The ATLAS SemiConductor Tracker Endcap

Peeters, S.J.M.

Citation for published version (APA):
Chapter 5

Single module read-out system

“There ain’t no rules around here! We’re trying to accomplish something!”

Thomas A. Edison

The single module read-out system at NIKHEF has to verify and fully characterise electrically the modules produced at NIKHEF, and to make a quick check on modules received from other institutes prior to mounting them on discs.

This section describes the set-up and the methods used to analyse the module data. The emphasis will be on the main differences between the set-up developed at NIKHEF and other set-ups used within the SCT community.

5.1 Overview

Figure 5.1 shows an overview of the single module read-out system used at NIKHEF.

The system can be divided in three main components. These are a testbox that holds and cools the module, a Data AcQuisition (DAQ) system and a Detector Control System (DCS).

The DAQ is the same as is used in the rest of the SCT community, except that NIKHEF uses an optical connection to the module, as will be done in ATLAS. Two VME \(^1\) modules, the CLOAC and SLOG, generate the clock and the commands that need to be sent to the module. There is one VME module, MuSTARD, which reads the module out. A PC that is connected to the VME bus using a standard National Instruments card controls these VME modules. The software package SCTDAQ, running on the PC, controls the read-out process. A so-called ‘BiPhase-Mark’ (BPM) encoder does the optical conversion of the clock and the commands onto a single fibre. The data from the modules is received via two optical fibres, one for each side of the module. A prototype

\(^{1}\) VME crates are commonly used in high energy physics. A VME crate can hold several electronics modules and provides a standardised communication bus.
optical receiver converts the optical signals to electrical signals. The optical connection to the module is made by a special optical plug-in built at NIKHEF.

The DCS at NIKHEF consists of conventional power supplies and a multichannel digital multimeter which are read-out. A Labview program running on a PC controls and reads these units out via a General Purpose Interface Bus (GPIB). This program sets and logs the module supply voltages and logs hybrid and module box temperatures via the multimeter. If the hybrid temperature gets too high, the power supplies are switched off.

The module box is custom made. It makes sure that the module runs in a dark, cooled and electronically clean environment.
5.2 Hardware

The single module readout system is designed to be an optimal electrical environment for the detector module. The power supplies used are low noise bench power supplies. All the cables are kept as short as possible. All electrical devices are connected at the same point to the main power and ground, to avoid ground loops.

This section describes the details of the different components. The standard components or the components supplied by the SCT community are only described briefly, while the components developed at NIKHEF are described in more detail.

5.2.1 DAQ

5.2.1.1 VME modules

Figure 5.2 shows an overview of the interaction of the individual VME modules used in the data-acquisition. The data-acquisition is controlled via a PC running the standard software SCTDAQ (see Section 5.3.1). The PC running this software is connected by a National Instruments PCI-VXI card to a VXI-VME card in the VME crate. The VME crate holds three electronics cards developed by the SCT community, which are used to send the clock and the commands to, and decode the data from, the detector module under test. Although at NIKHEF this set of modules is currently only used to read out one single module, it is capable of controlling and reading out up to six modules (which each produce two data streams). It can be expanded to handling up to twelve modules by adding one MuSTARD.

5.2.1.1.1 CLOAC The acronym CLOAC stands for ‘CLOck And Command’. The CLOAC module has been developed by the University College London (Postranecky 1998). It generates the system clock of 40.08 MHz. It also generates the following, so-called fast commands that are only three or seven bit long (Dabrowski 2002):

- Trigger;
Single module read-out system

- Calibration pulse;
- Soft reset (resets the bunch crossing and trigger counter, and clears a possible randomiser buffer-overflow flag);
- Bunch crossing reset (resets the bunch crossing counter).

The calibration-pulse command is followed by a trigger command after a programmable delay of 0 up to 143 clock periods. The calibration pulse has to be processed by the analogue circuit and the 132-bit long pipeline before it can be read-out. The delay enables the CLOAC to be programmed with the correct delay to obtain the hit information of the calibration pulse. This calibration-pulse plus trigger sequence can be issued either regularly or randomly, at a rate up to about 10 kHz. This burst can be interrupted by an externally applied busy signal.

Similar facilities for the other commands are available. The CLOAC can be synchronised to an external clock and can use an external trigger signal.

5.2.1.1.2 SLOG The main function of the SLOG (Morrissey 1999) is to add slow commands, which are longer than 7 bits, to the command channel. The acronym stands for ‘SLOw command Generator’. It was developed at the Rutherford Appleton Laboratory. It receives clock and fast commands from the CLOAC, and fans them out to up to 12 channels; and it can be programmed to inject slow commands synchronised to the clock. Fast commands can be masked off during slow commands to avoid corruption.

The slow commands are used to set the registers of the ABCD, controlling DACs and read-out mode etc. The required command sequences are loaded into a 62 kilobit long stream, independently for each of the 12 modules. When SCTDAQ sends a start command to the SLOG, it loads the bit-stream onto the command channel. Sequences can be repeated up to 16 times.

5.2.1.1.3 MuSTARD The acronym MuSTARD stands for ‘Multichannel Semiconductor Tracker ABCD ReaDOut’. The MuSTARD module has been developed by the Rutherford Appleton Laboratory and Cambridge University (Morrissey & Goodrick 1998). It can read out twelve data streams. Since a module produces two data streams, one for each side of the module, it can read out up to six modules. Two MuSTARDs can be operated in combination with one CLOAC/SLOG combination. The MuSTARD receives its clock directly from the CLOAC. Since the phase of the data is not defined with respect to the clock, each data stream can be delayed between 0 and 24 ns. This delay can be set in steps of 1 ns.

If a channel is enabled, any data received is written into a memory block associated with that channel. Once initialised, the MuSTARD loops over the individual channels’ memory and decodes the data. If no data is found in a channel’s memory, the MuSTARD waits for it. The MuSTARD provides no time-out mechanism. The DAQ software does this.

The MuSTARD also receives the commands from the CLOAC. This is used to count the number of triggers issued. A counter is kept for each channel. This counter is decremented after an event is read (i.e. an event trailer is decoded). For each stream, a maximum can be set, on which the almost-full flag of this channel is issued. Also a limit on the number of filled bytes can be used to set the almost-full flag.

If one of the memories of the input channels is flagged to be almost full, a busy signal is asserted to stop the CLOAC sending triggers.
The decoded data can be read out directly, however this is only meant for debugging purposes. During normal operation the decoded data is used to fill internal histograms, allowing the system to accumulate the data resulting from a burst of triggers. When a burst is finished, the DAQ program running on the PC reads out the histograms.

5.2.1.2 Optical readout

Since NIKHEF will have to mount and test the modules on discs, it was decided to use optical read-out to test the prototype modules, in order to gain experience with this. Figure 5.3 shows the main components.

Next to the main components, three different tools are used with the set-up. A simple circuit with a laser diode has been made which is used in combination with a signal generator to generate test signals. A circuit with a light sensitive diode has been made, which is used to display the optical signals onto an oscilloscope. Also, an optical power meter has been acquired to calibrate the plug-ins. These tools have been extremely useful to find problems in the set-up as well as in the prototype modules.

5.2.1.2.1 Rudge optical receiver  The optical receiver used is a prototype from CERN that was developed by Alan Rudge, and is referred to as the Rudge optical receiver. This receiver was a test board to try out a MITEL p-i-n diode array with an MT connector. MT is a standard connector used in the telecom world which combines up to 12 fibres into a single connector. The Rudge receiver converts the optical signal into an electrical signal, then amplifies, shapes and discriminates it to give a logic signal suitable for the MuSTARD. For convenience, the receiver has been converted to fit in a NIM (Nuclear Instrument Module) bin to use standard NIM power-supply.

The analogue signal can be probed after the amplification and shaping. The threshold can be set; however, there is only one threshold for all eight channels. This has the disadvantage that the optical signal has to be checked for each combination of optical plug-in and module. An optical plug-in uses a VCSEL to convert a current into an optical signal (see the description
in Paragraph 5.2.1.2.3). The optical power produced as function of the supply current varies significantly between individual VCSELs. The supply current is generated by the VDC (VCSEL Driver Chip) on the module. A control voltage that is supplied to the hybrid determines the amplitude of the current signal feeding the VCSEL. The amplitude of the current produced by the VDC can also be different for different modules.

To set the threshold of the Rudge receiver, the following procedure is followed. The module is powered and the ABCD chips are clocked but not configured. In this state the module generates a 20 MHz clock (the so-called ‘clock divided by two’) on the two outputs. The analogue output of the Rudge receiver is viewed on an analogue oscilloscope, and the module’s control voltages are adjusted to make the signals as large as possible while keeping them both equal. Then the threshold is set such that the digital outputs show a 20 MHz clock with a 50 percent duty cycle on both channels. The control voltages of the VDC on the module may need to be fine-tuned to achieve this.

5.2.1.2.2 Biphasic Modulation

The electronics needed to encode the clock, combine it with the command signals, and convert them into an optical signal were not available when the single module read-out system was built; so NIKHEF built its own multi-channel unit.

Biphasic modulation enables sending both the clock as well as the commands over one signal line. The principle of biphasic modulation (BPM) is demonstrated in Figure 5.4 (a). The clock is reduced to half the frequency, 20 MHz in this case. Both edges of the 20 MHz clock are used to regenerate the clock at the hybrid. If there is a data bit, the BPM signal is a 40 MHz modulation.

The logic to encode the data and clock runs on a clock that is twice the frequency of the clock sent to the hybrid. Each logical time bin is indicated in Figure 5.4 (a). The BPM signal is constructed from the so-called switch signal, which itself is constructed from the clock and the command signals. If the switch signal is one, the value BPM of that time bin is different from the one before, otherwise it is the same. The diagram in Figure 5.4 (b) shows the logic needed to construct this signal from the clock and the data signals, and this logic is recognised as the logic of an OR port. The logic needed to determine the value of a time bin of the BPM signal thus depends on the value of the switch signal of that time bin and of the value of the previous time bin of the BPM signal itself. Figure 5.4 (c) shows the logic needed to produce the BPM signal and is recognised as the logic of an XOR port. Figure 5.4 (d) shows the circuit diagram with the two logic ports needed to produce the BPM.

Two remarks need to be made. First of all, the phase of the BPM is undefined, since the first time bin is undefined. The absolute phase of the BPM signal is not important, since it encompasses both the clock and the command signal. Secondly, the processing time of the XOR port needs to be fast enough so that its output can be used for the next time bin. This prevents using standard NIM-modules to do the encoding of the data and the clock signal.

Due to the speed requirement, it was decided to have a special box made by the NIKHEF electronics department that can handle up to six clock and command streams and outputs six BPM optical signals. Appendix A gives the electronic diagrams of one BPM channel.

Figure 5.5 shows a simplification of these diagrams. A roboclock receives and reproduces the clock. It also generates the 80 MHz clock needed for the logic. A programmable GAL (Gate Array Logic) provides this logic. The GAL also uses a delay that can be set to the correct value to feed back the BPM signal. This is the main part. The other elements in the diagram are used to
provide a NIM\textsuperscript{2} output, an LVDS\textsuperscript{3} output and an 840 nm laser light output produced by a VCSEL. The light output is enough to damage the eye when the retina is exposed for a few seconds. This is a non-negligible risk since the light is in the invisible (infrared) part of the spectrum and one will therefore not look away by reflex. The retriggerable chip ensures that the signal is zero when there is no clock. The interlock ensures that when no fibre is mounted onto the laser, there is no light output. A special laser driver chip drives the laser.

\textsuperscript{2}NIM stands for Nuclear Instruments and Methods and is a standard current signal type.

\textsuperscript{3}LVDS stands for Low Voltage Differential Signalling and is used between the VME modules and on the hybrid.
5.2.1.2.3 **Optical plug-in** The optical plug-in was also not available when the single module read-out system was built. Figure 5.6 shows two drawings and a picture of an optical plug-in. The plug-in consists of one small electronics board and a piece of plastic to hold the fibres. For safety the fibres sit within a small metal capillary near the plug-in. The two VCSELs sending the data and the diode receiver are commercial components. To make them small enough the original housing is removed.

The main difficulty in making these plug-ins is to get a good alignment of the fibres with respect to the diode and VCSELs. First, the fibres are glued into the plastic housing. Then the electrical components are glued, held by a pair of tweezers that can be accurately moved by a set of micrometers. During the curing of the glue the signal transmission between fibre and electrical component (VCSEL or diode) is measured and optimised by using the micrometers to align the fibres with respect to the components. A small wire is used to make a temporary contact.

### 5.2.2 Detector Control System

The Detector Control System (DCS) consists of commercial components. The high voltage needed to bias the sensors is supplied with a Keithley 2410 1100V sourcemeter. The power and control voltages needed for the ABCDs, the VDC and the DORIC are supplied using two ‘TTi PL330TP triple power supplies’. These power supplies have very low noise (less than 5 mV RMS). All the supplies are connected via a patch panel to a DB-25 connector with the same pin-out as the ATLAS SCT VME power supply units. It is therefore easy to change to ATLAS SCT VME power-supplies when required. The power is supplied to the module testbox via a flat cable. This cable is kept as short as possible (approximately 1 meter), but is unshielded.
5.2. Hardware

Figure 5.6: Figure (a) shows a drawing of an optical plug-in. Figure (b) shows a (partially cut) side view of the optical plug-in when mounted onto a module hybrid. The units in the two drawings are in millimetre. The diode and the VCSELs are from Honeywell. Figure (c) shows a photo of one of the optical plug-ins made.

The power supplies are controlled and monitored via a GBIP interface to a PC, running a Labview program. This DCS also reads out a multichannel digital multimeter, which is connected to a PT1000 (temperature sensor) on the hybrid and three temperature sensors on the module testbox. These last sensors monitor the cooling system used to cool the hybrid.

All currents, voltages and temperatures that are measured are stored in a simple text file with the time of the measurement. The measurement interval is approximately 1 minute.

5.2.3 Module testbox

Figure 5.7 shows the testbox used in the single module readout system. Only inner modules fit. The testbox is designed to have easy access for a human hand to manipulate the module and the screws to hold the module are large and easy to handle; this minimises the risk of damaging a module.

The module testbox also holds a so-called Melbourne support card, which passes the power to the module hybrid via a small Kapton cable. All power supplies are floating and the individual grounds are connected together on the hybrid. To avoid noise injected due to capacitive coupling of the module with the box, the digital ground is connected to the module box from the support card. The mains grounds of the power supplies, as well as the mains grounds of the read-out system, are connected directly to the same ground to avoid any external ground loops. The Melbourne support card also allows for electrical read-out of the module, but this is not used at NIKHEF.
A small patch panel connects the optical fibres from the Rudge optical-receiver and the BPM encoder to the optical plug-in.

The module is actively cooled using a Peltier element of 70 Watts input power, which is mounted on the bottom of the testbox as close as possible to the module cooling block. When running, this gives a cooling block temperature of 20 °C and a hybrid temperature of 45 °C. Without cooling the hybrid is more than 70 °C. The warm side of the Peltier is cooled with cooling ribs and a fan. Although this provides a way of operating the module without the risk of overheating the chips, the temperature of the module is high with respect to the expected running temperature in ATLAS. However, the results presented in Chapter 7 will be corrected for this, using the model that is presented in Chapter 6.

When the lids are mounted the box is light tight. The testbox has four legs, which are of the correct height to use in combination with a small probing station. In this way, the front side of the module can be probed during operation. Special lids are available with thin aluminium windows for use with sources. The testbox has been carefully designed to be usable in the testbeam facility at CERN (with legs, cooling and optical patch panel removed).

### 5.3 Software

This section describes the software controlling the DAQ and used for the data analysis. Following an overview, the methods used to analyse the data are give in more detail. Binary read-out forces the user to use statistical analysis techniques to acquire information about the analogue performance of a chip. It was found that the default methods used in the SCT community had systematic errors which showed up in precision measurements, when trying to determine the deviation of the distribution of the noise of a module from a Gaussian. Improved methods have been developed
5.3. Software

at NIKHEF and are given here.

5.3.1 Overview of SCTDAQ

![Diagram of SCTDAQ software components]

SCTDAQ (Phillips et al. 1999) is the standard data-acquisition software used within the SCT community. Figure 5.8 shows an overview. The top-level routines (STDLL) are written in C++ and heavily use the ROOT histogramming and analysis package (Brun & Rademakers 1996). Lower level routines (STLIB, MuSTARD etc.) are written in C, allowing them to also be used in the CERN Testbeam data-acquisition system.

The library labelled 'VME' in Figure 5.8 contains the functions to access the registers of the VME modules. This depends on the interface between the PC and the VME-crate used. At NIKHEF, the National Instruments library supplied with the so-called 'PCI-VME' card that is used. The libraries CLOAC, SLOG, and MuSTARD contain the basic routines to use the corresponding VME-modules. The library STLIB (System Test LIBrary) has functions to control the different modules coherently with one another.
The library STDLL (System Test Dynamic Link Library) is the heart of the data-acquisition software. It has a module class to store the required settings of a module and the way that module is connected to the system. It calls STLlib routines to make scans (see Section 5.3.2). It is dynamically linked to ROOT, and its functions and classes are made available in CINT, the ROOT C++ interpreter.

STDLL reads two types of configuration files: system configuration files, detailing how modules are connected to the data-acquisition system; and module configuration files, detailing the initial module settings.

The system is controlled via the CINT interpreter: either interactively, via macros, or via the graphical push-button SCTDAQ menu-bar. The menu bar allows common tasks such as setting parameters or making scans to be carried out. A set of standard test macros allow more sophisticated control of the hardware. In particular, there are two high-level macros to carry out the Confirmation Test Sequence and the Characterisation Test Sequence (see Sections 5.4.2 and 5.4.3). These macros also make use of analysis macros, which process the histogram data, adding fits to the histograms, calculating summary data, and making graphical plots.

The main output is in three types of files. The basic histograms are stored in ROOT histogram files, which can have fit results added by analysis macros. The plots are (usually) stored in postscript form. Summary data is stored in ASCII files, in a format which can easily be uploaded to the SCT database using a Java script.

5.3.2 Performance analysis

The basic data set acquired with a data-acquisition run is a scan. A scan results in a two-dimensional histogram. One histogram axis contains a bin for each channel. The other axis contains a bin for each value of the module parameter varied in the scan (e.g. threshold). Only one parameter is changed during the scan. The histogram is filled with the number of hits found for each setting and each channel after sending a large number of triggers. In the analysis, a one-dimensional histogram is extracted from this, containing the number of hits versus the scanned parameter for a single channel. In addition, a one-dimensional histogram stores the number of triggers sent for each setting of the parameter.

The scan that is most used to evaluate the analogue performance is the threshold scan. A known charge is injected at the input of a channel and a known number of triggers are read out to evaluate the response. This is repeated for a range of thresholds.

This section discusses how to evaluate the histogram resulting from a threshold scan for a single channel. Since one binary data is available, it is important to have an analysis that is sensitive to small, may be unexpected, effects. This analysis is used in Chapter 7.

5.3.2.1 The S-curve

Figure 5.9 shows a possible result for a single channel of a threshold scan and the statistics behind it. The measurement in Figure (a) is generally referred to as an S-curve (although the name Z-curve may be more appropriate).

Setting the threshold at a certain voltage, say $V_{\text{thr}}$ (see Figure 5.9 (b)), the probability $p$ that
Figure 5.9: Illustration of an S-curve. Figure (a) shows a possible result of a threshold scan for a single channel, using 1000 samples per bin. Figure (b) shows the underlying statistical principles leading to this result. The solid line shows a possible signal-height pdf for an ABCD channel at the comparator. The dashed line shows the hit probability: the probability that the signal is larger than the threshold, as a function of threshold setting. A possible threshold setting is $V_{\text{thr}}$ (dotted line), and the shaded area illustrates the fraction of events above threshold.

A channel registers a hit can be expressed as:

$$p(V_{\text{thr}}) = \int_{V_{\text{thr}}}^{\infty} f_{\text{spd}}(s) \, ds$$  \hspace{1cm} (5.1)

where $f_{\text{spd}}(s)$ is the signal height probability-distribution function (pdf) and $s$ is the signal height. Figure 5.9 shows a Gaussian $f_{\text{spd}}$ and illustrates the integral as the grey area; the dashed line shows the hit probability $p$ as a function of the threshold.

The signal pdf is the convolution of the input-signal pdf and the input-noise pdf. In an ideal system with no noise, the signal pdf due to charge injection would be a $\delta$-peak and the resulting hit probability function would therefore be a step function. Adding noise smears $f_{\text{spd}}$, but from Equation 5.1 and $f_{\text{spd}}(s) > 0$ it always follows that the hit probability-function is a monotonically decreasing function. However this assumes that the signal amplitude distribution is independent of the threshold setting; in particular, feedback effects can make $p(V_{\text{thr}})$ non-monotonic, as discussed in Section 7.3.2.

We take the average signal-height to be the threshold for which $p = 0.5$, i.e. we use the median signal-value as the ‘response’. For a Gaussian signal-distribution, or any other symmetric monomodal distribution, the median coincides with the mean and the maximum of the distribution (the mode). The width of the hit-probability function is a measure of the RMS of the noise amplitude. To evaluate this more precisely, the exact noise distribution has to be known.
5.3.2.2 Maximum likelihood fit to S-curves

Let's return to the measurement of the hit probability function, as given in Figure 5.9 (a). The measurement of the hit probability at a given threshold is essentially a 'hit-or-miss' experiment and the measured number of hits follows binomial statistics. Therefore the probability of measuring \( n \) hits after reading out \( N \) times with a hit probability \( p(V_{thr}) \) is given by the binomial distribution:

\[
\text{prob}(n) = \binom{N}{n} p^n (1 - p)^{N-n}
\]

The expectation value of the number of triggers and the mean-square deviation are:

\[
\langle n \rangle = N p
\]

\[
\langle (n - \langle n \rangle)^2 \rangle = N p (1 - p)
\]

Let's assume the hit probability is a known function of a set of parameters \( a \), so \( p = p(V_{thr}, \bar{a}) \), and \( \text{prob} = \text{prob}(n; \bar{a}) \).

Bayes' theorem provides a way to compute the probability that \( \bar{a} \) represent the underlying parameters resulting in the measurement of \( n \). In general, Bayes' theorem says:

\[
\text{prob(hypothesis|data, knowledge)} = \frac{\text{prob(data|hypothesis, knowledge)} \times \text{prob(hypothesis|knowledge)}}{\text{prob(data|knowledge)}}
\]

Or, in words, the probability for a certain hypothesis given some data is equal to the product of the probability of the data given the hypothesis and the probability of the hypothesis divided by the probability of the data. 'Knowledge' represents everything the user knows, which can have important effects. The probability of the hypothesis in the absence of data is known as the prior. This can be simplified to:

\[
\text{prob(hypothesis|data, knowledge)} \propto \text{prob(data|hypothesis, knowledge)}
\]

assuming the probability of the data is not influenced by the user's knowledge and a flat prior. The first assumption can be interpreted as "the data is not manipulated". The second assumption can be interpreted as "all values for the noise and the signal are assumed equally possible, before taking the measurement".

It is usually much easier to evaluate the r.h.s. of Equation 5.6 than the l.h.s; for the current problem we quantify the probability that a set of parameters \( \bar{a} \) is the correct hypothesis given a measurement of \( n_i \) hits out of \( N_i \) triggers for some particular threshold labelled by \( i \) as:

\[
\text{prob}(\bar{a}|n_i) \propto \text{prob}(n_i|\bar{a}) = \binom{N_i}{n_i} p_i^{n_i} (1 - p_i)^{N_i-n_i}
\]

where we have used Equation 5.2.

For a series of measurements at different thresholds the probability that \( \bar{a} \) describe the S-curve is the product of the probabilities for each threshold:
\[
\text{prob}(\bar{a} \mid \{n_i\}) \propto \prod_{i=1}^{n_b} \left[ \binom{N_i}{n_i} p_i^{n_i} (1 - p_i)^{N_i - n_i} \right]
\]  

(5.8)

where \(n_b\) is the number of bins (i.e. the number of thresholds) used in the scan.

We can now find the most-probable values for \(\bar{a}\) by maximising \(\text{prob}(\bar{a} \mid \{n_i\})\). In practice, we minimise minus the logarithm of the probability:

\[
- \ln (\text{prob}(\bar{a})) = \sum_{i=1}^{n_b} \left\{ c - \ln \binom{N_i}{n_i} - n_i \ln(p_i) - (N_i - n_i) \ln(1 - p_i) \right\}
\]

(5.9)

where the constant \(c\) accounts for the proportionality in Equation 5.8. Both this constant as well as the term \(\ln \binom{N_i}{n_i}\) can be neglected, since they are independent of the assumed values for \(\bar{a}\). Hence the function:

\[
\text{FCN}(a) = - \sum_{i=1}^{n_b} \left\{ n_i \ln(p_i) + (N_i - n_i) \ln(1 - p_i) \right\}
\]

(5.10)

is minimised to find the most likely value of \(\bar{a}\).

### 5.3.2.3 S-curve fitting under the assumption of Gaussian noise

The noise in a silicon strip detector in an ideal system is determined by shot noise and thermal noise sources, which have a Gaussian amplitude distribution (see Chapter 6). Approximating the detector signal by a \(\delta\)-pulse, the convolution of the signal and noise amplitude probability distribution is also a Gaussian. The resulting hit probability, setting a threshold at \(V_{\text{thr}}\), is from Equation 5.1:

\[
g(V_{\text{thr}}; \mu_s, \sigma_s) = \frac{1}{\sigma_s \sqrt{2\pi}} \int_{V_{\text{thr}}}^{\infty} e^{-(s - \mu_s)^2 / 2\sigma_s^2} ds
\]

(5.11)

where the mean \(\mu_s\) of the signal pdf is the detector signal and the width \(\sigma_s\) is the RMS noise. Using the definition of the error function:

\[
\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du
\]

(5.12)

Equation 5.11 can be rewritten as:

\[
g(V_{\text{thr}}; \mu_s, \sigma_s) = \frac{1}{2} \left( 1 - \text{erf} \left( \frac{V_{\text{thr}} - \mu_s}{\sqrt{2}\sigma_s} \right) \right)
\]

(5.13)

In summary, assuming a \(\delta\)-like signal pulse and Gaussian noise, we obtain the signal \(\mu_s\) and RMS noise \(\sigma_s\) by minimising FCN in Equation 5.10 with \(p_i \equiv g(V_{\text{thr}}; i; \mu_s, \sigma_s)\) in Equation 5.13.

To find the minimum the program MINUIT (James & Roos 1975) is used. To calculate the error function the CERNLIB implementation (Cody 1969) is used.
Figure 5.10: The results of fitting simulated data using Equation 5.10. The parameters of the data are given in the text. Figure (a) shows the average residual of the 1536 S-curves as function of the distance from $\mu_s$ in units of $\sigma_s$. The error bars indicate the average spread of the residuals. The lines indicate the expected error according to Equation 5.4. Figure (b) shows the pull distribution of the response $\mu_s$. Figure (c) shows the pull distribution of $\sigma_s$ for the 1536 S-curves.

Figure 5.11: Same as the above figure, only now the original SCTDAQ fit method has been used. Figure (a) shows a clear systematic deviation of the average residuals from 0. Figure (c) clearly shows that the width of the S-curve is incorrectly fitted.

The fit method has been tested using simulated data. The necessary S-curves have been generated by comparing the random amplitude of a signal, which had a Gaussian distribution, with a threshold. If the signal amplitude was higher than the current value of the threshold, a hit was registered. All parameters have been chosen to be close to reality. The mean of the Gaussian signal amplitude distribution is 83.0 and the width is 9.3. The threshold has been varied between 0 and 150, with a step size of 2.5. Each threshold setting has been sampled 1000 times. In total, 1536 S-curves have been generated in this way and fitted using Equation 5.10. Figure 5.10 shows the residuals and the pull distribution of the response and the noise. The pull is defined as $(\text{generated-fit})/\sigma_{\text{fit}}$. These pull distributions (Figure 5.10 (b) and (c)) show that the correct values of the response and noise are found, with the correct error. Both pull distributions are
well fitted with a Gaussian. The distribution of the errors is also understood, as can be seen from Figure 5.10 (a).

SCTDAQ originally used a $\chi^2$-fit method where the error on each bin is calculated using binomial statistics, i.e. Equation 5.4. Figure 5.11 shows the results for this obsolete fit method. The residual distribution shows a systematic structure. Below $\mu_s$ the fitted curve is systematically lower than the data, while above $\mu_s$ the fitted curve is systematically higher than the data. This is also reflected in the pull distributions. The pull distribution for $\mu_s$ is correct, although the error seems to be slightly overestimated. The mean of the pull distribution for $\sigma_s$ is clearly negative, indicating that the fitted signal distribution is too wide, i.e. the fitted noise is too high. This is a significant systematic effect, in the order of the fit error. The error is clearly overestimated by almost 10%.

The deviations in Figure 5.11 can be explained. If the number of hits in a bin is equal to the number of samples taken, the error according to Equation 5.4 is zero. However, it is possible that in the extremes of the S-curve, the bin content is larger than zero or less than the number of samples. If the neighbouring bins, on the side towards $\mu_s$ have a zero error, they are not used in the $\chi^2$-fit method. Thus, the fit will be ‘pulled’ to the non-zero bin, resulting in a too large value for $\sigma_s$.

The correct fitting method has the extra advantage that it is significantly faster. And, since MINUIT is a very general minimisation program, a specific minimisation routine for this problem probably would make the correct fitting method even faster.

5.3.2.4 Determining systematic deviations from Gaussian noise

In the previous section the main assumption made was that the signal-pdf is Gaussian, implying that the noise amplitude has a Gaussian distribution. To test this assumption, the goodness of fit needs to be quantified. The $\chi^2$ normalised to the number of degrees of freedom (NDF) is used to do this, which is calculated using Equation 5.4 for the error, with $p_i \approx y_i^{\text{data}}/N_i$:

$$\chi^2/\text{NDF} = \frac{1}{N_p - 2} \sum_i \frac{(y_i^{\text{fit}} - y_i^{\text{data}})^2 N_i}{y_i^{\text{data}} (N_i - y_i^{\text{data}})}$$

(5.14)

where $y_i$ is the value of the point labelled $i$ given by either the data or the fit, $N_i$ is the number of triggers used for the measurement of this data-point, and $N_p$ is the number of points.

Since in principle a $\chi^2$ assumes Gaussian errors, the distribution has been evaluated for simulated data (see Section 5.3.2.3). It was found that the $\chi^2$/NDF distribution correctly has a mean value of one and that this is independent of the number of samples per threshold and the threshold step.

It is common practice to use the probability of the fit, which can be directly related to the $\chi^2$/NDF. The probability is not used in this case for two reasons. First of all, especially the first prototype modules discussed in this thesis do not have Gaussian noise. Secondly, in a module with problems, the distribution of $\log(\chi^2$/NDF) behaves well enough to observe structures along chips (see for example Figure 7.15).
5.3.2.5 Amplifier Gain Measurement

The variable \( \mu_s \) is also called the response. When injecting a known charge in the input of an ABCD channel (see Section 3.3.3) \( \mu_s \) is a measurement of the output of the analogue part of the ABCD in response to the injected charge. If the response is measured for several input charges, the resulting plot of response versus input charge is called the response curve.

We use the following empirical function (known within the SCT community as the Grillo-function, named after the person who introduced it) to fit the response \( R \) for a charge injection \( q \):

\[
R = p_0 + \frac{p_2}{\sqrt{1 + \frac{p_1^2}{p_2^2} q^2}} \tag{5.15}
\]

The three fit parameters are: \( p_0 \) which is the offset (the output of the analogue part of the ABCD when there is no signal); \( p_1 \) which is the small-signal gain (see below); and \( p_2 \) which is a measure of the non-linearity.

The gain \( G \) is defined as the derivative of the response curve:

\[
G(q) = \frac{p_2^3}{p_1^2 \left(1 + \frac{p_2^2}{p_1^2} q^2\right)^{3/2} q^3} \tag{5.16}
\]

and the small signal gain is:

\[
\lim_{q \to 0} G = p_1
\]

Knowing the gain, the output noise \( \sigma_n \) in mV can be translated to input noise in fC, making a comparison with both the expected signal and with the noise calculated in Chapter 6 possible. Also, this makes a noise comparison between chips possible, because the gain is likely to vary between different chips and even between different channels. The usual units used are response in mV, injection charge in fC, and gain in mV/fC. Input noise charge is usually expressed in number of electrons.

5.4 System operation

5.4.1 Test procedure

During the prototyping of the module a set of tests have been developed and standardised. These tests will be used to achieve two goals during the module production: characterisation of the module performance and confirmation of the module performance.

Two types of tests are used during production: Section 5.4.2 presents the Confirmation Test Sequence and Section 5.4.3 presents the Characterisation Test Sequence. The last part of this section describes the individual tests. These tests are in two classes, corresponding to the part of the chip being tested: digital tests and analogue tests.

Results of the prototype modules are presented in Chapter 7. A module is excepted if at least 99% of its channels will operate efficiently with low noise occupancy at a threshold setting of 1 fC (Dolezal 2001), however this is likely to evolve with experience.
Table 5.1: The default configuration of the chip when testing.

Unless mentioned otherwise, in all tests each link is configured to read out all chips in data-taking mode, and the hybrid is controlled via the standard command link. The analogue supply voltage $V_{cc}$ is 3.5 V and the digital supply voltage $V_{dd}$ is 4.0 V. The standard module-settings are given in Table 5.1. Detailed information can be found in (Phillips 2002).

### 5.4.2 Confirmation Test Sequence

The Confirmation Test Sequence verifies that no damage has occurred on a hybrid or module that has been undergoing some handling and/or shipping. The expected faults are mechanical: disconnected bonds or damaged silicon surfaces (chip or detector). In order to make sure that all the connections of the modules are intact, three digital tests are performed:

- Reset Test;
- Redundancy Test;
- By-pass Test.

In order to detect any damage to the sensors or to the input connection of the chips, two of the analogue tests are performed:

- Strobe-Delay Test;
- Three-Point Gain Test.

During the testing the power-supply currents of the hybrid and the bias currents of the sensors are stored and compared to previously measured or nominal values.
5.4.3 Characterisation Test Sequence

The Characterisation Test Sequence needs to fully quantify the detector module, so all the tests are performed. First the digital tests are performed in the following order:

- Reset Test;
- Redundancy Test;
- By-pass Test;
- Pipeline Test;

Next, the analogue tests are performed in the following order:

- Strobe-Delay Test;
- Trim Test;
- Ten-Point Gain Test;
- Noise-Occupancy Test;
- Time-Walk Test.

During the testing the power-supply currents of the hybrid and the bias currents of the sensors are stored and compared to previously measured or nominal values.

5.4.4 Digital tests

5.4.4.1 Reset Test

This tests the initialisation of the module and the response to a hard reset.

The data-acquisition system is initialised, so that the module receives a clock. The module power is turned on. The user checks via an oscilloscope connected to the MuSTARD spy channels that the module returns clock/2 (20 MHz clock, generated by the chips from the 40 MHz clock) from both links. Then the configuration register is set to the standard values, except with edge detect on and the data compression is 01X, mimicking the default settings when reading out the module. The user checks that the clock/2 signal disappears, and records the digital and analogue supply currents. These currents are close to those expected at ATLAS, due to the configuration of the module, and both supply currents should be recorded.

Next, a hard-reset command is issued and the user checks that the clock/2 signal returns to both links and records both supply currents. Finally, the clock is turned off and the user again notes the supply currents.
5.4.4.2 Redundancy Test

The response to commands from both the standard and redundant command-lines is tested.

The hybrid is configured in the default way, except the chips are configured to read-out the mask register. The module is triggered and read-out 100 times. Before each trigger the mask register is alternately loaded with a '101010...10'-pattern and a '010101...01' pattern. All channels should record 50 hits. This test is repeated using SELECT=1 to test the redundant read-out.

5.4.4.3 By-pass Test

All possible by-pass routings on the hybrid for the case a chip fails are tested.

The hybrid is configured in the standard way, except the chips are configured to read-out the mask register and the mask register is loaded with all 1's. The module is triggered and read-out 100 times; all channels should have 100 hits. Then one of the possible by-pass routings is configured and tested; this is repeated for all 62 by-pass configurations.

Tests have shown that some irradiated chips do not correctly pass the token and this has been correlated with the need for a high $V_{dd}$ before irradiation. We check that the token passing between links works at low supply voltage by repeating the sequence at 3.9, 3.8, 3.7, 3.6 and 3.5 Volts.

5.4.4.4 Pipeline Test

The functionality of the ABCD pipeline is tested.

The hybrid is configured in the standard way, except the chips are configured to read-out the mask register and the mask register is loaded with all 1's. A soft reset is issued, setting the write-pointer of the pipeline to the first bin and after a specific number of bunch crossings a pulse-input command is issued, writing a '1' at a specific cell of the pipeline. By changing the timing between the soft reset and the pulse-input command different pipeline cells are tested. For each cell this is repeated 50 times. Then the mask register is loaded with all 0's, and again each cell is tested 50 times. For each cell 50 hits should be registered. By writing both 1's and 0's both dead and stuck cells can be identified.

An incorrect number of hits found in this test could also be due to a mask failure. This test can be combined with the redundancy test to identify dead or stuck cells in the mask register.

5.4.5 Analogue tests

The analogue tests are performed using the default settings of Table 5.1, except that:

1. The calibration pulse and threshold settings are specific to each test;

2. The strobe delay, trim range and trim setting are left at their optimised values once they have been determined.

The standard number of triggers at each setting is 1000.
5.4.5.1 Strobe-Delay Test

The calibration-strobe delay delays the calibration pulse with respect to the arrival time of the command to issue a pulse. It is adjusted to optimise the timing of the output of the comparator to the latch (see Figure 3.13) which is generated by the clock.

First, a threshold to use in this test has to be found. The Strobe-Delay Test uses a calibration-pulse of 40 mV, with the threshold set at the response to a calibration pulse of 20 mV, so the hit-probability is very high. To find this response, the default settings are used, except the strobe delay is set to a nominal value to ensure a reasonable timing of the calibration pulse. The calibration-pulse is set at 20 mV and a threshold scan is performed (see Section 5.3.2). The S-curves are fitted and the average response is used to set the threshold, chip-by-chip.

The default settings are used, except that edge-detection is turned on, data compression is set to 01X, and the calibration-pulse is set to 40 mV. A *strobe-delay scan* is performed: the strobe delay is set to 0; the set charge is injected at the front-end; a trigger is sent and the hybrid is read-out. The strobe-delay setting is sampled 1000 times this way. Then the strobe delay is increased and the hybrid is again sampled 1000 times. This is repeated until the maximum strobe-delay setting is reached. The strobe-delay range corresponds to approximately two clock-cycles. If the strobe delay is too large, the signal arrives too late, and a ‘0’ is read-out. If the strobe delay is too small, the signal is already registered in the previous time-bin and the edge-detection circuitry rejects this hit. In between, in a region corresponding to one clock cycle, a hit is registered. The edges have some width, due to noise and time jitter. Figure 5.12 shows an example of the result of a strobe-delay scan.

The 50%-occupancy point of each edge is found using linear interpolation of the bin contents. The difference between these points corresponds to 25 ns (one clock cycle). The strobe delay is set to: the high strobe-delay edge - 0.75×25 ns (75% of the distance between the two edges of the strobe-delay curve), which is expected to correspond to the peak position of the

![Figure 5.12: A typical result of a strobe-delay scan for a single channel. The star indicates the strobe-delay setting that will be selected.](image-url)
response of the amplifier-shaper circuit to the calibration pulse.

5.4.5.2 Trim Test

This test sets the trim-DAC and trim-range to ensure a minimal response-deviation between all channels of a single chip when injecting a 1.0 fC calibration charge. In this way, if the threshold is set to this response, as is expected to be done in ATLAS, the expected efficiency for all channels is the same.

A newly built module is expected to have only a small number of channels that cannot be optimised using the smallest trim-range. After irradiation of the module it is found that the spread on the response within one chip grows and larger trim-range settings have to be used to get a similar response. In this test, the default settings of the chips are used. The calibration-DAC is set to 10 mV, the nominal setting to produce a calibration pulse of 1 fC. Using this calibration pulse, a threshold scan for all 16 trim-DAC settings using the smallest trim-range is taken to determine the response of each channel for each setting of the trim-DAC. To reduce the testing time, the other three trim-ranges are only probed at 4 trim-DAC settings.

For each trim-range setting and each channel, the response is plotted against the trim-DAC setting and fitted with a straight line. Then, using the fit results, the number of channels in a chip that can be set to have a certain target response is plotted against the response, showing the target response for which the most channels can be trimmed. If a chip has more than one target-response that has the maximum number of trimmable channels, which is usually the case, the one with the least spread in actual response between channels is chosen.

The trim-DAC settings of all trimmable channels for this response are recorded and the trim-DACs set to these values. Currently, if a channel cannot be trimmed, it is masked.

5.4.5.3 Gain Test

The Gain Test measures and fits the response curve, as described in Section 5.3.2.5.

There are two standard Gain Tests. The Three-Point Gain Test uses calibration pulses of 15, 20 and 25 mV. The resulting response curves are fitted with a straight line. The Ten-Point Gain Test uses calibration pulses of: 5, 7.5, 10, 12.5, 15, 17.5, 20, 30, 40 and 60 mV.

The default Gain Test used in this thesis when discussing module results is the Ten-Point Gain Test, except the first two responses are ignored in the fit of the response curve. This will be explained in Section 7.4.2.

5.4.5.4 Noise-Occupancy Test

This test measures the noise occupancy, i.e. the occupancy with no calibration pulse and with the threshold set to the response of a 1 fC calibration charge.

The default settings are used for this test. A threshold scan is made with no charge injection. The number of triggers sent is varied between 2,000 and 1,000,000 according to the occupancy.

The Noise-Occupancy at a threshold of 1 fC, which is known from the gaintest and adjusted by the capacitor correction-factor, is determined and compared to the specification, which is $5 \times 10^{-4}$.
5.4.5.5 Time-Walk Test

Signals caused by different input charges at the comparator cross the threshold at different times. Figure 5.13 illustrates this effect. Time Walk is the comparator threshold-crossing time difference between two signals of different size. Ideally, in ATLAS the clock sent to a module will be timed such that the strobe coincides with the peak of the signal. In this case, a signal just crossing the threshold will still produce a ‘1’ (see Figure 5.13). However, the time walk should not be too large, or the deposit of a large amount of charge in the silicon could cause the comparator to produce a ‘1’ at the time-bin before the bin that is read out. The edge-detection circuitry would then reject the hit.

The time walk is measured using the charge injection circuitry. The default settings are used, but the edge detection is turned on, the compression mode is set to ‘01X’, and the threshold is set at 1 fC. These are the settings that are expected to be used at ATLAS. For a set of calibration charges between 1.25 fC and 10 fC the strobe-delay scan is performed (see Section 5.4.5.1). The edge of the resulting hit-probability curve at the high strobe-delay is a measure of the point where the threshold is crossed. The position of this edge at 1.25 fC is subtracted from its position at 10 fC to give the time walk in units of strobe-delay settings. This unit can be converted to nanoseconds, knowing that the rising and falling edges of the hit-probability curve resulting from a strobe-delay scan should be one clock-cycle apart, i.e. 25 ns.

The time walk should be less than 16 ns.