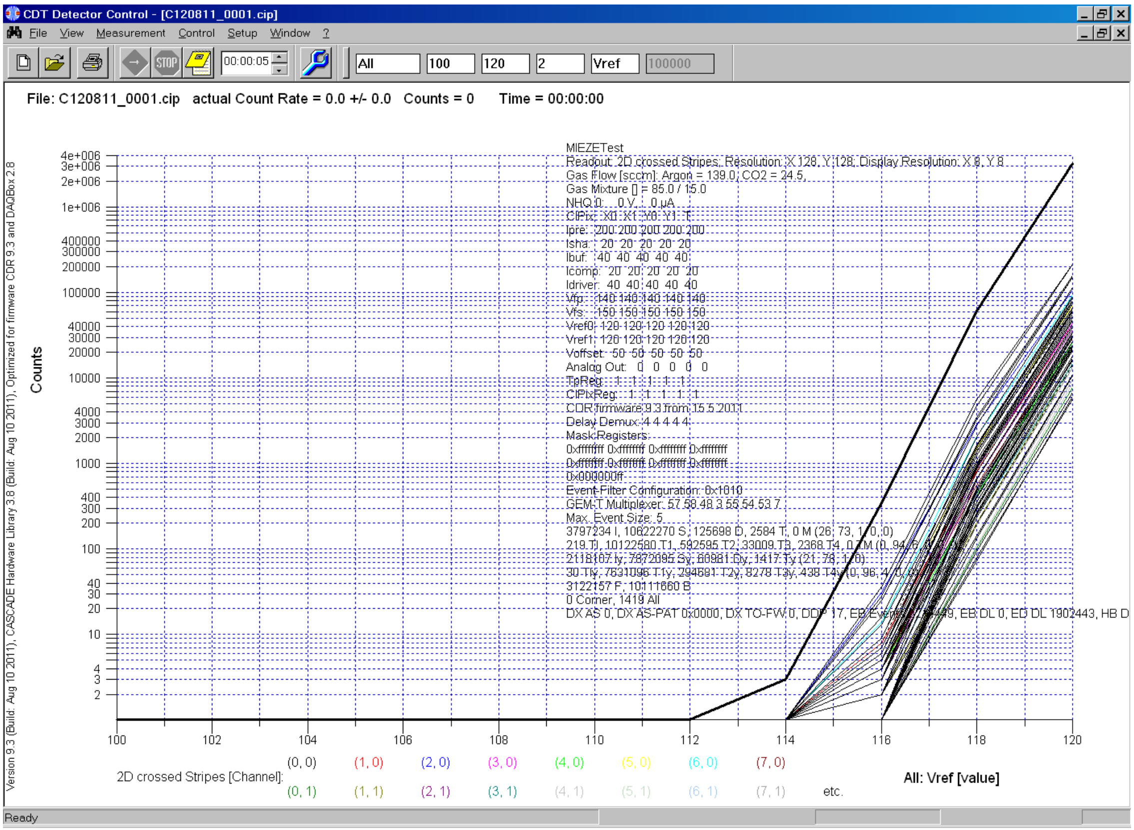


Figure 2.7: **CIPix Scan**: Data display of counts as function of DAC value for selected channels (legend at the bottom) and all channels summed up

2.6 HV Scan

HV Scan can be used to determine the optimal operating voltage (HV) of the detector. It shows the counts integrated over a given time as a function of the voltage at the whole stack of GEMs. After setting the measurement time for a single HV value, the software will perform a run counting the signals for each value within the range. To set up a HV Scan run, the **HV Control** (see 6.5) is used.

Information of the current settings and actual signal shape is presented within the display window (see 2.7.2). The header shows filename, total counts, the actual measurement time and the count rate, dividing both values.

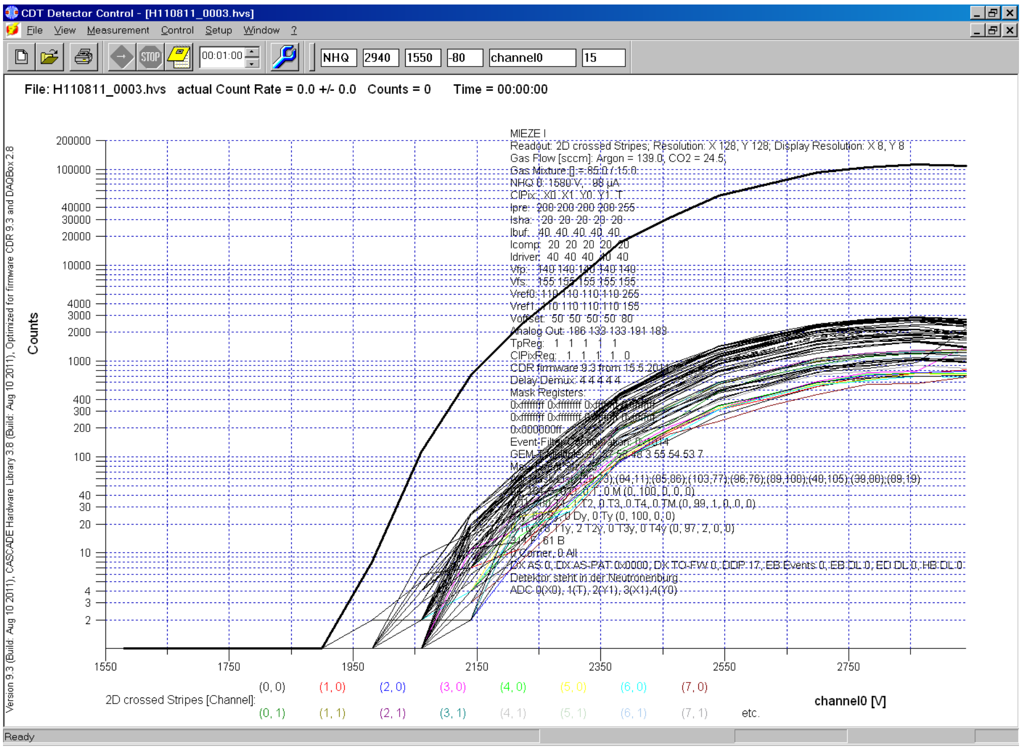


Figure 2.8: **HV Scan** with linear scaling

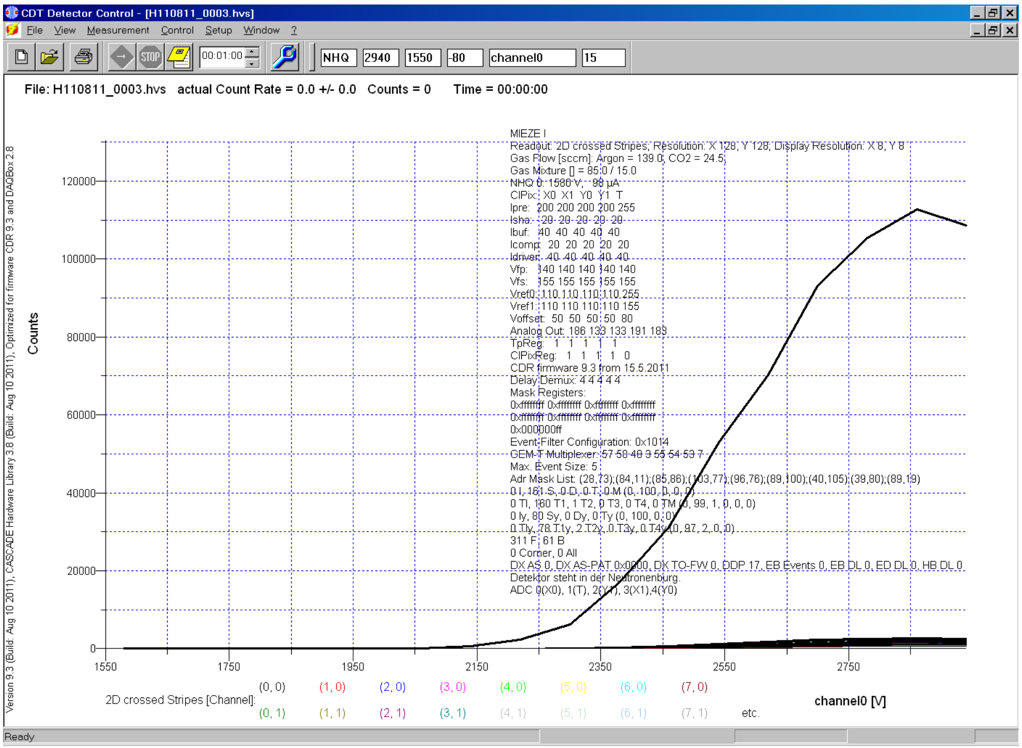


Figure 2.9: **HV Scan** with logarithmic scaling

2.7 Data Display and Data Files

CDT Detector control offers different types of displaying the detector output. It either shows either measured events as a function of one parameter (see 2.3) or a 2D X/Y display (see 2.2). All 1D displays are due to visual separability of line plot type. The standard output for 2D displays shows the absolute counts in an automatically scaled colour scheme. The display options can be modified by right-clicking on any position within the window opening the dialogue **Scale** (see 2.7.1).

2.7.1 Scale

By right-clicking on the data display and selecting **Scale** (see fig. 2.10) it is possible to modify the display options. All changes can be made during a running measurement.

1. Section **Colour:** colour scheme (12 values) or gray scale (256 values)
2. Section **Type:** linear or logarithmic scaling
3. Section **Scale:** automatic or manual scaling of the axis. For 2D displays it is not possible to change the X or Y axis, instead the Z axis scaling can be modified at the Y axis position fields

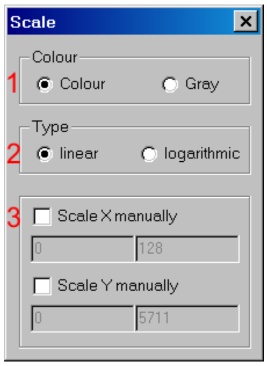


Figure 2.10: Dialogue **Scale**

2.7.2 Status Information within the Data Display

The right part of every data display window shows the status information of the current measurement. All status values, which are only shown partly during the actual measurement, are as well saved in the so-called *Settings-File* (see 2.7.4). The following information is displayed:

- name of the detector
- configuration of the readout channels and resolution
- slow control configuration for the gas flow and high voltage power supply
- CIPix parameter for the thresholds *Vref0* and *Vref1* (other DAC settings are saved in the settings file)
- firmware version
- event filter configuration and real-time statistics

2.7.3 Event filter and measurement statistics

The CASCADE DAQ firmware contains a real-time event filter algorithm. These settings can be accessed via the CDR Control panel, see 5.1.1.
The event filter suppresses fake events coming from sparks or firing pixels.
The measurement statistics display (see fig. 2.11) is a powerful tool to configure a CASCADE detector. The real-time data analysis shows statistics about the identified events. For configuring the detector for the first time, it can be used to optimize the settings by tuning these until 'typical' values are reached. As

well it characterizes the space and time structure of signals by counting the clock cycles an event can be measured within the detector. If the gas gain is too high, events will have a broad spacial width and will cause huge and long signals at the GEM layers. Too high rates will cause increasing numbers of invalid events, which fail in the X-Y(-T) reconstruction.

```
211 I, 34223 S, 69153 D, 46371 T (0, 23, 46, 31)
8 TI, 6126 T1, 64397 T2, 77903 T3, 1321 T4 (0, 4, 43, 52, 1)
202 Iy, 35439 Sy, 69094 Dy, 45850 Ty (0, 24, 46, 30)
7 TIy, 6080 T1y, 58115 T2y, 84849 T3y, 1339 T4y (0, 4, 39, 56, 1)
141334 F, 282 B
0 Corner, 0 All
```

Figure 2.11: Event statistics display during a measurement

In detail, the given information, see as well fig. 2.11, contains information about the spatial width (**I** = invalid, **Si** = single = 1 channel, **Dbl** = double = 2 channels, **Tr** = triple = 3 channels, **Mu** = multiple = more than 3 channels) and the number of clock cycles (**Tn** = n clock cycles of 100 ns, **TI** = invalid):

- Spatial distribution of signals for the X projection (stripe readout) or all pixels (single pixel readout). The 'Invalid' tag contains events which are excluded by the event filter or which are larger than the maximum size (see 5.1.1)
- Time distribution of signals for the X projection (stripe readout) or all pixels (single pixel readout). The 'Invalid' tag contains events which are too long and fail reconstruction.
- Spatial distribution of signals for the Y projection (stripe readout).
- Time distribution of signals for the Y projection (stripe readout).
- (MIEZE mode and GEM readout filter enabled) **F**: number of reconstructed events in correlation of X,Y and T. **B**: Number of events, which fail correlation with T or cannot be reconstructed.
- Event filter EMI Protection. Events which are suppressed in the corners by the **corner** analysis and on the full detector area (**All**)

All values are given in percental and absolute values (in brackets).

Any real neutron event will reveal itself as a signal in one or more neighbouring channels and over one or more connected time bins. Thus, in a second step towards identifying neutron events, this entity does an analysis in the time and space domain to reduce any spot of distributed but connected signals to one unique detector event. In the case of ambiguities as for example due to simultaneous disconnected multi-hits the data slice is rejected. In case of readout structures based on readout stripes, acceptance of filter algorithm can be configured for signals which are single stripe, 2 neighbouring stripes or up to 7 neighbouring stripes and signals up to 4 time cycles long (400 ns). Two spots of connected signals can be identified as two individual neutron events as long as they do not begin within the same time slice. Ambiguous signals or spots in the time-space domain, which cannot be assigned to an individual neutron event, are filtered out. In case of readout structures based on individual readout pixel, the filter algorithm will accept signals which are single pixel, 2 neighbouring pixel or up to 3 neighbouring pixel with one common corner and signals again up to 4 time cycles long (400 ns). A set of counters is used to monitor filter activity by counting signals successfully reduced to neutron events, signals that could not be correlated to the other coordinate or such that had to be filtered out as the signal spot occurred too broad in the space or time domain to be attributed to a single neutron. These counters provide an online analysis on the statistics of signals in space and time, that occur in the detector. Optionally, signal filtering and reduction may be switched off entirely, so that the raw data are transferred for histogramming.

2.7.4 Data Files

Every measurement will be saved in a file with an automatically generated name in the folder configured in **Software Setup...** (see 3.4.1). The filename contains the type of measurement (by the first letter and suffix E+.pha for pulse height spectra, T+.tof for time-of-flight, P+.pad for 2D, C+.cip for CIPix scan), the date (TTMMYYYY) and a four-digit consecutive number, like for example E031209_0033.pha. The numbering is individual for every folder and is not counting the general number of the measurement. The type of encoding is either ASCII or binary and can be configured in the **Software Setup...** dialogue.

Besides the file for the actual measurement there are additional files with the same prefix:

- *.txt: *Settings-File* (ASCII) containing all settings, status information, hardware and software report codes, as well as user comments.
  This file is essential for opening data files using the CDT Detector Control software as it contains the information for interpreting the data values.
- *.log (optional): *Log-File* (ASCII) containing a report about the values of the slow control which were enabled in **Monitoring and Controlling** in the **Detector Hardware Setup...** dialogue (e. g. see fig. 3.12).
- *.lst (optional): *List-Mode Data File* (ASCII or binary) containing the zero-suppressed RAW data after every clock cycle directly from the CIPix boards prior to the event reconstruction. Zero-suppression is configured not to record clock cycles without signal at any CIPix. The RAW data is structured tab separated in the following columns:

  1. 2 columns for the absolute time stamp (1st: high bits 2nd: low bits) in clock cycles of 100 ns starting with 0
  2. 4 columns for the CIPix boards X0 (lower channels), X1 (upper channels), Y0 (lower channels) and Y1 (upper channels)
  3. T-GEM CIPix

  The CIPix channel binary information ('1' for signal above threshold, '0' for no signal) is saved hexidecimally by grouping 4 channels and converting the bit pattern to a HEX value. Accordingly the T-GEM CIPix is encoded, whereas the lowest bit is identified with the GEM next to the readout (GEM 1) and the highest with the outermost. The left HEX value contains the information of the bottom half and right of the upper half of the detector, respectively.

Finally, the log file History.txt contains chronologically the content of all *Settings-Files* (*.txt).

Chapter 3

The Menus

Menu items containing '**...**' will open a sub-menu, items without '**...**' execute directly.
Additional information is provided by tooltips, which open by hovering over with the cursor (see as well 6), and the information bar at the bottom of the window shows detailed help descriptions.

3.1 File and View

File opens the standard menu as used to be under Windows with the options to create a new measurement document, to open, to close or to print a measurement document (s. fig. 3.1). **View** opens the standard menu as used to be under Windows with the options to enable or disable the various toolbars and the status bar (see fig. 3.2).

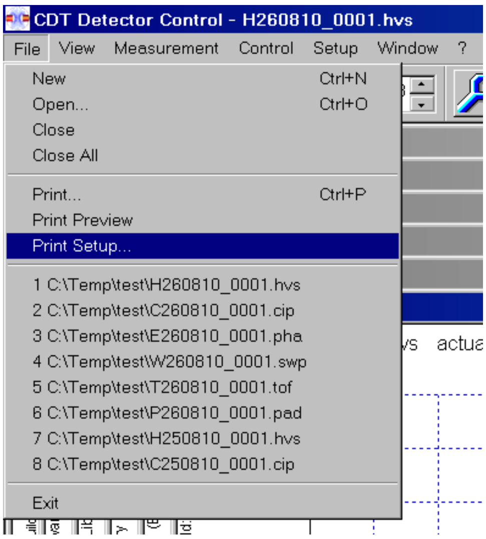


Figure 3.1: Menu **File**

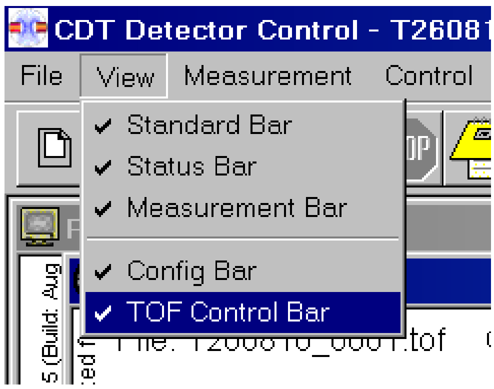


Figure 3.2: Menu **View**

3.2 Measurement

Measurement (see fig. 3.3) allows controlling starting and stopping of a measurement. The toolbar contains equivalent buttons for **Start**, **Stop** and **Comments** (see 6.1).

3.2.1 Start

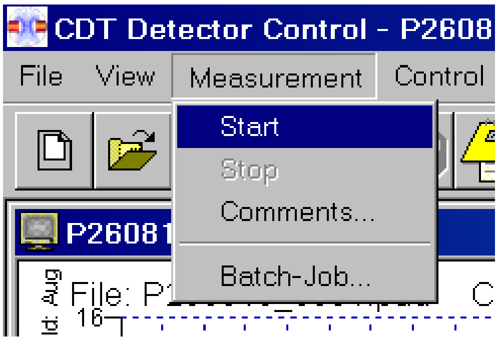


Figure 3.3: Menu **Measurement**

Start initializes a single measurement in the active window. Before executing a measurement the software checks if the settings are consistent concerning the chosen type of measurement and the allocatable hardware. Any errors are shown in a pop-up window.
As long as a measurement is running (the title bar shows the additional phrase 'active') or is finished (the title bar shows the additional phrase 'finished') **Start** is disabled.

3.2.2 Stop

Stop stops the running measurement of the active window. Only as long as a measurement is running (the title bar shows the additional phrase 'active') **Stop** is enable.

3.2.3 Comments...

Comments... opens a pop-up window to add a text phrase to a measurement. Only if a measurement is active, local comments are accepted. There is the possibility to set local (only for the active window) and global (applied to every measurement) comments.

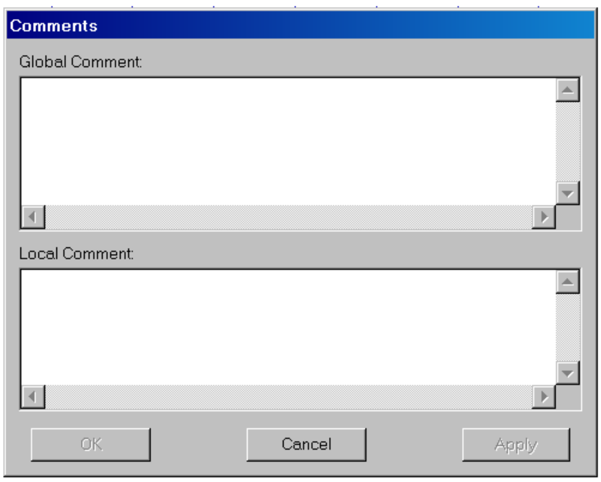


Figure 3.4: Menu **Comments**

3.2.4 Batch-Job...

Batch-Job... opens a pop-up window to enter a batch job containing an arbitrary series of measurements.

1. Window for editing the batch job.
2. **Take Changes:** Changes of the batch job list will automatically applied by clicking **Save** and **Start**. To edit a running batch job changes can be explicitly be taken into the actual series by using the button **Take Changes**. This will but only effect pending measurements.
3. **Example:** Inserts an example batch job explaining syntax and possible key words.
4. **close old documents automatically:** Long batch jobs can contain hundreds of single measurements, all opening in an individual window allocating memory. To reduce the amount of RAM used, finished measurements can be closed automatically.
5. **External control via RS-232:** Jobs can be submitted via RS-232 as well. This section allows activating and configuring the serial interface. **Time Out Factor** sets the time before an internal timeout is triggered by missing response of the serial interface.
6. **Name of the batch job file:** Name of the ASCII file (*.job), which contains the job list.
7. **Start:** Start of the job processing.
8. **Stop:** Stop of the job processing. The actual running measurement will not be stopped and completed. d

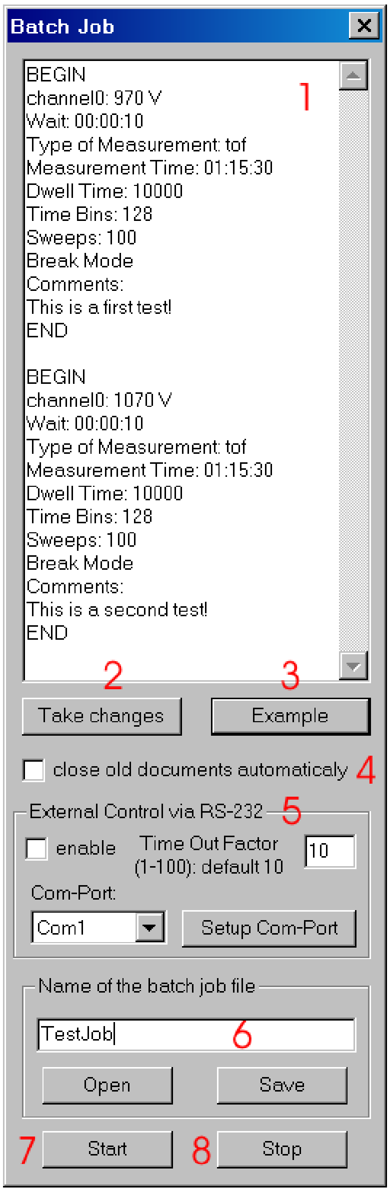


Figure 3.5: Dialogue **Batch Job**

Every individual measurement within a batch job is marked by the key words `BEGIN` and `END` (see fig. 3.5). A list of all possible key words is shown in table 3.1. Space delimiters have to be set correctly. Parameters, which do not change from one measurement to the following, do not have to be repeated. The only parameter to be defined in every single job is `Measurement Time`. Comments in round bracket will be ignored.

Fig. 3.6 shows a simple example for a batch job. At the beginning a 2D measurement (pad) will be started, lasting 1h 15min 30sec. After waiting 15 seconds, specified by `Wait`, a second measurement in TOF mode (tof) will be carried out.

Parameters concerning the slow control, like for example gas flow and high voltage values, and settings of the DAQ electronics, like the CIPix parameters, have to be declared prior to the `Wait` statement. Parameters configuring measurements, like those set by the toolbar (see chapter 6), have to be declared after the `Measurement Time` keyword.

```
BEGIN
Type of Measurement: pad
Measurement Time: 01:15:30
END

BEGIN
Wait: 00:00:15
Type of Measurement: tof
Measurement Time: 00:15:30
END
```

Figure 3.6: Example of a batch job consisting of two jobs.

Group	Key Word and Syntax	Description
	BEGIN	Begin of job definition
Slow-Control	Gas A: 70% Gas B: 30% Sum flow: 50 sccm	fraction of gas A in % fraction of gas B in % total gas flow in units of sccm[1]
	channel0: 970 V channel0: 521 V	absolute high voltage at channel 0 absolute high voltage at channel 1
	Isel x: 1200 Isel y: 40 Isel z: 2900	X axis will be shifted an amount of 1200 steps Y axis will be shifted an amount of 1200 steps Z axis will be shifted an amount of 2900 steps
	Wait: 01:10:33	waiting time in HH:MM:SS
	Type of Measurement: pad Type: pad	measurement type: ps1, pad, tof, swp, pha, cip, hvs alternative declaration
	Measurement Time: 01:15:30 Time: 100	measurement time in HH:MM:SS alternative declaration in seconds
	Loop: 10	looping a job 10 times
TOF	Dwell Time: 10000 Time Bins: 128 Sweeps: 100 Break Mode Sweep Mode	in units of 100 ns number of Time Bins number of sweeps to be measured Break Mode Sweep Mode
PHA	ADC Resolution: 128 ADC 0	resolution of the ADC ADC 0 used
CIPix Scan	CIPix: All Register: Isha CIPix Start: 0 CIPix End: 120 CIPix Delta: 10	CIPix to be scanned: All, X0, X1, Y0, Y1 register to be scanned start value: 0 - 255 end value: 0 - 255 step width: 1 - 255
	HV Device: NHQ HV Start: 0	device to be scanned start value [V]

[1] standard cubic centimetre per minute

Group	Key Word and Syntax	Description
HV SCAN	HV End: 120	end value [V]
	HV Delta: 10	step width [V]
	HV Parameter: channel0	HV channel to be scanned
	HV Pause: 100	pause between two positions within a scan [s]
	Comments: Hello world	comment
	END	end of job definition

Table 3.1: Available key words for batch job definitions. The names and the usage of many key words is identic to the names and meaning of the control elements in the corresponding toolbars (see chapter 6).

3.3 Control

Control allows configuring the initialized hardware (see fig. 3.7).
Submenu items are only available if the corresponding hardware is already configured via **Setup** → **Detector Hardware Setup...**, see also 3.4.2.

It is possible to configure the slow control (see section 4), including gas flow control (**Gas System...**), high voltage management (**HV System...**) and the Isel stepper motor control.

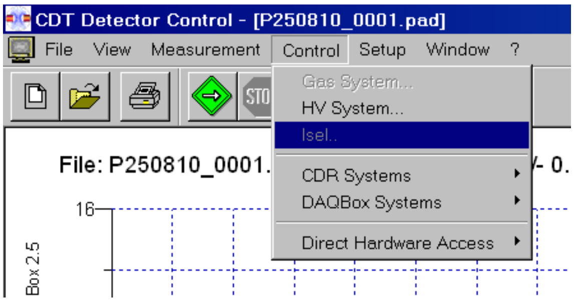


Figure 3.7: Menu **Control**

CDR Systems and **DAQBox Systems** allow configuring both types of CASCADE DAQ systems. Both are described in detail in 5.1.

3.4 Setup

Setup allows configuring the CDT Detector Control software (**Software Setup...**) and the entire hardware components (**Detector Hardware Setup...**).

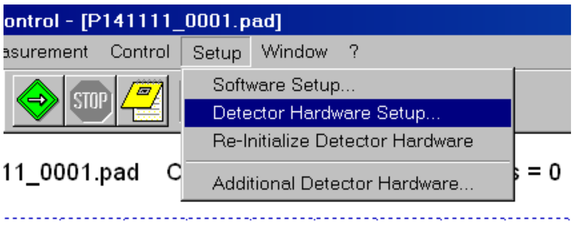


Figure 3.8: Menu **Setup**

3.4.1 Software Setup...

Software Setup... opens the dialogue **Software Setup** (see fig. 3.9), which allows configuring the CDT Detector Control software.

1. **Directory:** location for the data output files.
2. **During a measurement:**
   - A **Refresh Cycle** defines the measurement display refresh time [s].
   - A **Check Cycle** defines the period of time [**Refresh Cycle**] to check the actual hardware return values comparing them to the set values. A log file (*.log) keeps track of the data.
3. **Data-File Format:** ASCII or binary formatting for the data and List-Mode files.
4. **List-Mode:** ASCII or binary formatting of the zero suppressed raw data files.
5. **Sum-Curve:** Adds an additional curve to the CIPix scan plot summing all individual channels. In order to increase the statistics this will integrate over the whole detector area.
6. **Advanced menus:** Activates additional menu items for special purposes like the configuration of the MIEZE mode.
7. **DAQ systems supported:** Number of simultaneous CDR DAQ systems, CASCADE DAQ boxes and simulated hardware items.

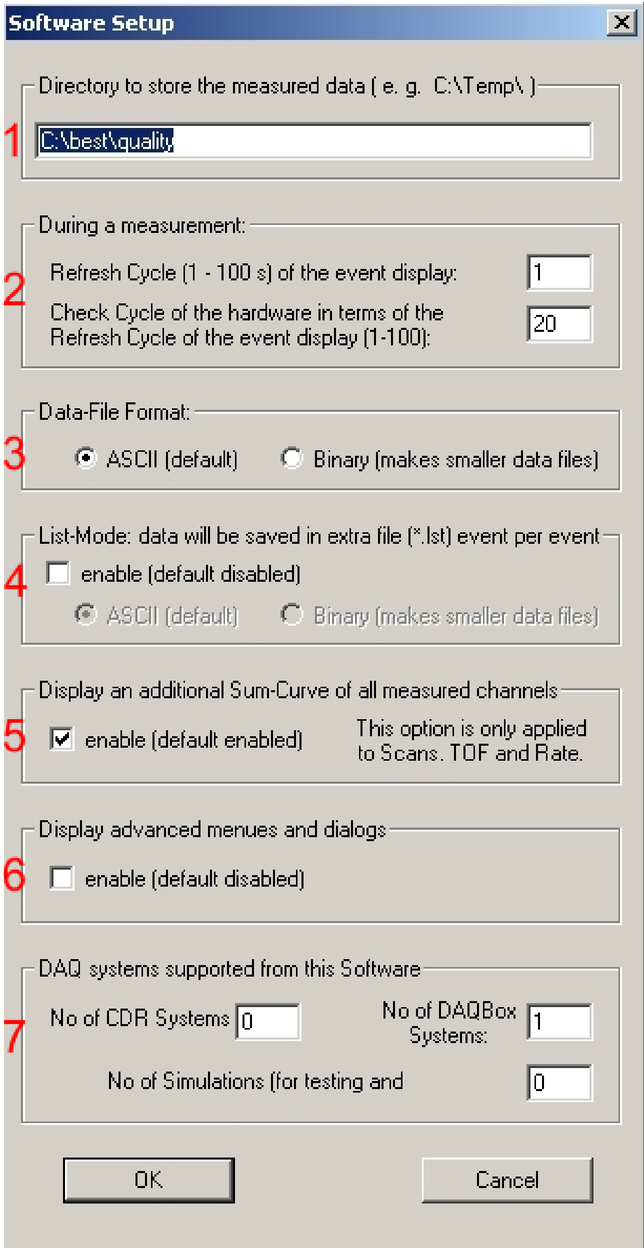


Figure 3.9: Dialog **Software Setup**

3.4.2 Detector Hardware Setup...

Detector Hardware Setup... opens the dialogue **Setup and Initialize the Detector Hardware** (see fig. 3.10), which allows initializing all hardware components (DAQ electronics, high voltage, gas flow control, ...). Depending on the configured hardware (see section 3.4.1, option **DAQ systems supported**), every item will be shown in a separate tab. For every successfully initialized component, the corresponding sub menu item of **Control** (see 3.3) will be activated.

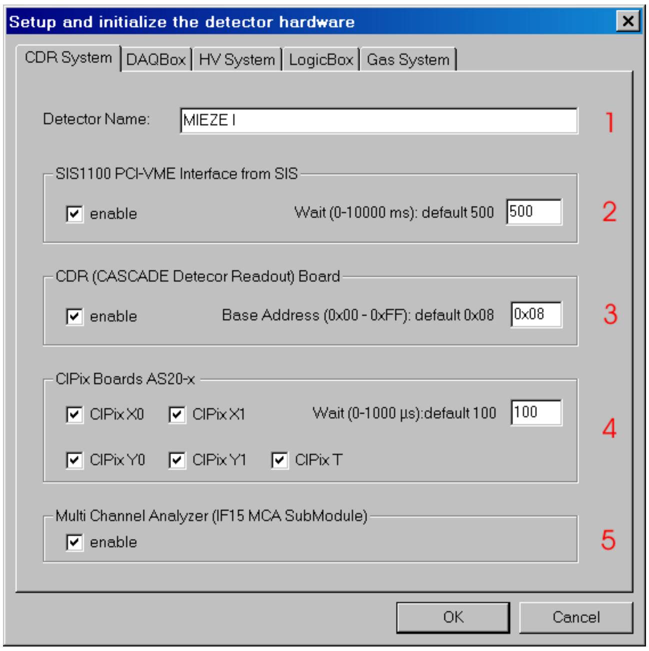


Figure 3.10: Dialogue **Hardware Setup**, Tab **CDR System**

CDR System

1. **Detector Name:** Freely selectable name of the detector, which will be shown in all data displays.
2. **SIS1100 PCI-VME Interface from SIS:** Data transfer between **CDR Board** and computer is handled via an optical GBit interface, which can be activated by this statement. **Wait** defines the hardware fail timeout.
3. **CDR (CASCADE Detecor Readout) Board:** Initializing of the FPGA based DAQ electronics **CDR Board**, which is at default on address 0x08.
4. **CIPix Boards AS20-x:** Initializing of the CIPix preamplifier ASICs to be used. **Wait** defines the hardware fail timeout.
5. **Multi Channel Analyzer (IF15 MCA SubModule):** Initializing of the MCA module **IF15**, which allows measuring pulse height spectra.

DAQBox

1. **Detector Name:** Freely selectable name of the detector, which will be shown in all data displays.
2. **USB 2.0 Interface (FX2 Cypress CY7C68013):** Data transfer between the **CASCADE DAQBox** and the computer is handled via a USB 2.0 interface, which can be activated by this statement. **Wait** defines the hardware fail timeout.
3. **CIPix Boards AS20-x:** Initializing of the CIPix preamplifier ASICs to be used. **Wait** defines the hardware fail timeout.
4. **Multi Channel Analyzer (SU706 MCA SubModule):** InInitializing of the MCA module **SU706**, which allows measuring pulse height spectra.

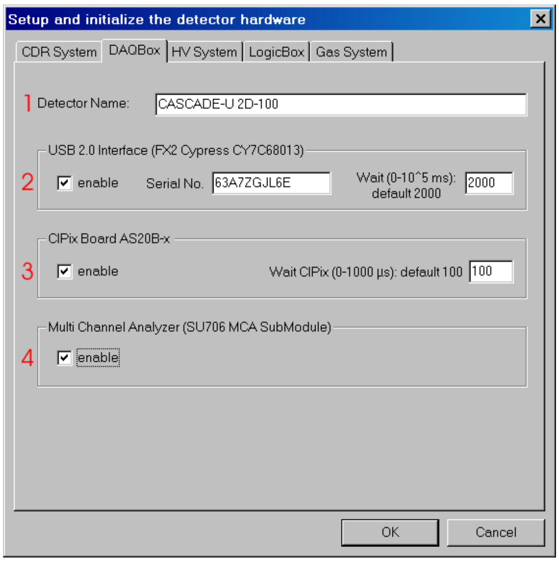


Figure 3.11: Dialogue **Hardware Setup**, tab **DAQBox**

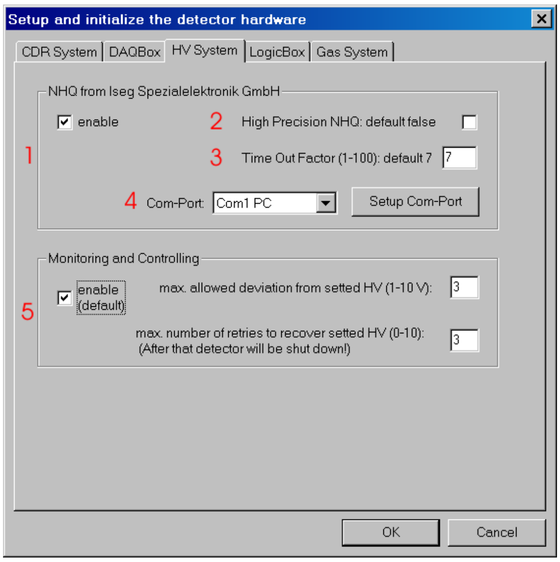


Figure 3.12: Dialogue **Hardware Setup**, tab **HV System**

HV System

1. **NHQ from Iseg Spezialelektronik GmbH:** High voltage power sources of the type NHQ from Iseg Spezialelektronik GmbH are supported.

2. **High Precision NHQ:** To be activated if using a high precision NHQ power source.

3. **Time Out Factor:** Defines the hardware fail timeout.

4. **Com-Port:** The serial port has to be activated for data transfer to NHQ systems.

5. **Monitoring and Controlling:** Enables logging of the high voltage return values. If the actual value deviates from the set value by more than in **Range** is defined, CDT Detector control tries to adjust to the set value an amount of times defined in the second box. If adjusting fails, the measurement will be stopped and the high voltage power source will be ramped down.

3.4.3 Re-Initialize Detector Hardware

Reinitializes all connected and initialized hardware components again using standard values. Reinitializing may be helpful if some components have been powered off or deactivated. A manual configuration is not required then.

3.5 Window

Standard menu as used to be under Windows with options to rearrange the appearance of the opened documents.

3.6 ? (About)

? (About) shows the software version and its compile date and operating system. Furthermore the version of the *CASCADE Hardware Library*[2] is given.

[2]For more details, see CASCADE Hardware Library 3.8, Reference Manual, May 27, 2011.

Chapter 4

The Slow Control

4.1 Gas Control

Control → **Gas System...** opens the pop-up window **Gas Control** (see fig. 4.1), which allows the adjustment and monitoring of up to three gas flow controllers to set up a gas mixture.

> **Caution:** For safty reasons, after restarting the computer, the gas flow has to be checked explicitly before starting a new measurement.

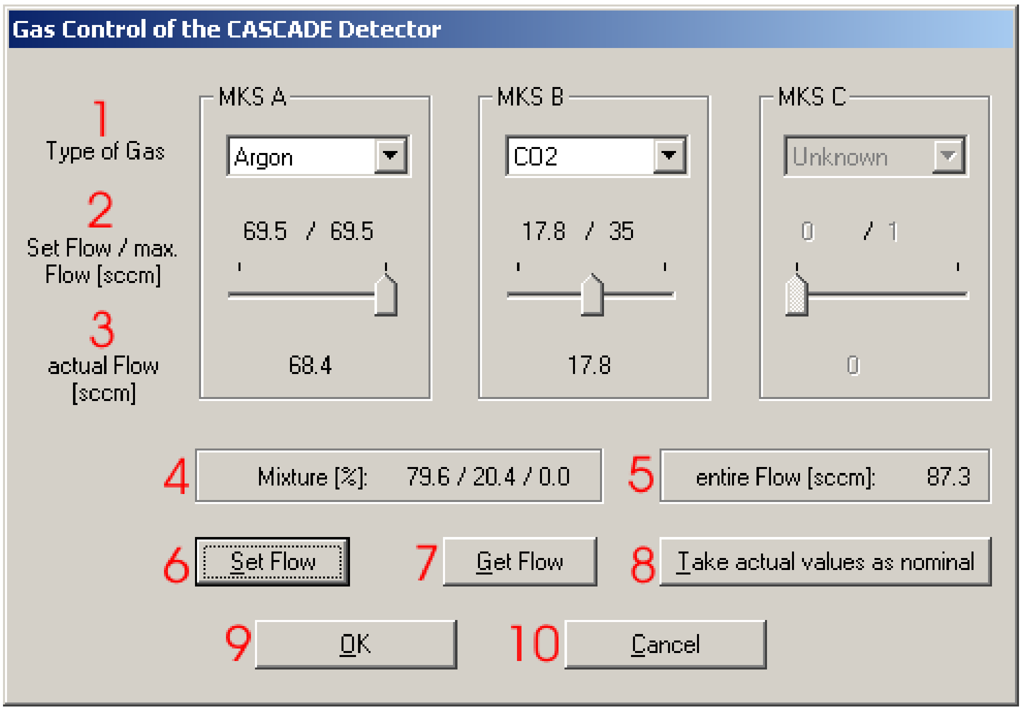


Figure 4.1: Pop-up window **Gas Control**.

1. **Type of Gas:** Set the type of gas used. This will adapt the correction factor for mass flow controller

(this type of controller is calibrated for the use of N_2 and correction factors have to be applied regarding the specific heat of other counting gases)

2. **Set Flow / max. Flow:** Ruler to adjust the amount of gas flow in sccm[1]. The available maximum value depends on the gas flow controller used. After a restart of the software the ruler may be set to zero, which does not imply an actual flow of zero sccm, but only that there has no value been set by the user.
3. **Actual Flow:** Display for the actual gas flow in sccm. The number will turn red if the actual value differs more than the value specified in **Range** (see 3.4.2)
4. **Mixture:** Display of the gas mixture ratio. For CASCADE 2D detectors a ratio of Ar/CO_2 between 90/10 and 70/30 should be chosen.
5. **entire Flow:** Display of the entire flow. For CASCADE 2D detectors a value between 50 sccm and 100 sccm should be chosen.
6. **Set Flow:** Button to set the chosen gas values to the gas flow controllers.
7. **Get Flow:** Button to read out the actual values of the gas flow controllers.
8. **Take actual values as nominal:** Button to take over the measured actual flow values as set flow values for the future (just like a shortcut to be relieved form setting all by hand).
9. **OK:** The chosen values will be set to the gas flow controllers and the dialogue will be closed.
10. **Cancel:** Close the dialogue without setting values to the gas flow controllers.

4.2 HV Control

Control → **HV System...** opens the dialogue **HV Control** (see fig. 4.2), which can be used to control up to two channels of a high voltage power supply.

Caution: After restarting the software it has to be checked via **HV Control** if the intended value for the high voltage is actually set. Before setting the high voltage, it has to be checked if the gas flow is set at least to the minimal operation value. This safety issue should prevent the detector from being damaged.

1. **Voltage Ramp:** **Set** the ramp [2 - 255 V/s] the voltage will be ramped up or down with. **Get** shows the actual value, which will be drawn in red as long as their is a difference between **Get** and **Set**. We recommend values below 10 V/s.
2. **Voltage:** **Set** the voltage [0 - Maximum]. The maximum is dependent from the HV device. **Get** shows the actual value, which will be drawn in red as long as their is a difference larger than **Range** (see 3.12) between **Get** and **Set**. When taking a detector into operation, we recommend within a first step about 100 V, to check the expected and measured current under **3**. Then steps of about 300 V together with measurements can be used to check, if everything is working fine (current, no sparks, etc.).
3. **Current:** Shows the actual current [μA].
4. **Status Word:** Shows the actual activity (voltage ok, ramping up or down, voltage off, error).
5. **Status Code:** Further detailed status information from the HV device.

[1] standard cubic centimetre per minute

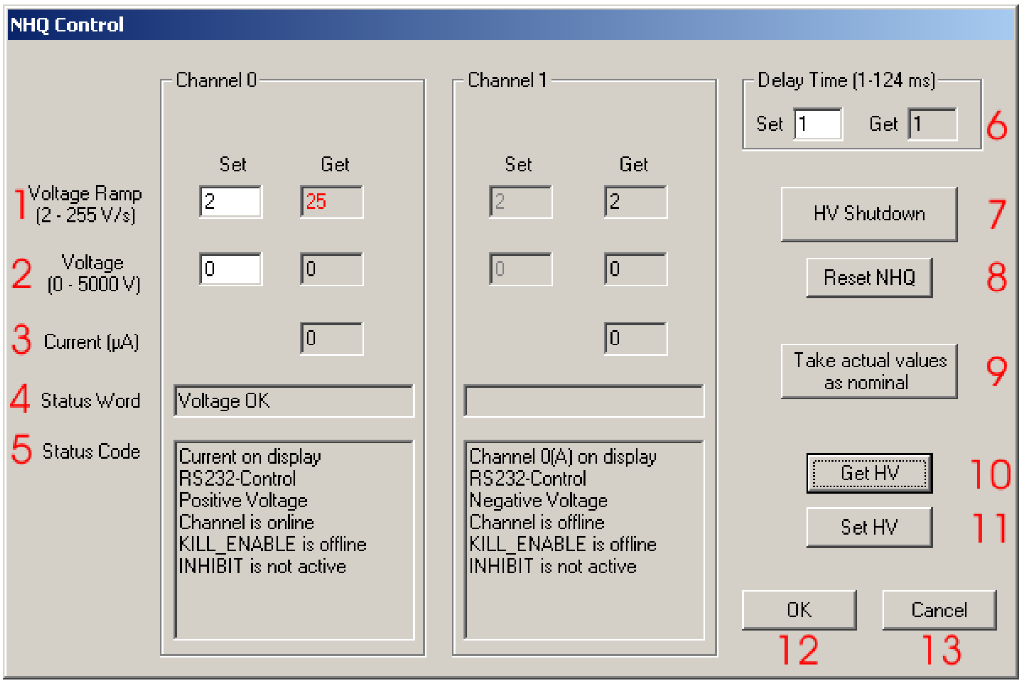


Figure 4.2: Dialog **HV Control**.

6. **Delay Time: Set** the Delay Time [1 - 124 ms] the HV device will wait until changing the voltage. **Get** shows the actual value, which will be drawn in red as long as their is a difference between **Get** and **Set**. We have no recommend on the value.
7. **HV Shutdown:** If there is a problem within the detector, gas supply or HV supply, one can press this button like a shortcut to initiate a fast but safe ramping down of the HV.
8. **Reset:** Reset the communication with the HV device.
9. **Take actual values as nominal:** Button to take over the measured actual values as set values for the future (just like a shortcut to be relieved form setting all by hand).
10. **Get HV:** Press to start a read-out loop for all actual values from the HV device. As long as the loop is active, the display will be refreshed every 1.5 s and **Get HV** will be replaced with **Reading HV**. Pressing again the button will end the loop.
11. **Set HV:** Button to set the chosen values to the HV device. Automatically the button **Get HV** will be pressed also.
12. **OK:** The chosen values will be set to the HV device and the dialogue will be closed.
13. **Cancel:** Close the dialogue without setting values to the HV device.

Chapter 5

The CASCADE DAQ Systems

Control allows configuring the initialized detector hardware components. For controlling the detector itself there are two CASCADE data acquisition (DAQ) systems (**CDR Systems** and **DAQBox Systems**). Both systems are based on a FPGA driven data acquisition, which is due to its modularity only adapted in specific parts.

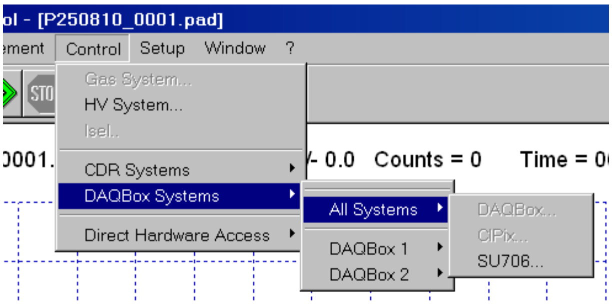


Figure 5.1: submenu **DAQBox Systems**

Detailled information about the firmware is given in the following manuals:

- CDR, Firmware Reference Manual, CDT GmbH,
- CASCADE DAQBox, Firmware Reference Manual, CDT GmbH.

Both, **CDR Systems** and **DAQBox Systems**, open similar sub menu items, which list the initialized DAQ systems (see fig. 5.1). These items only appear if the corresponding hardware had been initialized using **Setup** (see section 3.4.2). If several systems are active the sub menu **All Systems...** allows configuring all of them at the same time.
For a DAQ system these functional parts are available:

- **CDR.../DAQBox...** opens the corresponding **Control** dialogue to configure the firmware of the FPGA
- **CIPix...** opens the dialogue **CIPix Control**, which allows configuring the CIPix ASICs (charge sensitive preamplifiers).
- **IF15.../SU706...** allows configuring the additional Multi Channel Analyzer to measure pulse height spectra.

5.1 CDR / DAQBox Control

CDR.../DAQBox... opens the dialogue **Control**, which contains, depending on the systems defined in the **Setup** dialogue (see section 3.4.1, option **display advanced menus and dialogs**), up to four tabs for configuring the CASCADE firmware.
All settings are initialized to default values, which set the detector into a ready to use state.

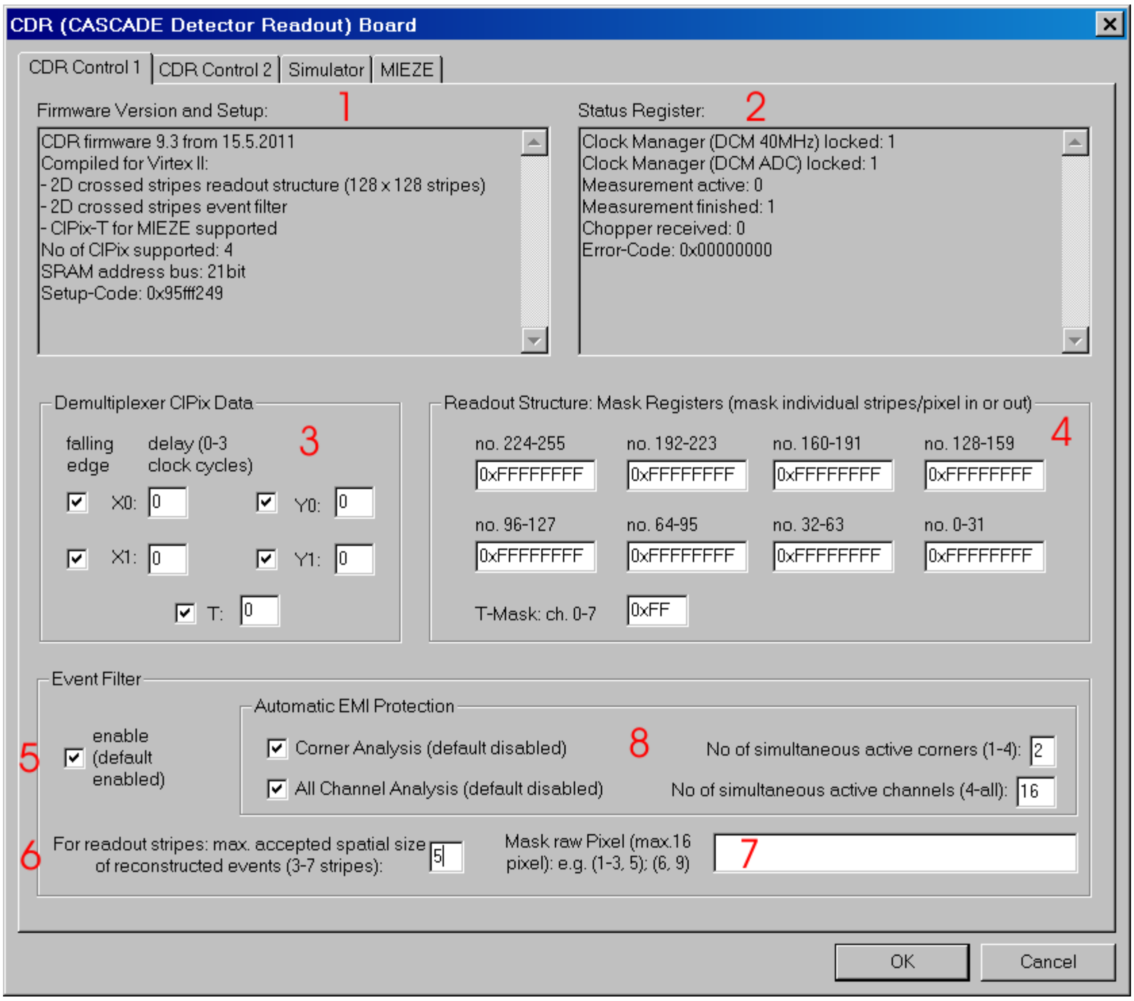


Figure 5.2: Dialogue **CDR / DAQBox Control 1**

5.1.1 CDR / DAQBox Control 1

1. **Firmware Version and Setup:** Displays the firmware version and the specific features of the firmware.

2. **Status Register:** Displays the status register.

3. **Demultiplexer CIPix Data:** The demultiplexer synchronizes the CIPix signals to the 40 MHz clock of the FPGA using the rising or the falling edge of clock signal. CIPix ASICs can be delayed up to three clock cycles to adjust synchronization to different signal path lengths.

4. **Readout Structure: Mask Registers:** Every single stripe or pixel can be masked by deleting its address bit in the 32 bit range. This may be necessary to exclude noisy or defect channels, which would increase the dead time of the **Event Filter**.

5. **Event Filter:** Processes the incoming signals of the data line to events. By turning off the Event Filter every single signal above threshold in every channel is counted.

6. **max. spatial size:** This option is only relevant in case of readout structures based on stripes. The maximum spatial size of accepted and reconstructed events can be set between 3 and 7 stripes.

7. **Mask raw Pixel:** This option is only relevant in case of readout structures based on stripes. In case of low or medium count rate distortions (so that no relevant dead time inside the event filter arises), which are concentrated only on a few pixel, (e. g. in case of so called burning pixel), these individual pixel can be masked out from further data processing and histogramming.

8. **Automatic EMI Protection:** Before all of the coincidence analysis within the filter is done the occurrence of sudden common mode signal distortions due to electromagnetic interference EMI can be detected. For EMI identification, the simultaneous firing of channels at the specified number of corners of the readout structure is detected and an analysis performed on how many channels over all are firing simultaneously. The data stream is then nulled for the relevant clock cycle.

5.1.2 CDR / DAQBox Control 2

1. **Type of List-Mode Data:** Inside the FPGA's firmware a data fifo of 4096 elements depth is available to store CIPix data vectors (64 bit for one CIPix chip) as they are enter in time slices of 100 ns. This fifo serves to store alternatively either raw CIPix data directly after demultiplexing or processed data after event reconstruction (filtered data). The former is used for testing, the latter may be employed for operational data acquisition if histogramming is not desired.
   For zero suppression of the raw data the number of zero lines after a finished signal can be adjusted.

2. **Master / Slave Mode:** All detectors configured as Slave are measuring synchronous to a detector set as Master.

3. **Digital Input/Output Multiplexer:** Each digital input signal can be mapped to one of the internal control signals via a multiplexer, which can be configured here. The same is valid for the mapping of the internal signals onto any one of the available outputs. The settings of all registers are not influenced by a global reset! Attention: the incoming digital signals have to be active for at least 50 ns! The length of the output signals can be set in units of 25 s in the field **Length of Output**. The following inputs are available:

   - Chopper
   - External Time Bin
   - Measurement
   - Counter

   The following outputs are available:

   - Chopper
   - New Time Bin
   - Measurement
   - New Event
   - MCA Gate
   - Time Clock

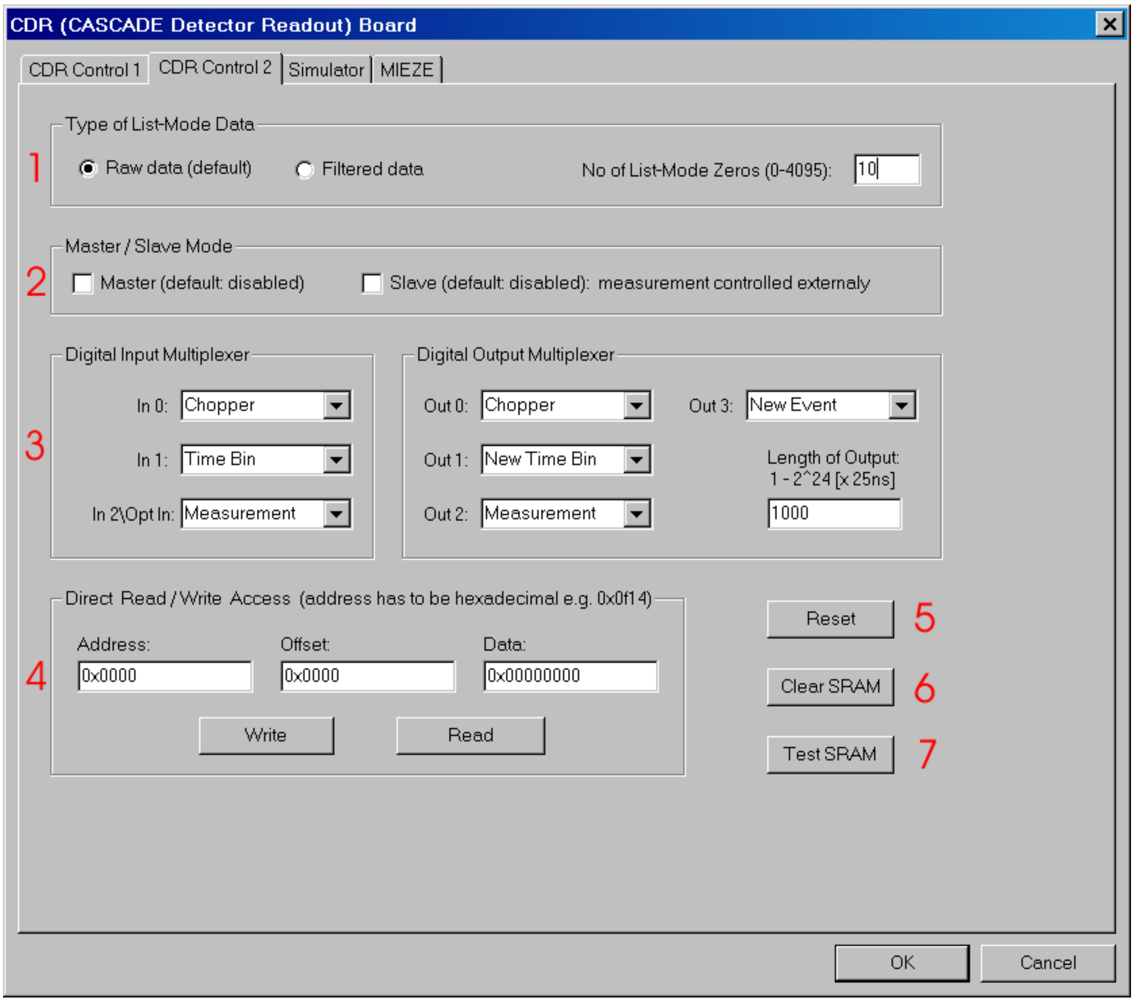


Figure 5.3: Dialogue **DAQBox Control 2**.

4. **Direct Read / Write Access:** Direct read and write access to any address of the firmware.
5. **Reset:** Global reset of the entire FPGA.
6. **Clear SRAM:** Clear the entire SRAM and all stored histogram data within the DAQ system.
7. **Test SRAM:** Initiate the FPGA to write a test pattern into the SRAM and read it back via software to be able to check SRAM functionality and consistency.

5.1.3 Simulator

Development work on the firmware needs to be proven repeatedly. To facilitate the analysis of the firmware's correct logical functionality after compilation as well as the correct timing behavior, the Simulator block was added, which generates artificial data and mimics CIPix and detector operation with a freely programmable time pattern. This simulator may be multiplexed into the data stream within the final firmware under test. The non statistical, pre-determined data serves to test the data pipeline, event reconstruction, data histogramming and communication with external hardware and software as it is to be processed in a deterministic way. Even very special, rare data structures or even the outcome of non physical data input may then be studied.

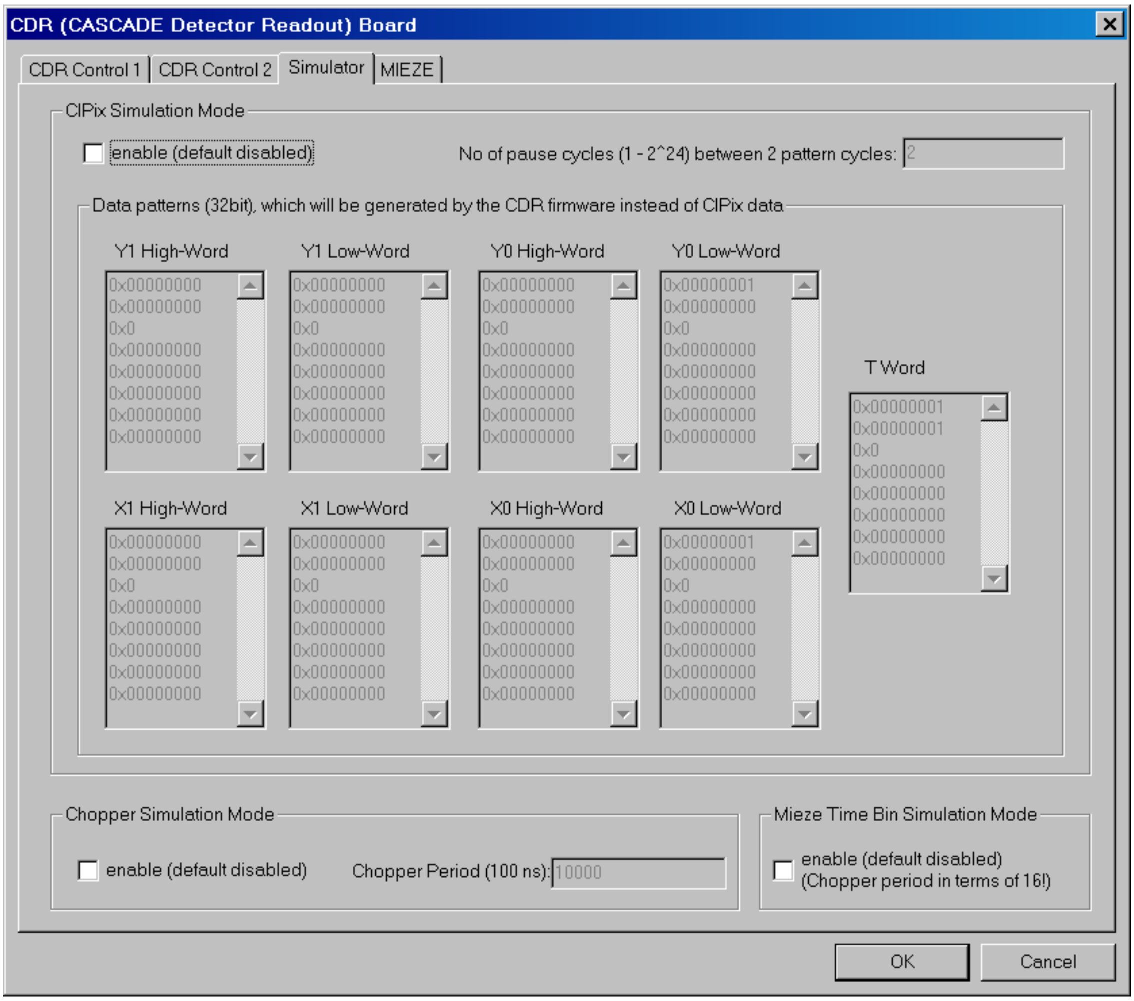


Figure 5.4: Dialogue **Simulator**

5.1.4 MIEZE

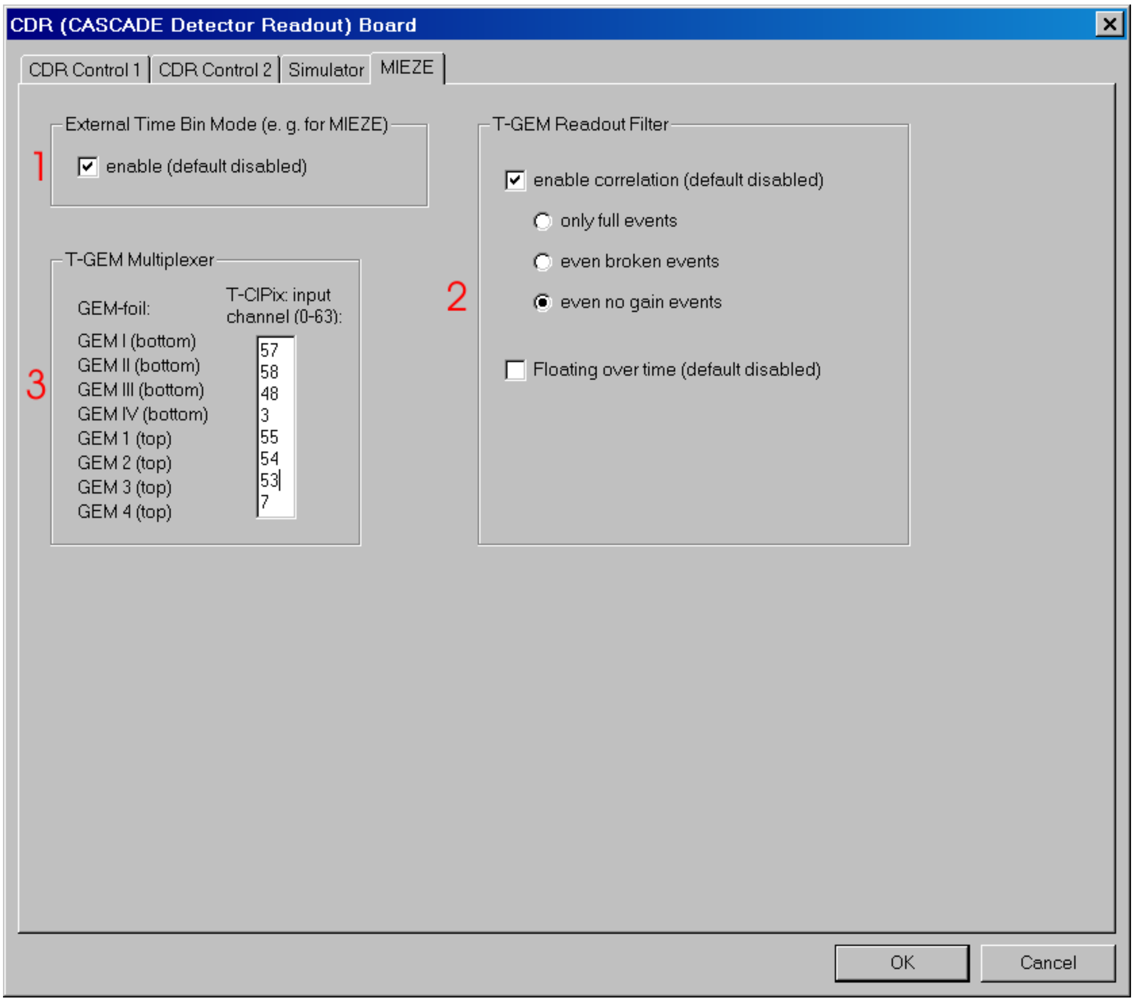


Figure 5.5: Dialogue **MIEZE**

1. **External Time Bin Mode:** The internal time bin counter, which is used in TOF mode to generate the time stamp **t**, is usually derived from the DAQ's internal clock oscillator. This can be changed to an external driven time bin counting, for example to guarantee, that the entire time stamp generation is done phase synchronous to an external oscillation frequency, as needed in MIEZE measurements.

2. **T-GEM Readout Filter:** For the CASCADE-MIEZE detectors it is possible to determine the particular boron layer in which the neutron was absorbed. For this purpose the detector is equipped with additional readout channels (CIPix T). Enable here to use its data to identify, in which boron layer **n** coated onto a GEM the absorption of the neutron took place. Option **even no gain events** is recommended to be used.

3. **T-GEM Multiplexer:** Which readout channel of the CIPix T is used to read out which boron layer on a GEM. The sorting scheme starts with the bottom half-room of the detector starting with GEM I, which is directly located beneath the readout structure followed by GEM II and so on (see Fig. 5.6). The second half of TOF spectra represents the top half-room of the detector starting with GEM 1, which is directly located above the readout structure followed by GEM 2 and so on up to the neutron entrance window of the detector (also see Fig. 2.4).

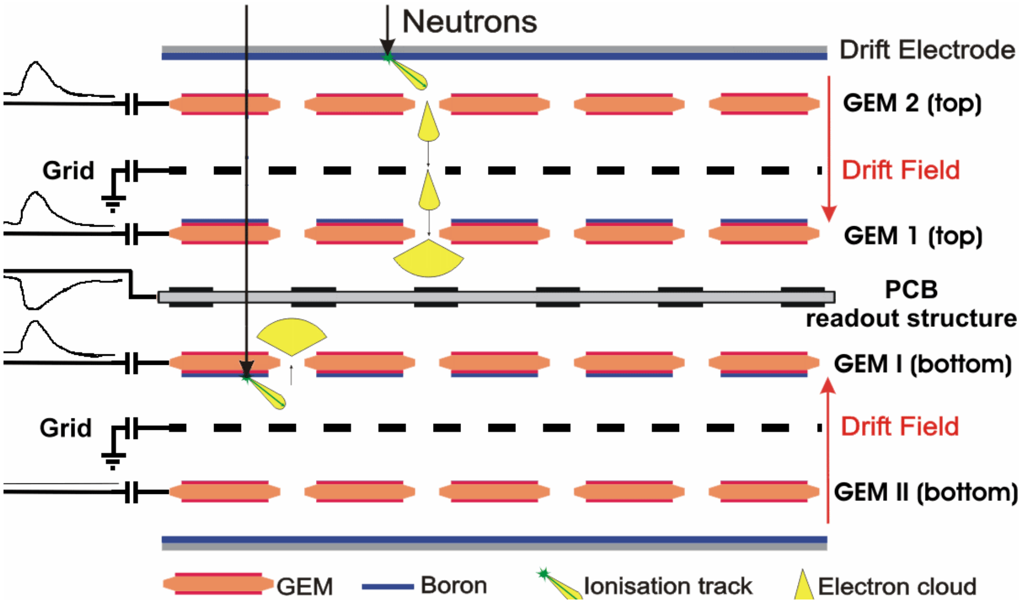


Figure 5.6: Schematic of the CASCADE-MIEZE detector concept and naming scheme of the GEM foils, which are read out through CIPix T.

5.2 CIPix Control

This dialog **CIPix Control** provides access to the registers of the CIPix preamplifier ASIC. The CIPix chip integrates 64 channels of a low noise charge sensitive preamplifier followed with a shaper and discriminator. The amplified and shaped input signals are digitized by an AC-coupled comparator with configurable polarity sampled at a clock rate of 10 MHz. The bias settings and various other parameters are programmable and can be monitored via an I^2C-bus serial protocol. For diagnostic purposes, the analogue output of any one shaper may be switched to an output monitoring channel. Thus, for one selectable channel, the full analogue behavior may be observed. Furthermore, the chip is equipped with a programable internal test-pulse generator that allows to generate test-pulses in any given channel. A detailed description of the CIPix registers and their functionality can be found in the following manual: *AS20 CIPix-Boards Reference Manual, July 31, 2011*.

The **CIPix Control** dialogue contains tabs for configuring every single ASIC (see sig. 5.7). A **Get** value will be displayed red if it is different to the **Set** value. During the initialization phase of the systems, the CIPix ASICs will be set to default values, which are read and displayed. The following DACs are available:

1. **Vref0 and Vref1:** Control of the comparator threshold level of the first 32 readout channels (channels 0-31) and for channels 32-64.

2. **Analog Out:** The frontend chip may be programmed to relay the analogue signal of any one analogue channel out of the 64 channels to the external world.

3. **TpReg:** Activation of internal test-pulse generator, which acts in two different modes:

   - Single channel mode: test-pulses are injected into the analogue input channels `AnalogIn<0>`, `AnalogIn<21>`, `AnalogIn<42>` and `AnalogIn<63>`.
   - Multi channel mode: test-pulses are injected into all analogue input channels.

4. **CIPixReg:** This register controls the polarity of the comparator. Unfortunately **positive** enables so far the discrimination of negative input charges as gas detectors usually provide on their readout anodes.
5. **Hard Reset:** Resets all internal registers of CIPix to zero and restarts the internal state machine.
6. **Soft Reset:** Restarts the internal state machine only. All internal registers of CIPix will be left unchanged
7. **MIEZE X/Y:** Default values for the configuration of the X-Y readout CIPix chips within a CASCADE-MIEZE detector.
8. **MIEZE T:** Default values for the configuration of the CIPix T chip within a CASCADE-MIEZE detector used to read out the direct GEM signals.
9. **Default:** Default values for the configuration of CIPix chips within a standard CASCADE detector.
10. **Take actual values as nominal:** Button to take over the measured actual values as set values for the future (just like a shortcut to be relieved form setting all by hand).

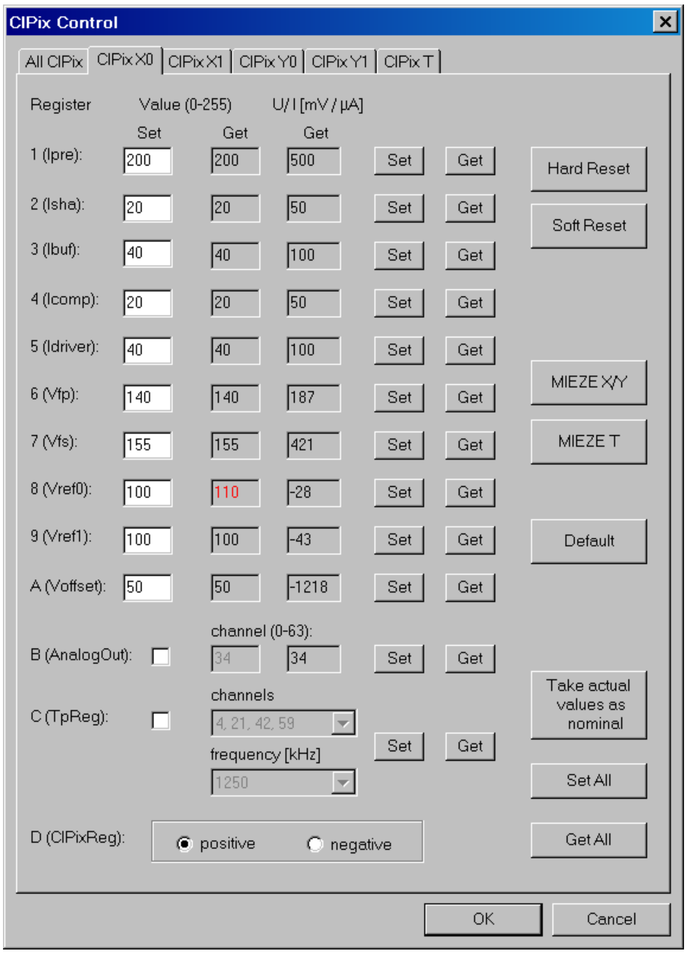


Figure 5.7: Dialogue **CIPix Control**, tab **CIPix X0**

In case of more than one CIPix ASIC, the tab **All CIPix** is shown (see fig. 5.8), which can be used to set a common configuration to all ASICs . Register values, which are different among the individual CIPix ASICs, will be displayed as zero and marked in blue.

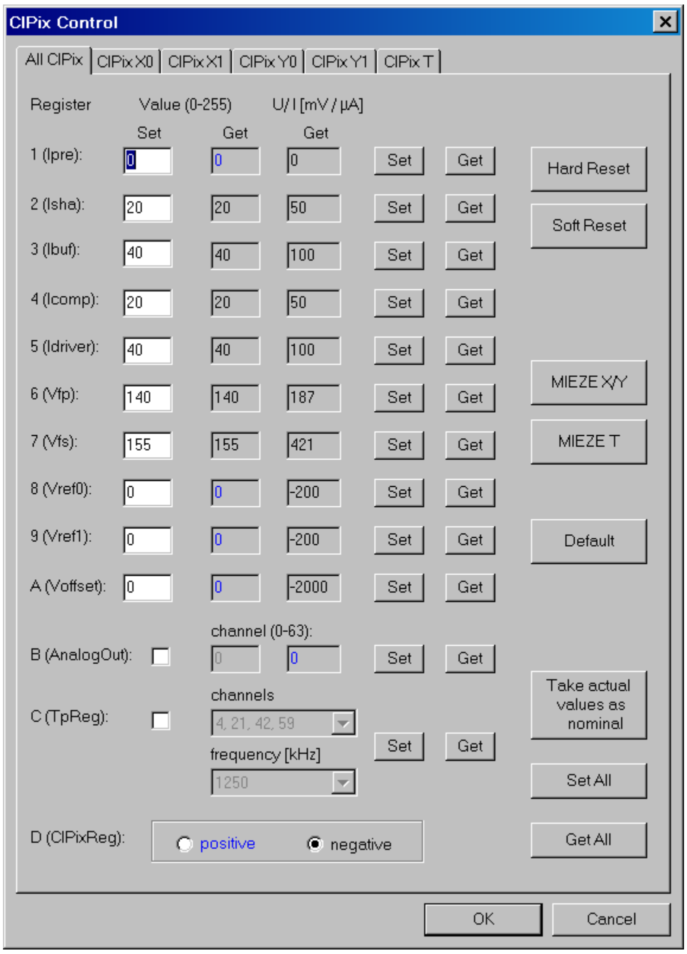


Figure 5.8: Dialogue **CIPix Control**, tab **All CIPix**

5.3 IF15 / SU706 Control

IF15 / SU706 Control allows configuring the additional Multi Channel Analyzers (MCA) IF15 and SU706, which can be used to measure pulse height spectra.

The IF15 module consists of five independent Analog-to-Digital converts (ADC), which sample the CIPix signals continuously at a frequency of 40 MHz and a resolution of 12 bit. This data stream will be converted to a pulse height spectrum by the local Spartan-3 FPGA and saved to the SRAM of the detector.

The SU706 module consists of a single ADC, which samples the CIPix signals continuously at a frequency of 40 MHz and a resolution of 14 bit. This data stream will be converted to a pulse height spectrum by the firmware of the DAQBox and saved to the SRAM.

1. **Firmware Version and Setup:** Displays the firmware version and the specific features of the firmware.
2. **Status Register:** Displays the status register.
3. **ADC Mode:** The MCA firmware analyzes the data stream and can be configured as follows:

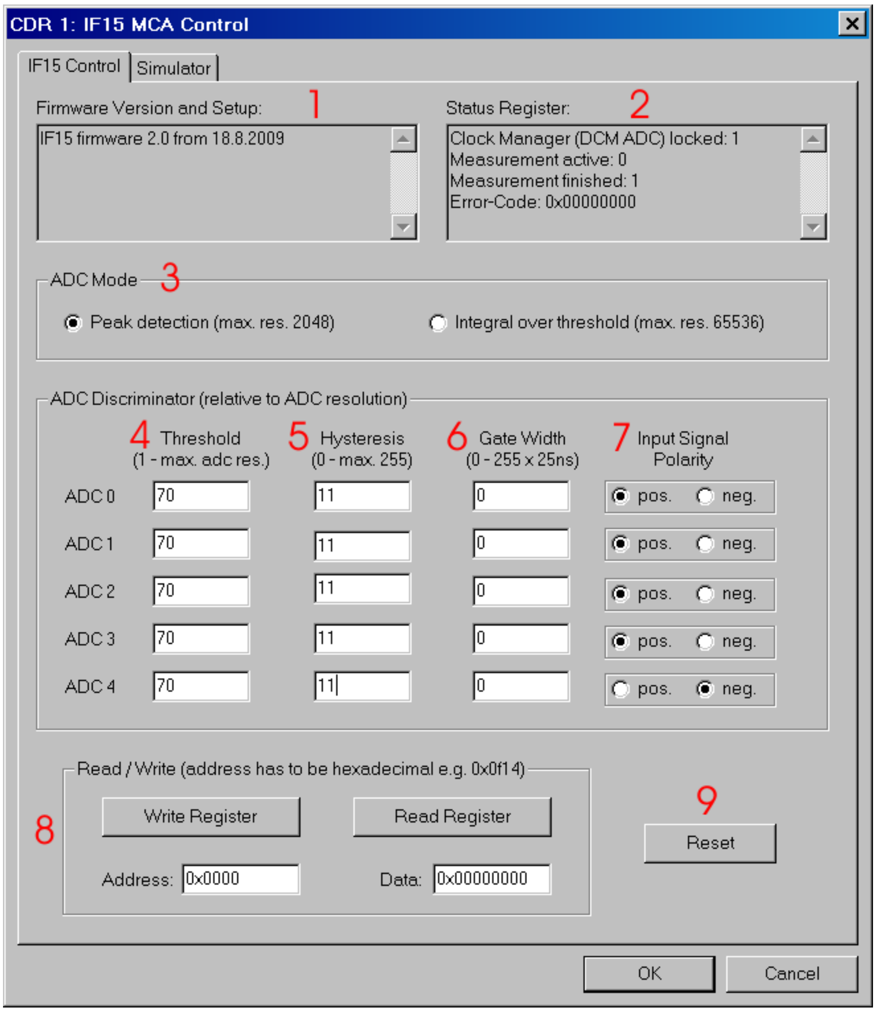


Figure 5.9: Dialogue **IF15 MCA SubModule**

- **Peak Detection** gives the maximal signal amplitude above the configured **Threshold** independent of the duration of the signal.
- **Integral over Threshold** gives the sum of all ADC values while a signal stays over the configured **Threshold**

In both cases a too low **Threshold** (below the noise level) will lead to a fake count rate.

4. **Threshold:** will set the threshold for incoming signals to the ADC. This value refers to the configured ADC resolution (see 6.3) as also can be seen in the data display (see fig. 2.6). Changing the resolution will as well lead to a change of the absolute value of the threshold as well.

5. **Hysteresis:** The digital 'hysteresis' defines the value, a signal has to be below threshold in addition to the discriminator threshold to be set as finished internally.

6. **Gate Width:** Instead of defining a **Threshold - Hysteresis** a fixed time can be configured (gate width), after which a signal will be set as finished internally. A value different to zero will override the **Hysteresis** option.

7. **Input Signal Polarity:** Sets the signal polarity for the ADC. Either positive or negative peaks can be analyzed. This reduces the resolution to 11 bit (IF15) and 13 bit (SU706) respectively.

8. **Read / Write Access:** Direct read and write access to any address of the MCA firmware.
9. **Reset:** Global reset of the entire MCA-FPGA firmware.

Chapter 6

The Toolbars

For every toolbar additional information is provided, which is given by a tooltip by hovering over with the cursor (see as well 6). The information bar at the bottom of the window shows as well a more detailed help description.

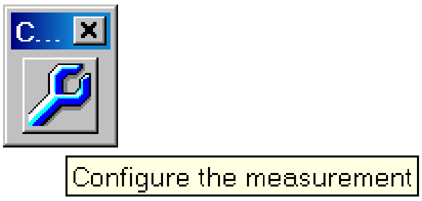


Figure 6.1: **Config** toolbar

6.1 Standard Bar, Measurement Bar and Config Bar

The default toolbars **Standard**, **Measurement** and **Config** will be shown for every type of measurement (see fig. 6.4). The **Standard** toolbar provides functionalities of the **File** menu (see fig. 3.1) and the **Measurement** toolbar provides functionalities of the **Measurement** menu (see fig. 3.3). In case of carrying out TOF- or Sweep-Spectrum measurements and HV-, or CIPix scans additional toolbars will be shown.

Figure 6.2: **Standard** toolbar

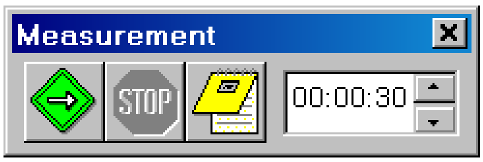


Figure 6.3: **Measurement** toolbar

Figure 6.4: **Standard, Measurement and Config** toolbars

1. **New:** Opens the dialogue **New** (see fig. 2.1) to start a new measurement selecting a certain type.

2. **Open:** Opens a saved measurement file.
3. **Print:** Prints the measurement of the active window.
4. **Start:** Starts a measurement (s. 3.2).
5. **Stop:** Stops the measurement.
6. **Comments:** Opens a dialogue window to give comments to every or the active measurement (s. 3.2.3).
7. **Measurement Time:** The measurement time can be set as HH:MM:SS. Caution: For a scan cycle consisting of several individual measurements, this is measurement time for every single run!
8. **Configure the Measurement:** Opens the dialogue window **Configuration of the Measurement** (s. 6.1.1).

6.1.1 Config Toolbar - Configuration of the Measurement

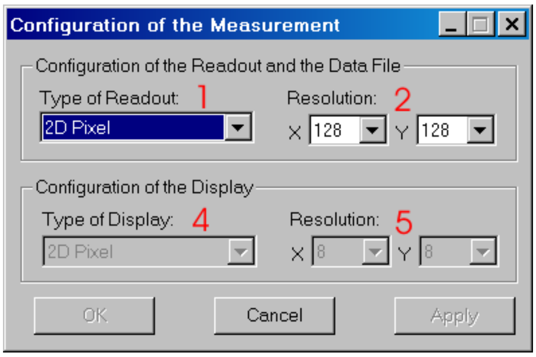


Figure 6.5: Dialogue **Configuration of the Measurement**

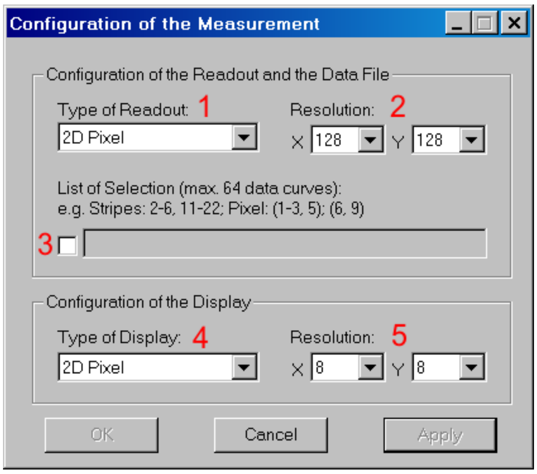


Figure 6.6: Dialogue **Configuration of the Measurement** with **Advanced menues** (see under **Software Setup...** 3.4.1)

The button **Configure the Measurement** of the **Config** toolbar (see fig. 6.1) opens the pop-up window **Configuration of the Measurement** (see fig. 6.6), which is used to configure the readout type.

1. **Type of Readout:** set the spatial type of readout structure.
2. **Resolution:** define spatial resolution of the data file.
3. **List of Selection:** configures the type for individual pixels or stripes
4. **Type of Display:** set the spatial arrangement of the channels in the data display.
5. **Resolution of Display:** define spatial resolution of the data display. This value is independent from the the resolution used in the data file. This may be helpful to reduce the full amount of data of 128 x 128 pixels. The default setting reduces the data display to a maximum of 64 plots.

Figure 6.7: **TOF Control** toolbar

6.2 TOF Control

1. **Stop:** Additional button for TOF/Sweep-Spectrum. This button will be active after pressing the stop button of the **Measurement** toolbar (see fig. 6.3) to terminate the measurement immediately (see 2.3).

2. **Dwell Time:** Set the dwell time in units of 100 ns.

3. **Number of Time Bins:** Define the number of TOF channels to use.
   Caution: in combination with the spatial X/Y resolution this defines the size of a data set and therefore the amount of allocated memory of the DAQ electronics as well as on the disk.

4. **Break Mode, Sweep Mode:** In **Break Mode** every chopper pulse resets the measurement to TOF channel '0'. In **Sweep Mode** incoming chopper pulses will be ignored until the given number of time bins is reached.

5. **No of Sweeps:** Instead of defining a measurement time, a fixed number of sweaps can be defined. A number different from zero overrides the measurement time setting.

6.3 PHA Control

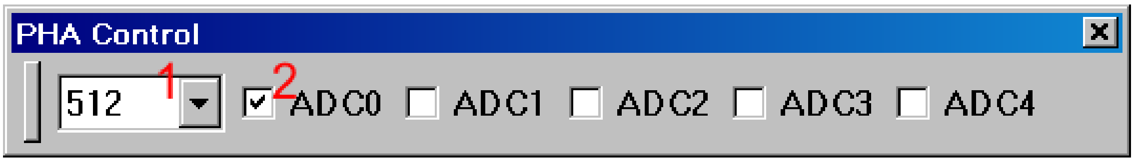


Figure 6.8: **PHA Control** toolbar

1. **Resolution of ADC:** Define the number of ADC channels to sample the ADC range (from 0 V up to the maximum voltage). This only affects the binning, not the range, and may be used to optimize statistics at low count rates.

2. **Activate ADC No. #:** Choice of the ADC to be read out.

6.4 CIPix Control

1. **CIPix:** Selection of the CIPix to be scanned (Single, X0, Y1, T, All)?

2. **Start value:** 0 to 255

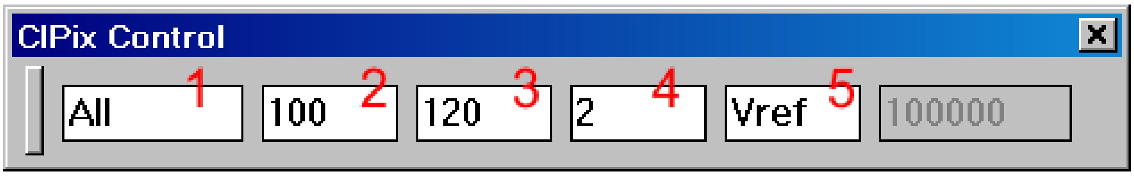


Figure 6.9: **CIPix Control** toolbar

3. **End value:** 0 to 255
4. **Step width:** -254 to 254
5. **CIPix register:** CIPix register to be scanned. The discriminator threshold registers Vref1 and Vref2 can be scanned synchronously by using the key word Vref.

6.5 HV Scan Control

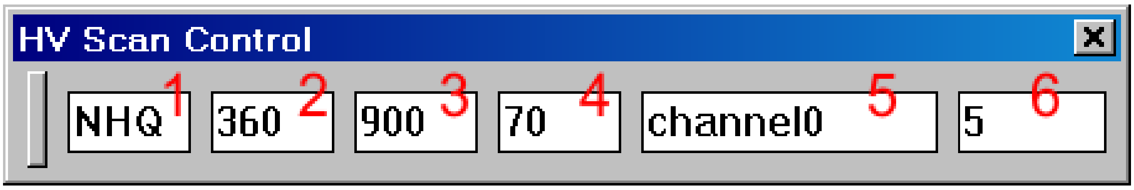


Figure 6.10: **HV Scan Control** toolbar

1. **HV device:** define the type of high voltage power supply to use (NHQ or EHQ).
2. **Start value:** Start value of the high voltage scan. If the start value is higher than the end value, the scan will be performed the other way round. Possible values are defined by the type of power supply which is used.
3. **End value:** End value of the high voltage scan. Possible values are defined by the type of power supply which is used.
4. **Step width:** Increment value of the high voltage scan. Values can be chosen from -500 V to 500 V
5. **Voltage to be scanned:** Choice of the channel to be scanned (channel0 or channel1).
6. **Pause:** define the time [s] to wait between two voltages as the power supply requires a certain amount of time to stabilize.

Appendix A

Detailed Description of the CASCADE Event-Filter Algorithm

Die Signal-Ladungswolke im Detektor nach einer Neutronenkonversion kann an den CIPix Vorverstärker-ASICs ein Signal erzeugen, welches gleichzeitig über mehrere Eingangskanäle (Ortskanäle) verteilt ist und das während mehrerer Taktzyklen (je 100 ns) über den Diskriminatorschwellen der betroffenen Vorverstärkerkanäle liegt. Jeder CIPix übergibt alle 100 ns eine Eins an den FPGA, wenn das vorverstärkte analoge Ladungssignal am jeweiligen Diskriminator größer als der per Software eingestellte Referenzwert (V_{ref}) war. Damit entsteht eine Datenmatrix, in der entweder eine 0 oder 1 steht und deren Indizes durch die diskreten Zeitschritte und die zweidimensionale Position des Pixels gegeben ist. Diese muss im FPGA verarbeitet werden.

Tabelle A.1 veranschaulicht den Datenfluss im FPGA als zweidimensionale Matrix mit der Kanalnummer als Abszisse und der Zeit in Einheiten von 100 ns als Ordinate.

Kanalnummer $\longleftarrow$

$$
\begin{array}{ccccccc}
\ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \iddots \\
\cdots & x_{i+2}^{j-1} & x_{i+1}^{j-1} & x_{i}^{j-1} & x_{i-1}^{j-1} & x_{i-2}^{j-1} & \cdots \\
\cdots & x_{i+2}^{j} & x_{i+1}^{j} & x_{i}^{j} & x_{i-1}^{j} & x_{i-2}^{j} & \cdots \\
\cdots & x_{i+2}^{j+1} & x_{i+1}^{j+1} & x_{i}^{j+1} & x_{i-1}^{j+1} & x_{i-2}^{j+1} & \cdots \\
\iddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{array}
\qquad \text{Zeit} \downarrow \qquad
\begin{array}{ccccc}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{array}
$$

Table A.1: Links: Repräsentation der CIPix-Daten. Der hochgestellte Index bezeichnet den Taktzyklus, der tiefgestellte Index den Ortskanal.
Rechts: Beispiel für ein Event, das sich nacheinander über 4 Zeit- und 3 Ortskanäle verteilt.

Eine Ladungswolke, die von einem Neutron kommt, erzeugt so einen Block zusammenhängender Einsen. Deshalb darf nicht grundsätzlich jede Eins als ein einzelnes Neutron aufgefasst werden. Leider kann man gar nicht unterscheiden, ob es doch zufällig mehrere Neutronen sind oder nur ein einziges. Am physikalisch sinnvollsten ist es daher, mit einigen plausiblen Annahmen die untere Grenze der Anzahl an detektierten Neutronen zu einer bestimmten Matrix zu finden. Dazu braucht man einen Filter, der den zweidimensionalen Datenstrom verarbeitet. Dabei kann dieser Filter nicht für alle auftretenden Konfigurationen die untere Grenze erreichen, da die untere Grenze stark von den Annahmen abhängt, die man macht. Unter Umständen würde der Filter auch viel zu viele Ereignisse verwerfen. Zum Beispiel sind zwei isolierte Einsen mit mehreren Zeittakten Abstand mit ziemlicher Sicherheit zwei echte Neutronenereignisse. Tritt jedoch ein Muster auf, bei dem fast alle Pixel zusammenhängend über einen gewissen Zeitraum ansprechen, so wird ein Filter, der rigoros jedes zusammenhängende Muster verarbeitet und reduziert, diesen Cluster zu einem einzigen Event zusammenfassen. Dieses Muster wurde aber ziemlich sicher von mehreren Neutronen erzeugt oder rührt sogar von einer elektromagnetischen äußeren Störung her.

Bei einer Rohdatenrate von 320 MByte pro Sekunde bei 4 gleichzeitig arbeitenden CIPix Chips kann diese Menge nur schwer an einen externen Rechner übermittelt und abgespeichert werden. Die nachträgliche Verarbeitung solcher Datenmengen ist ebenfalls mit erheblichem Aufwand verbunden. Deshalb muss die Datenverarbeitung schon auf dem FPGA stattfinden. Für alle Arten von Histogrammen, die auf dem FPGA erstellt werden, braucht man einen Filter. Ein sehr allgemeiner Filter muss beliebige Muster verarbeiten können. Dies kann zu sehr komplexen Filtern führen, die meist viele Taktzyklen und viele Ressourcen verbrauchen. Im Gegensatz dazu hat ein Computer einen festen Befehlssatz. Das Computer-Programm kann im Verlauf seiner Verarbeitung aber je nach Muster sehr unterschiedliche Wege im Programmablauf gehen. Dies kostet nur Taktzyklen und damit Zeit. In einem FPGA wird jedoch Hardware realisiert, die gleichzeitig arbeitet. Ein Filter muss deshalb für verschiedene zu verarbeitende Bedingungen, eine Vielzahl von Logikbausteinen verwenden, die alle gleichzeitig arbeiten, wobei am Ende aber nur eines der berechneten Ergebnisse auch tatsächlich benutzt wird. Hier kostet der Filter viele Ressourcen des FPGA.

Prinzipiell ist der Algorithmus des Filters in mehrere hintereinander ausführbare Schritte aufgeteilt (Pipeline). Zunächst wird der Datenstrom kontinuierlich in der Zeitdomäne analysiert und zu einem einzigen räumlich zusammenhängenden Cluster an Einsen zusammengefasst. Ein geschlossener Rand von Nullen in Raum und Zeit definiert dabei den Cluster. Gleichzeitig wird dabei auch schon die zeitliche Länge des Events ermittelt, indem für jeden Ortskanal die Anzahl der eingetroffenen Einsen unter Berücksichtigung der Nachbarkanäle hochgezählt wird (s. Tabelle A.2).

$$
\begin{array}{ccccc}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{array}
\Rightarrow
\begin{array}{ccccc}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 1 & 3 & 4 & 0 \\
0 & 0 & 0 & 0 & 0
\end{array}
$$

Table A.2: Links: Beispiel für ein Event, das sich nacheinander über 4 Zeit- und 3 Ortskanäle verteilt. Rechts: Ergebnis der ersten Stufe der Event-Filter-Pipeline. Das Event verteilt sich über die angezeigten 3 Ortskanäle und war bis zu 4 Taktzyklen lang.

Aber auch die Vielzahl der verbleibenden räumlichen Formen und Positionen, die nun noch möglich sind, übersteigt im Allgemeinen die Ressourcen des FPGA. Da die Reichweite der geladenen Reaktionsprodukte der Neutronenkonversion im Detektor jedoch endlich ist, können nicht beliebig große Flächen an Einsen von einem Neutron erzeugt werden. Deshalb lassen sich physikalisch wahrscheinliche Flächen definieren, nach denen gesucht wird. Dies heißt aber auch, dass für jeden Auslesestrukturtyp ein eigener Filter notwendig wird.

A.1 The 2D-Pixel Event-Filter Algorithm

Die Anzahl der potentiell in Frage kommenden Eventmuster wird dadurch festgelegt, dass die Pixel der Auslesestruktur $12.5\ mm \times 12.5\ mm$ groß sind und dass die Reichweite von ^7Li- und α-Teilchen nicht größer als 5-7 mm ist. Es ist daher unmöglich, dass die Ladungswolke eines Neutrons über mehr als zwei Pixel in einer Richtung verteilt ist. Für jedes Pixel der Auslesestruktur wird überprüft, ob eines dieser Muster vorliegt. Die für die 2D-Pixel Auslesestruktur akzeptierten Ereignismuster sind in den Schemata A.3 - A.5 dargestellt.

$$\begin{array}{ccc} 0 & 0 & 0 \\ 0 & |1| & 0 \\ 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{array}$$

Table A.3: Single-Pixel Event

$$\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & |1| & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \; und \; \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & |1| & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}$$

$$\begin{array}{ccc} 0 & 0 & 0 \\ 0 & |1| & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \; und \; \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & |1| & 0 \\ 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{array}$$

Table A.4: Double-Pixel Event

Das Pixel, dessen Umgebung untersucht wird, ist mit $|X|$ gekennzeichnet. Die Muster stellen von Nullen umrandete Pixelflächen dar. Dies sorgt dafür, dass die Muster sich gegenseitig ausschließen. Das heißt, es können nicht gleichzeitig zwei der Muster die Umgebung eines Pixels sein. Tritt ein anderes Muster auf, so wird es bei dieser räumlichen Analyse verworfen und als "Invalid" gezählt.

Die zum Schluß noch verbleibende Eins wird als die räumliche Position des Neutronenevents interpretiert. Dabei die werden die sogenannten Double-Events, also Events auf 2 benachbarten Pixeln, rein statistisch mal dem einen oder dem anderen Pixel zugeordnet (s. Schema A.4). Die entsprechende Pixelnummer (= Adresse) dient nun als Grundlage für das Inkrementieren des entsprechenden Histogrammzählers im lokalen SRAM der Detektorelektronik.

$$
\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & |1| & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array}
$$

$$
\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & |1| & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}
$$

$$
\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & |1| & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}
$$

$$
\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & |1| & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \Rightarrow \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}
$$

Table A.5: Triple-Pixel Event