Characterization of an ADC

Carl-Fredrik Tengberg
Master of Science Thesis in Electrical Engineering

Characterization of an ADC
Carl-Fredrik Tengberg
LiTH-ISY-EX–17/5065–SE

Supervisor: Dr. Erik Backenius
Examiner: Dr. J Jacob Wikner

Division of Integrated Circuits and Systems
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Copyright © 2017 Carl-Fredrik Tengberg
Abstract

Analog-to-Digital converters, ADCs, introduces the possibility of performing digital computation on real world analog signals by being the interface between the two domains. The demands on the performance of the ADC is steadily increasing, which also comes with an increased difficulty of actually being able to guarantee the functionality of the device. Therefore, the art of estimating the characteristics of the ADC has flourished over the years where researchers has suggested ways of performing different test procedures to arrive at the most accurate method suited for a specific application.

In this thesis, a measurement device containing an ADC is the subject of investigation. Some of the static characteristics of the ADC were estimated by isolating the ADC in the circuit and putting it through a set of tests developed in a lab environment and analyzing the resulting data offline. By then also carrying out measurements of the whole system, the amount of input referred noise added by the ADC could be estimated.
## Contents

List of Figures vii  
List of Tables ix  
Notation xi  

1 Introduction 1  
1.1 Introduction ............................................. 1  
1.1.1 Motivation ........................................... 2  
1.1.2 Problem statements ................................... 2  
1.2 Background .............................................. 2  
1.3 Analog-to-Digital Converter 4  
1.3.1 Registers in the ADC 7  
1.3.2 Offset mode ........................................... 7  
1.3.3 ADC gain ............................................. 8  
1.3.4 Common mode adjust 9  
1.3.5 Settings inside the comparator 9  
1.4 Characteristics of the ADC 11  
1.4.1 Common mode input range 11  
1.4.2 Gain 12  
1.4.3 Input referred shift and offset 13  
1.4.4 Input referred noise 14  

2 Method 17  
2.1 The test setup 17  
2.1.1 The LabView Software ................................ 20  
2.2 Register combinations .................................... 24  
2.3 Estimating the characteristics of the ADC 26  
2.3.1 Common mode input range 28  
2.3.2 Gain estimation 29  
2.3.3 Input referred shift and offset estimation 30  
2.3.4 Noise estimation 30  
2.3.5 Nonlinearity estimation 31
2.4 Input referred noise estimation of circuit A .......................... 31

3 Results 33
  3.0.1 Common mode range ........................................... 33
  3.1 Gain ................................................................. 37
    3.1.1 Combinations two ........................................... 37
    3.1.2 Combinations three ......................................... 38
    3.1.3 Individual register sweep ................................ 39
    3.1.4 Filter ......................................................... 40
  3.2 Offset and shift .................................................. 40
    3.2.1 Combinations two ........................................... 40
    3.2.2 Combinations three ......................................... 41
    3.2.3 Individual register sweep ................................ 42
    3.2.4 Filter ......................................................... 44
  3.3 Input referred noise .............................................. 44
    3.3.1 Combinations two ........................................... 45
    3.3.2 Combinations three ......................................... 47
    3.3.3 Individual register sweep ................................ 49
    3.3.4 Filter ......................................................... 51
  3.4 Quantization noise ................................................ 52
    3.4.1 Combinations two ........................................... 52
    3.4.2 Combinations three ......................................... 53
    3.4.3 Individual register sweep ................................ 54
    3.4.4 Filter ......................................................... 55
  3.5 INL/DNL ................................................................ 56
    3.5.1 Combinations two ........................................... 57
    3.5.2 Combinations three ......................................... 58
    3.5.3 Individual register sweep ................................ 59
    3.5.4 Filter ......................................................... 61
  3.6 Input referred noise of circuit A ................................. 62

4 Conclusions 63
  4.1 Future work ......................................................... 64

Bibliography 65
List of Figures

1.1 Rough overview of the measurement device. ............................................. 1
1.2 Example of a differential SAR ADC. ...................................................... 4
1.3 Transfer curve of a differential ADC. ..................................................... 5
1.4 Capacitor switches set depending on the comparator decision. ................. 6
1.5 Principle of operation for offset mode. .................................................. 8
1.6 Transfer curves with different values of $V_{\text{offset}}$. ............................ 8
1.7 Transfer curves with different values of $V_{\text{high}}$. ............................... 9
1.8 Principle of operation of common mode adjust. ...................................... 9
1.9 Rough block diagram of the comparator. ................................................. 10
1.10 Two ways of determining the common mode input range. ....................... 11
1.11 Thresholds of the common mode input range. ....................................... 12
1.12 Different input ranges for a set of different ADCs. ............................... 12
1.13 Offset error. ....................................................................................... 13
1.14 Jittering code transitions due to the presence of noise. ............................ 14
1.15 DNL and INL. .................................................................................... 15

2.1 Test setup. ............................................................................................ 17
2.2 Connecting the Keithley 2701 via ethernet [2]. ...................................... 18
2.3 Using the Keithley 2701 multimeter with LabView. ................................. 18
2.4 Reading the IP address of the Keithley 2701 [2]. .................................... 19
2.5 Measuring the accuracy of the two voltage sources. ............................... 19
2.6 Pin out of the PXIe 6361 [4]. ................................................................. 20
2.7 Flowchart of the labview program. ......................................................... 21
2.8 Example of the test signal behaviour. ..................................................... 22
2.9 Example of a list of register writes used for sweeping register values. ....... 23
2.10 Extracting the measurement data for each input voltage. ...................... 23
2.11 Organizing the data. ............................................................................. 26
2.12 Cell array containing the file names together with the current setting combination, ideal input voltages and measured voltage difference. ................................................................. 26
2.13 Conversion data stored in matrixes. ...................................................... 27
2.14 Plotting the mean value of the N conversions for each input voltage. ....... 27
2.15 Interpolating the data to find the transitions. ........................................ 28
2.16 Finding the common mode range in terms of lost codes. ....................... 29
2.17 Input referred noise of the signal chain. ................................................. 31
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Sweeping values of Latch current and comparator start.</td>
<td>34</td>
</tr>
<tr>
<td>3.2</td>
<td>Sweeping two of the filter coefficients.</td>
<td>35</td>
</tr>
<tr>
<td>3.3</td>
<td>Sweeping values of Pre-amp current, common mode adjust and ADC gain.</td>
<td>36</td>
</tr>
<tr>
<td>3.4</td>
<td>Input ranges of combination set two.</td>
<td>37</td>
</tr>
<tr>
<td>3.5</td>
<td>Input ranges of combination set three.</td>
<td>38</td>
</tr>
<tr>
<td>3.6</td>
<td>Input ranges of the ADC when sweeping through the settings individually.</td>
<td>39</td>
</tr>
<tr>
<td>3.7</td>
<td>Input ranges for different filter settings.</td>
<td>40</td>
</tr>
<tr>
<td>3.8</td>
<td>Input referred shift and offset for combinations two.</td>
<td>41</td>
</tr>
<tr>
<td>3.9</td>
<td>Input referred shift and offset for combinations three.</td>
<td>41</td>
</tr>
<tr>
<td>3.10</td>
<td>Input referred shift and offset for individual register sweeps.</td>
<td>42</td>
</tr>
<tr>
<td>3.11</td>
<td>Input referred shift and offset for individual register sweeps.</td>
<td>43</td>
</tr>
<tr>
<td>3.12</td>
<td>Shift and offset for different filter settings.</td>
<td>44</td>
</tr>
<tr>
<td>3.13</td>
<td>Input referred noise distributed over the full input range of the ADC.</td>
<td>44</td>
</tr>
<tr>
<td>3.14</td>
<td>Input referred noise (codes) of combinations two.</td>
<td>45</td>
</tr>
<tr>
<td>3.15</td>
<td>Input referred noise (volts) of combinations two.</td>
<td>46</td>
</tr>
<tr>
<td>3.16</td>
<td>Input referred noise (codes) of combinations three.</td>
<td>47</td>
</tr>
<tr>
<td>3.17</td>
<td>Input referred noise (volts) of combinations three.</td>
<td>48</td>
</tr>
<tr>
<td>3.18</td>
<td>Input referred noise of individual register sweeps.</td>
<td>49</td>
</tr>
<tr>
<td>3.19</td>
<td>Input referred noise of individual register sweeps.</td>
<td>50</td>
</tr>
<tr>
<td>3.20</td>
<td>Input referred noise for different filter settings.</td>
<td>51</td>
</tr>
<tr>
<td>3.21</td>
<td>Quantization noise of combinations two.</td>
<td>52</td>
</tr>
<tr>
<td>3.22</td>
<td>Quantization noise of combinations three.</td>
<td>53</td>
</tr>
<tr>
<td>3.23</td>
<td>Quantization noise individual register sweep.</td>
<td>54</td>
</tr>
<tr>
<td>3.24</td>
<td>Quantization noise for different filter settings.</td>
<td>55</td>
</tr>
<tr>
<td>3.25</td>
<td>Fitted data.</td>
<td>56</td>
</tr>
<tr>
<td>3.26</td>
<td>DNL and INL for combinations two.</td>
<td>57</td>
</tr>
<tr>
<td>3.27</td>
<td>DNL and INL for combinations three.</td>
<td>58</td>
</tr>
<tr>
<td>3.28</td>
<td>DNL and INL for individual register sweeps.</td>
<td>59</td>
</tr>
<tr>
<td>3.29</td>
<td>DNL and INL for individual register sweeps.</td>
<td>60</td>
</tr>
<tr>
<td>3.30</td>
<td>DNL and INL for different filter combinations.</td>
<td>61</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Register settings for individual sweeps. . . . . . . . . . . . . . . . . . 24
2.2 Register settings for combination two sweep. . . . . . . . . . . . . . 25
2.3 Register settings for combination three sweep. . . . . . . . . . . . . 25
2.4 Combinations of pre amp current and filter coefficient A. . . . . . . 25

3.1 Input referred noise of circuit A . . . . . . . . . . . . . . . . . . . . . . . . 62
## Notation

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC</td>
<td>Analog-to-Digital Converter</td>
</tr>
<tr>
<td>DAC</td>
<td>Digital to Analog Converter</td>
</tr>
<tr>
<td>SAR</td>
<td>Successive Approximation Register</td>
</tr>
<tr>
<td>CMM</td>
<td>Common Mode</td>
</tr>
<tr>
<td>DNL</td>
<td>Differential Non-Linearity</td>
</tr>
<tr>
<td>INL</td>
<td>Integral Non-Linearity</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>FS</td>
<td>Full Scale</td>
</tr>
<tr>
<td>OS</td>
<td>Offset</td>
</tr>
<tr>
<td>AVG</td>
<td>Average</td>
</tr>
<tr>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>REF</td>
<td>Reference</td>
</tr>
<tr>
<td>POS</td>
<td>Positive</td>
</tr>
<tr>
<td>NEG</td>
<td>Negative</td>
</tr>
</tbody>
</table>
1.1 Introduction

In this thesis, methods for characterizing an ADC are developed and tested on a manufactured device that contains an ADC that is fed by an adjacent circuit, here referred to as circuit A. A very rough overview of this, excluding a lot of other parts, can be seen in fig. 1.1.

Figure 1.1: Rough overview of the measurement device.

To give the most accurate digital representation of the incoming analog voltage, the device has registers for parameter settings within the ADC and circuitry A, which can be used to optimize its performance.
1.1.1 Motivation

By performing measurements on the full signal chain, the results from the characterization phase can be used to investigate how the non idealities of the ADC affects the total noise of in device.

When there is enough data from the characterization of the ADC, the results could be implemented in a high-level model for increased accuracy which can be of aid when developing a device that would actually be used in the future.

1.1.2 Problem statements

Having a set of parameters in the ADC which are of interest, the following characteristics will be estimated for some of combination of parameter settings:

- Common mode input range.
- Gain.
- Input referred shift and offset.
- Input referred noise and quantization noise.
- Differential and Integral non-linearity, DNL/INL.

This thesis work will, from the results of the estimation of the characteristics mentioned above together with simulation results try to answer the following questions:

- How does the ADC parameter settings affect the performance of the ADC?
- What impact does the ADC have on the noise of the whole system?

1.2 Background

To determine whether an ADC satisfies its expected requirements when compiling technical documentation for the device, the manufacturer need to test the design sufficiently to be confident on guaranteeing the functionality when delivering to the customer. Having a large set of units produced, this task can quickly become cumbersome if the testing procedure is to complex [15]. The testing procedure also often requires that the full transfer response between the analog input voltage and the generated digital code should be used, which becomes more difficult to extract with ADCs of higher resolution [18], since it is considered bad practise to estimate non idealities of an ADC by only using a part of the input range [14]. Due to this reason, a research field in ADC and DAC testing has emerged where the goal has been to minimize the time consumed by the testing phase while still maintaining accurate results, which will, for example lead to reduced time to market for manufacturers.
The efficiency of two test methods are compared and presented in [8] for estimating non-linearities in ADCs. One approach is an open-loop configuration using some type of digitally controlled voltage source for generating a voltage ramp, spanning the full input range of the ADC while recording the output code together with the input voltage and thus being able to map the relation between input voltage and output code. The other solution presented is a closed-loop configuration where instead the output code from the ADC is compared to fixed code and where the result of this comparison is determining whether the input voltage to the ADC should be increased or decreased in order to drive the difference to zero and read the input voltage after that it has settled [8].

IEEE has proposed a standard on methodology and terminology to be used both by the manufacturers as well as the user to be of assistance when testing and characterizing an ADC [7], similarly to the IEEE Standard for Digitizing Waveform Recorders [6]. The tests proposed in [7] is designed to identify the non idealities of the ADCs and can for example aid an organization that is looking for suitable device to meet some requirement specification in a new project or for evaluating an existing design that has been returned from tape-out to see how well the physical device correspond with the simulations.
1.3 Analog-to-Digital Converter

The ADC used for converting the incoming analog voltage is an Successive Approximation, SAR, ADC, with capacitor array and switches connecting the input voltage to the comparator. The conversion procedure is based on binary search and charge redistribution [16]. The ADC is of a differential type, meaning that it has two inputs that are compared to each other at the time of the conversion. Fig.1.2 shows a simplified circuit diagram of a SAR ADC that is slightly different to the ADC used within the device, but serves well as an example to explain the basic operation of a SAR ADC in general.

![Simplified circuit diagram of a SAR ADC](image)

**Figure 1.2: Example of a differential SAR ADC.**

**Transfer curve** An ADC with the resolution of N bits can represent the values between 0 and $2^N - 1$ where the values are spread out over the input range given in the difference of the two input voltages sampled on the negative and positive capacitor arrays. Fig.1.3 shows the transfer characteristics of a differential ADC of the full input range as well as a zoomed in part.
Having a voltage difference above the high threshold will only produce the maximum output code $1...11$, while having a voltage difference below the lower threshold will only produce the minimum output code of $0...00$. These two thresholds are centered around a voltage difference of 0 volts, which will generate the so-called mid code equal to $\frac{2^N - 1}{2}$.

**Operation of the ADC**  At the beginning of each conversion, all of the switches in the capacitor array is connected to $V_{\text{in,pos}}$ and $V_{\text{in,neg}}$, which is the output voltage of circuit A, as can be seen in fig.1.2, while the switch closest to the comparator is connected to the reference voltage. The switches of the capacitors are then disconnected from the input voltages which is followed by also disconnecting $V_{\text{sampref}}$. These sampling times, i.e. the time instances where the switches to the capacitor array disconnects from the input voltage may be spread out over time which comes with the benefit of averaging out some of the noise on the input voltages. When this has been done, a voltage is trapped over the capacitors equal to

$$V_{c,\text{pos}} = V_{\text{sampref}} - V_{\text{in,pos}}$$
$$V_{c,\text{neg}} = V_{\text{sampref}} - V_{\text{in,neg}}.$$  \hspace{1cm} (1.1)

The capacitors are then connected to $V_{\text{ref}}$, giving the inputs of the comparator the voltages equal to

$$V_{c,\text{pos}} = V_{\text{sampref}} - V_{\text{in,pos}} + V_{\text{ref}}$$
$$V_{c,\text{neg}} = V_{\text{sampref}} - V_{\text{in,neg}} + V_{\text{ref}}.$$  \hspace{1cm} (1.2)

When the capacitor array has settled with the voltages derived in eq.1.2, the comparator evaluates its positive input, $V_{\text{comp,p}}$, against its negative input $V_{\text{comp,n}}$ and gives the output

$$V_{\text{comparator}} = \begin{cases} 
1, & \text{if } V_{\text{comp,p}} > V_{\text{comp,n}} \\
0, & \text{else}.
\end{cases}  \hspace{1cm} (1.3)$$
The SAR ADC will now perform its search algorithm in order to make the two inputs as close to eachother as possible. If the output of the comparator is equal to 1 for the first comparison, the bottom plate of the largest capacitor group in the array on the positive side is connected to ground while the bottom plate of the largest capacitor group on the negative side is connected to $V_{\text{high}}$, or the other way around if the result from the comparator is equal to 0, as can be seen in fig.1.4.

![Capacitor switches set depending on the comparator decision.](image)

**Figure 1.4:** Capacitor switches set depending on the comparator decision.

Given the situation that the first comparison is equal to 1 and the capacitor groups in the charge distribution array are binary weighted, the two input nodes to the comparator will now have the voltage potential equal to

$$V_{\text{comp},p} = V_{\text{sampl}} - V_{\text{in, pos}} + \frac{V_{\text{ref}}}{2}$$

$$V_{\text{comp},n} = V_{\text{sampl}} - V_{\text{in, neg}} + \frac{V_{\text{ref}}}{2} + \frac{V_{\text{high}}}{2}.$$  

(1.4)
After the capacitors has settled to this new value, another comparison will be performed which result will be the basis of how the switches for the second largest capacitor group should be set, which, if the voltage at the negative input are now larger than the positive input of the comparator due to the way the switches were set in eq. 1.4, will be set in the following way

\[
V_{\text{comp}, p} = V_{\text{sampref}} - V_{\text{in}, \text{pos}} + \frac{V_{\text{ref}}}{4} + \frac{V_{\text{high}}}{4}
\]

\[
V_{\text{comp}, n} = V_{\text{sampref}} - V_{\text{in}, \text{neg}} + \frac{V_{\text{ref}}}{4} + \frac{V_{\text{high}}}{2}.
\]

This procedure will continue until all of the capacitor groups has been tested and all of the switches at each of the capacitor groups has been set to either ground (0) or \(V_{\text{ref high}}\) (1), which gives the digital representation of the incoming analog voltage. Due to the operation of the differential ADC, the bits on the opposite inputs are inverted with respect to each other, presenting the possibility to also extract the inverted code if desired.

**Offset compensation** Even if the two input nodes of the comparator are connected to the same potential, there will be an offset present that will affect the accuracy of the conversion. This offset can be dealt with by taking two samples of the same voltage with different polarities as

\[
\text{Code}_1 = V_{\text{in}+} - V_{\text{in}-} + V_{\text{offset}}
\]

\[
\text{Code}_2 = V_{\text{in}-} - V_{\text{in}+} + V_{\text{offset}}.
\]

By then taking the difference between the two codes and dividing the result by two, the final output code with the offset subtracted can be calculated as

\[
\text{Code} = \frac{\text{Code}_1 - \text{Code}_2}{2} = \frac{(V_{\text{in}+} - V_{\text{in}-} + V_{\text{offset}}) - (V_{\text{in}-} - V_{\text{in}+} + V_{\text{offset}})}{2} = V_{\text{in}+} - V_{\text{in}-}.
\]

### 1.3.1 Registers in the ADC

As mentioned in sec.1.1, the ADC has a set of registers which can be used to optimize the performance. The actual names or resolution of these settings will not be revealed, but the basic principles will be explained in this subsection.

### 1.3.2 Offset mode

The two switches connecting \(V_{\text{sampref}}\) to the two inputs of the comparator at that the time of the conversion, are both connected to a voltage potential of some value. What differs between the two inputs of the comparator is that the positive and input has an additional voltage potential that can be added to \(V_{\text{sampref}}\), which
introduces the possibility of moving the sample reference point for the positive input with respect to the negative input of the comparator, as illustrated in fig1.5.

![Figure 1.5: Principle of operation for offset mode.](image)

Adding this additional voltage to the sample reference of the positive input of the comparator will increase the difference between the positive and negative inputs, which means that in order for the SAR to even out the voltage difference on the inputs, the switches to the capacitor groups on the positive side will more likely be set to ground since this voltage is initially higher than the negative input. Equation 1.5 can now be rewritten as

\[
V_{\text{comp},p} = v_{\text{SampRef}} + V_{\text{shift}} - V_{\text{in},p} + V_{\text{ref},p} \\
V_{\text{comp},n} = v_{\text{SampRef}} - V_{\text{in},n} + V_{\text{ref},n}.
\]  

What this will mean in practice is that each of the converted digital values will be lower for a higher value of Offset mode. Fig.1.6 shows the transfer curves with an increase of the value for \( V_{\text{offset}} \).

![Figure 1.6: Transfer curves with different values of \( V_{\text{offset}} \).](image)

### 1.3.3 ADC gain

The value of \( V_{\text{high}} \), used by the successive approximation of the ADC for adjusting the voltage potential of the inputs in order to make them equal, can be altered to change the transfer characteristics. This voltage can be seen as the correction constant, and by setting it to a low value, the SAR will have a more difficult time equalizing the two inputs which means that the range of possible input voltages
are reduced. fig.1.7 shows the transfer characteristics for two different values of $V_{\text{high}}$.

![Graph showing transfer curves with different values of $V_{\text{high}}$.]

*Figure 1.7: Transfer curves with different values of $V_{\text{high}}$.*

Setting this voltage a high value, the SAR will have a better chance of equalizing two input voltages that are further apart. This means that having a high value of $V_{\text{high}}$ gives a larger input range and vice versa.

### 1.3.4 Common mode adjust

The common mode voltage of the two inputs can be calculated as [17]

$$V_{\text{cmm}} = \frac{V_{\text{in},p} + V_{\text{in},n}}{2}. \quad (1.9)$$

Looking at eq.1.2, the common mode level of the voltages $V_{\text{in},\text{pos}}$ and $V_{\text{in},\text{neg}}$ are subtracted from the inputs to the comparator, resulting in a reduced common mode level. In order to compensate for this, the common mode voltage level of the inputs can be adjusted, as illustrated in fig.1.8.

![Diagram showing the principle of operation of common mode adjust.]

*Figure 1.8: Principle of operation of common mode adjust.*

### 1.3.5 Settings inside the comparator

The comparator used in the ADC consists of a pre-amplifier together with a rail to rail latch circuit, which has the flexibility of programmable current sources...
and start timing, with an adjustable filter in between. The pre-amplifier comes with the benefit of protecting the capacitor array from the kick-back generated by the latch circuit when producing its rail to rail decision, as well as producing some low noise amplification of the input signals.

![Pre-amp Filter Latch](image)

*Figure 1.9: Rough block diagram of the comparator.*

The timing sequence should behave in such way that the capacitor arrays have had time to settle before the pre-amplifier starts to produce an output and that this output has been gained enough from the pre-amplifier for the latch circuit to make a good enough decision on. The registers for the above mentioned settings can be used to achieve faster operation of the comparator, meaning that the signals on inputs of the latch circuit takes less time to reach its final value as well as the time needed for the latch to make its decision.
1.4 Characteristics of the ADC

The metrics of an ADC are grouped in static and dynamic, where static metrics describes the characteristics when the input voltage to the ADC is of low frequency and the dynamic metrics for when the input voltage is of high frequency [13] [7]. This thesis will focus on characterization of the static metrics.

1.4.1 Common mode input range

The common mode input voltage can then be calculated with eq.1.9. When keeping the voltage difference, $\Delta V$, between the two input terminals of the ADC constant, the output code should ideally remain the same. As seen in fig.1.10 (b), the actual transfer curve should lie as close to the ideal flat curve as possible since the $\Delta V$ is not changing. Although, due to the increase of the common mode voltage level of the input signals, the output code will start deviating from the ideal code. Subtracting the value of the ideal code from the actual generated code will give the offset in terms of codes, as will be explained further down in sec.1.4.3.

Since every measurement taken at each input level is in the form of a N times 1 matrix, all of the N conversions should ideally be the same. However, increasing the common mode level will affect the ADCs and introduce the possibility of incorrect conversions. The N times 1 matrix can be used to calculate the standard deviation of the N conversions for that specific input, which is illustrated in Fig.1.10.

Figure 1.10: Two ways of determining the common mode input range.

The common mode input range can therefore be defined as the range of common mode voltage levels where the ADC produces acceptable output codes. Il-
illustrated in fig.1.11 are two ways of determining the common mode input range.

![Image of Standard Deviation and Lost Codes](image1.png)

(a) ADC Noise.  
(b) Lost codes of the ADC.

Figure 1.11: Thresholds of the common mode input range.

The two criterias shown are where the standard deviation of N conversions is kept within a known boundary whereas the other boundary is how many output codes that can be lost due to the offset in the signal, which will give a lower and an upper bound on what the common mode input should lay between.

### 1.4.2 Gain

fig.1.12 shows the transfer curves of an number of different ADCs, all having a unique input range.

![Image of Digital Output Code](image2.png)

Figure 1.12: Different input ranges for a set of different ADCs.

As can be seen in the figure, the ADCs produce the same number of digital codes with the expection that they are spread out over different input ranges, while keeping the location of the mid code centered around \( V_{\text{in,\text{diff}}} = 0 \). The gain can therefore be expressed as the range between the all zero code and the maximum code,
1.4 Characteristics of the ADC

or the width of an output code, defined as

$$\frac{V_{FS}}{2^N - 1},$$

(1.10)

where \( N \) is the resolution of the ADC in number of bits and \( V_{FS} \) is the input range of the ADC. It might be the case that the code width is not constant throughout the whole input range if the ADC suffers from non-linearity errors, which will be explained further in this section.

1.4.3 Input referred shift and offset

In a single ended ADC, the offset can be seen as the difference between the actual and the ideal input voltage where the output code of the ADC transitions from \( V_{0..00} \) to \( V_{0..01} \). Using eq 1.10, the offset can be given in bits as [11]

$$E_{offset} = \frac{V_{0..01,measured} - V_{0..01,ideal}}{\frac{V_{FS}}{2^N - 1}}.$$  

(1.11)

Since the ADC is of a differential type, the ideal location of the mid code, which should be located where the difference between the two input signals is equal to zero, i.e., in the middle of \(-\frac{V_{FS}}{2}\) and \(\frac{V_{FS}}{2}\), should be used as reference to the measured location of the midcode when calculating the offset. Eq.1.11 could therefore be re-written as

$$E_{offset} = \frac{V_{midcode,measured} - V_{midcode,ideal}}{\frac{V_{FS}}{2^N - 1}}.$$  

(1.12)

Illustrated in fig.1.13 is some transfer curves of one of the ADCs for a few different gain values with an ideal transfer curve as a reference.

![Figure 1.13: Offset error.](image-url)
The offset can be due to the input referred shift of the ADC, which is seen as the offset in volts between the ideal mid code and the actual mid code. An increase of the gain can therefore create a larger offset in terms of codes for the same value of the input referred shift.

### 1.4.4 Input referred noise

There exists a probability that the ADC will produce an output code than does not correspond to the input voltage due to the fact that the ADC introduces some noise during the conversion. If the input voltage $x$ is also held $N$ times at each voltage level, the standard deviation, $\sigma$, of the output code $y$ can be with Eq.1.13[10]

$$
\sigma_y = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \bar{y})^2},
$$

where $\bar{y}$ is the mean value

$$
\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i,
$$

of the N conversions for each input step corresponds. This noise will be a lot smaller when the input level are located at the middle of an output code and larger when approaching a transition between two output codes, as can be seen in fig.1.14. Given this fact, the standard deviation will be averaged for all input voltage levels, giving an average representation of the noise over the full input range.

![Jittering code transitions due to the presence of noise.](image)

Figure 1.14: Jittering code transitions due to the presence of noise.

**Quantization noise** In an ideal ADC, the only error at the output is due to the quantization of the input values due to the fact that the ADC has a finite resolution of N bits [12]. The quantization error can be described as

$$
e(t) = t, \quad \frac{-q}{2} < t < \frac{q}{2},
$$

(1.15)
where q is the width of the code. The root mean square quantization error for that digital code can then be described as [12]

$$\sqrt{e^2(t)} = \sqrt{\frac{1}{q} \int_{-q/2}^{q/2} t^2 dt} = \sqrt{\left[ \frac{t^3}{3q} \right]_{-q/2}^{q/2}} = \sqrt{\frac{q^2}{24} - \left( \frac{q^2}{24} \right)} = \frac{q}{\sqrt{12}}.$$ (1.16)

Given the fact that the quantization noise of the whole input range should be calculated, eq.1.16 should be re-written as

$$\sqrt{e^2(t)} = \sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \frac{q^2}{12}}.$$ (1.17)

**Differential and Integral Nonlinearity**  The Differential Nonlinearity, DNL, is the deviation of the width of the output code from the ideal width of the digital output code [9]. Figure 1.15(a) illustrates the transfer characteristics of a differential ADC with non-linearities. The DNL for each output code can be found by subtracting and dividing with the width of an ideal output code as

$$DNL_n = \frac{q_{\text{measured}} - q_{\text{ideal}}}{q_{\text{ideal}}},$$ (1.18)

which gives the DNL error in terms of bits [12]. The transfer function of an ideal ADC is a straight line. The Integral Nonlinearity is the difference between a the straight line of an ideal ADC and the actual transfer function of the ADC [9] as can be seen in fig.1.15(b). The INL can be found by simply integrating the DNL [7] [12] as

$$INL_n = \sum_{i=0}^{N} DNL_i.$$ (1.19)
2.1 The test setup

A rough block diagram of the setup can be seen in fig.2.1. The PC is running a LabView [3] program for communicating with the PXIe [5][4] which is used for generating the two test voltages for the inputs as well as acquiring the results from the ADC.

![Diagram of test setup](image)

**Figure 2.1: Test setup.**

The setup also includes a Keithley 2701 [2], a multimeter that is the used to feed back the two test signals generated from the voltage source to the PC so that information of the actual generated voltages can be used for more precise results.
Including the multimeter  When using the setup seen in fig.2.1, the ethernet connection between the multimeter and the PC goes through a hub to still allow the PC to be connected to the rest of the office network, as seen in fig.2.2.

![Diagram showing ethernet connection through a hub](image)

*Figure 2.2: Connecting the Keithley 2701 via ethernet [2].*

The multimeter needs to be initialized in the LabView program to be able to record its measurements. The code in fig.2.3 below should be executed once at the beginning of each measurement phase.

![LabView code for connecting the multimeter](image)

*Figure 2.3: Using the Keithley 2701 multimeter with LabView.*

As seen in the figure above, the IP address of the multimeter needs to be specified. This address can be found by pressing SHIFT followed by ETHERNET on the front panel of the Keithley 2701.
Figure 2.4: Reading the IP address of the Keithley 2701 [2].

The IP address of the Keithley 2701 can be changed from time to time, and if there is any uncertainty on whether or not the PC can establish a connection with the multimeter, one can try to “ping” the Keithley 2701 by opening the command window and typing ping followed by the IP address acquired from the previous step and hitting the enter key.

Analyzing the voltage source  
Available inside the PXIe system are two modules capable of generating voltages. Both of them were tested and evaluated by setting both of the outputs to the same voltage level in the software and then sampling the difference of the actual generated voltages. fig.2.5 shows the results where it can be seen that the PXIe 6361 are preferred over the PXIe 6556.

Figure 2.5: Measuring the accuracy of the two voltage sources.

In order to get a reliable measurement of the ADC, i.e., keeping the a stable output for a fixed input, the voltage levels of the PXIe system should be stable enough so that the magnitude of the ripple does not exceed the resolution of the
ADC. After performing the test described above, it was shown that the standard deviation of the voltages generated by the PXIe system could be neglected. The two inputs of the ADC is connected to pin 21 and 22 of the termination of the PXIe 6361, which is two analog voltage outputs, seen in fig.2.6.

![Pin out of the PXIe 6361](image)

\textbf{Figure 2.6: Pin out of the PXIe 6361 [4].}

\subsection{The LabView Software}

The LabView program running on the PC has the functionality of setting the input signals $V_1$ and $V_2$, writing to registers in the ADC, extracting measurement data and reading the measurements of the Keithley 2701 multimeter. fig.2.7 show a flowchart of the LabView program.
2.1 The test setup

![Flowchart of the Labview Program]

**Figure 2.7:** Flowchart of the Labview program.
**Test Input Signals**  The two test signals are defined as

\[
V_1 = V_{CM} + V_{1os} + K_1 V_{in} \\
V_2 = V_{CM} + V_{2os} + K_2 V_{in},
\]

where \( V_{CM} \) is the offset voltage of the two signals, \( V_{1os} \) and \( V_{2os} \) is the individual offset voltage of the two signals and \( K_1 \) and \( K_2 \) is the gain constant of how much of the voltage \( V_{in} \), which is a value that will be swept from 0 to \( V_{stop} \) in steps of \( V_{stepsize} \), will be added to the resulting signal. Fig.2.8 shows an example of the two kinds of input test signals for estimating the characteristics mentioned in sec.1.4.

![Figure 2.8: Example of the test signal behaviour.](image)

The signals shown in fig.2.8 (a) above is used for estimating the common mode input range while the signals in fig.2.8 (b) is used for estimating the gain, offset, shift, DNL, INL, inputreferred and quantization noise. The later mentioned signals should be swept so that the whole input range of the ADC are tested. This means that the difference of the two signals should span from slightly below \(-V_{FS}/2\) to slightly above \(V_{FS}/2\). Due to the fact that the ADC might be configured with a value of \( V_{shift} \) present, the signals shown in fig.2.8 (b) should continue until the maximum code of an ADC with the maximum value of \( V_{shift} \) has been generated.

**Register writes**  The bits in each register are set by writing the register address followed by the values for each byte in hexadecimal form as 0xRR 0xBN 0xBN-1 ... 0xB0, where 0xRR is the register address and 0xBX is the value of the byte X. By putting the register writes in a textfile, one line for each register, a list of register settings can be formed. Each group of register settings are separated by a comma at the end of the line of the last register write, as seen in fig.2.9.
2.1 The test setup

Figure 2.9: Example of a list of register writes used for sweeping register values.

Since the whole register needs to be written to, even if only one byte should be set, the register should be written with consideration to the other bytes in that register.

Data returned For each input voltage setting, a set of register settings, which is stored in a text file, will be written to the ADC. For each input voltage and register setting, the program will save N measurements. The measurements are saved in the form of a matrix with N rows and 1 column where each value corresponds to a digital conversion of the input voltage. Each of the result vectors are stored individually in a text file where the file name includes information on which of the settings from the list were written at that time, what the two ideal input voltages were together with the measured difference between the actual generated voltages. An example of the file name of a measurement is

\[ \text{ListSetting}_0 \_ V1 \_ 0, 120000 \_ V2 \_ 0, 520000 \_ \text{diff} \_ - 0, 400000 \_ \text{Image} \_ 1 \_ .txt \] (2.2)

Figure 2.10 shows an illustration of the process of collecting data from the ADC for one of the test cases where all of the output codes should be generated by sweeping the two input voltages, covering the full range of the ADC.

Figure 2.10: Extracting the measurement data for each input voltage.
As seen in the figure above, each input voltage level are kept for a period of time, allowing the ADC to perform N conversions. The size of the incremental input voltage steps are directly related to how well the transfer characteristics of the ADC could be estimated. The approach were to start from two initial voltages and increase the levels in the smallest increments possible for the voltage source. This was proven a good method and yielded a high accuracy.

The measurement results are analyzed with the help of scripts written in GNU Octave, which is a programming language for scientific computing [1], that are explained more in detail in the following sections.

2.2 Register combinations

The registers described in sec.1.3.1 are to be investigated both individually and in different combinations. The amount of combinations scale quickly with an increase in number of registers. With N registers of size RegSize, the number of combinations can be calculated as

\[
\text{Combinations} = \text{RegSize}_1 \cdot \text{RegSize}_2 \cdot ... \cdot \text{RegSize}_N.
\]

Due to the fact that the collection of the data takes time some time, it would not make sense to evaluate all of the combinations. Therefore, the approach was instead to group some of the settings together in combinations. The different combinations of register settings were discussed with my supervisor together with some of the designers, which had input on different performance parameters that could be tested.

**Individual register sweep** Before trying out any of the combinations of the register values, each of the register in should be swept individually while keeping the others fixed in order to see how each of them affect the performance.

**Table 2.1:** Register settings for individual sweeps.

<table>
<thead>
<tr>
<th>Register</th>
<th>Start Value</th>
<th>Step Size</th>
<th>End Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Offset mode</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>ADCGain</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Latch Current</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Pre-amp current</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Filter coefficient A</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Filter coefficient B</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Comparator start</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Common mode adjust</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
</tbody>
</table>

The other registers were set to either a reset value or a fixed preferred value, given the register which were to be tested.
2.2 Register combinations

Register combinations two  These register combinations in table 2.2 should be swept to test how the timing inside the comparator together with some filter characteristics affects the ADC.

Table 2.2: Register settings for combination two sweep.

<table>
<thead>
<tr>
<th>Register</th>
<th>Start Value</th>
<th>Step Size</th>
<th>End Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latch Current</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Pre-amp current</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Filter coefficient A</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Common Mode Adjust</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Offset mode</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Filter coefficient B</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Comparator Start</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Adc Gain</td>
<td>0</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

Register combination three  The settings in table 2.3 should be combined together with table 2.4 to test how the values of the current sources together in combination with another filter coefficient affect the performance of the ADC.

Table 2.3: Register settings for combination three sweep.

<table>
<thead>
<tr>
<th>Register</th>
<th>Start Value</th>
<th>Step Size</th>
<th>End Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparator Start</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Common Mode Adjust</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Filter coefficient B</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Offset mode</td>
<td>Fixed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Latch Current</td>
<td>0</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Adc Gain</td>
<td>0</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

Table 2.4: Combinations of pre amp current and filter coefficient A.

<table>
<thead>
<tr>
<th>Pre-amp current</th>
<th>Filter coefficient A</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
2.3 Estimating the characteristics of the ADC

When having performed a test sequence by applying the input signals as seen in figure 2.10, the returned text files containing the conversion data can be found in the result folder specified in the LabView program. Since this folder will include the results for all of the different register combinations, an octave script was written in order to create sub-folders and categorizing the measurement data. While the program moves the files, it also converts the .txt files into .bin files as illustrated in fig.2.11.

![Folder organization](image)

(a) Unsorted .txt.  (b) Folders.  (c) Sorted .bin.

**Figure 2.11: Organizing the data.**

The reason for converting to .bin is that files in this format is handled a lot quicker by Octave. To import the data into Octave, the script changes directory to the path where all of the folders seen in fig.2.11 (b) are located and returns the number of folders i.e., the number of setting combinations. The program then enters the first folder, saves all of the file names in that folder, which is all of the .bin files containing the measurement results in a cell array, together with key numbers that are extracted from the file name on one row, as seen in fig.2.12. This introduces the possibility of sorting the the files by one of these key numbers.

```
full filename, setting, v1, v2, measured voltage difference
full filename, setting, v1, v2, measured voltage difference
... 
... 
full filename, setting, v1, v2, measured voltage difference
```

**Figure 2.12: Cell array containing the file names together with the current setting combination, ideal input voltages and measured voltage difference.**

When all of the file names has been read and sorted, the next function loops through all of the rows in the cell array, reads every file and stores the data by concatenating a matrix of values. In a separate result matrix, the read file with the dimension of N rows times 1 column is averaged, giving a matrix with the average value of the N conversions for that input voltage, as illustrated in fig.2.13.
2.3 Estimating the characteristics of the ADC

In addition to this, the voltage difference at the time of the conversion given from reading the result of the Keithley 2701 multimeter is stored in a third result matrix. Since this vector and the vector of the average conversion values is of the same length, the voltage difference of the input signals can be mapped to the mean value of the conversion from the ADC, as can be seen in fig.2.14.

The full transfer curve as shown in fig.1.3, can also be estimated by finding the
edges where the output of the ADC transitions between adjacent codes. This is
done by interpolating the mean values of the conversions for each input voltage,
i.e. the values shown in fig.2.14, finding which voltage difference of the input
signals generated the output code of 0.5, 1.5, ..., $2^N - 1.5$, illustrated as the red
circles in fig.2.15.

![Figure 2.15: Interpolating the data to find the transitions.](image)

This is done by using the function interp1 in octave in the following way

\[
transitions = \text{interp1(meanConvs, measuredVoltDiff, 0.5 : 1 : } 2^N - 1.5));
\]  

(2.4)

which returns a matrix with values of the measured voltage differences that would
correspond to the values between 0.5 and $2^N - 1.5$ in steps of 1, estimated in the
vector of the mean values of the ADC conversions.

Interpolating the data as described above comes with the advantage of increased
accuracy together with information that can be used to better estimate the code
widths of the ADC, which is needed to estimate some of the characteristics, as
will be explained later in the following subsections.

### 2.3.1 Common mode input range

The common mode input range can be defined for the ADC to fulfill one or two
of the criteria mentioned in sec.1.4.1. This range can be found by ramping up
both $V_1$ and $V_2$ equally much, keeping a constant $\Delta V$, showed in fig.2.8 (a), and
examining the output of the ADC.

By setting a limit on how many lost codes are acceptable, the common mode input
range can be found by subtracting the value of the midcode from the generated
code, which will give the number of lost codes due to the offset that the ADC
failed to remove. Figure 2.16 shows the two thresholds $\text{Cmm}_{\text{upper}}$ and $\text{Cmm}_{\text{lower}}$
that are estimated by analyzing the conversion data.
2.3 Estimating the characteristics of the ADC

2.3.1 Lost codes

Lost codes

Common mode voltage

Figure 2.16: Finding the common mode range in terms of lost codes.

The script has a reference value that defines the acceptable thresholds, which the number of lost codes should lie within, seen as the red lines in the figure above. Having subtracted the value of the ideal midcode from the conversion data, the resulting matrix is tested against the two thresholds to see if the value lies within the acceptable bound. As seen in fig.2.16, the common mode level for which the value exceeds either the upper or lower limit, that common mode level is said to be the lower or upper bound, even if it lies within the bound for higher common mode levels.

The standard deviation of the ADC can be found by using eq.1.13 on the returned measurement data matrix, which corresponds to N conversions for every input voltage of the ADC. The matrix containing the standard deviation for each common mode voltage level is then, just as for the case for lost codes, tested against a reference value. Since the standard deviation can not be less than zero, the reference value is only used for defining the upper bound, seen as the red line in fig.1.11 (a).

2.3.2 Gain estimation

By interpolating the conversion data and finding which voltage difference corresponds to the first and last code transition, i.e., the for which the ADC outputs the value of 0.5 and $2^N - 1.5$, the input range of the ADC, excluding the minimum and maximum code, can be estimated. Knowing the resolution of the ADC, this range can be divided by $2^N - 2$, i.e. the resolution of the ADC in bits minus one bit to calculate the mean width of a code in terms of input voltage.

The gain of the ADC can also be expressed in terms of the full input range which can be found by taking the range between the first and last transition and adding the mean code width, thus compensating for excluding it in the first calculation.

In the result section, the estimated gain is expressed in terms of the input range. The mean width of a code are however used when calculating some of the other characteristics which will be explained in the following sections.
2.3.3 **Input referred shift and offset estimation**

By interpolating the points from the conversion results with respect to the value of the mid code as

\[
shift = \text{interp1}(\text{meanConvs}, \text{measuredVoltDiff}, \frac{2^N - 1}{2}); \quad (2.5)
\]

the location of the mid code in terms of the voltage difference of the two input signals can be found, which is equal to the input referred shift marked at the X axis in fig.1.13.

Estimating the offset in terms of number of codes can be done by instead interpolating the conversion results, finding out which output code is corresponding to the voltage differences of zero volts as

\[
offset = \text{interp1}(\text{measuredVoltDiff}, \text{meanConvs}, 0) - \frac{2^N - 1}{2}; \quad (2.6)
\]

marked as the offset on the Y axis in fig.1.13.

2.3.4 **Noise estimation**

**Input referred noise** The input referred noise of the ADC is estimated by taking the standard deviation of the returned measurement matrix from the ADC, i.e. each of the matrixes to the left in fig.2.13, corresponding to the N conversions for every input voltage, according to the eq.1.13. Since the standard deviation may differ due to the proximity of the input voltage to the transition edges, as explained in sec.1.4.4, this calculation is performed for every input voltage and the result is then averaged to give an average input referred noise factor for the whole input range of the ADC.

Since changing the value on some of the parameters, especially the ADC Gain, may cause a change in input range, thus changing the mean value of a code width, the standard deviation in terms of codes can instead be expressed relative to the mean code width for that specific setting. This is done by multiplying the value of the standard deviation in codes by the value of the mean code width in terms of voltage that was calculated when estimating the gain of the ADC.

**Quantization noise** As described in sec.1.4.4, the width of each code is used to calculate the total quantization noise. These widths are estimated from the measurement data by taking the matrix of the interpolated transition edges, calculated in eq.2.4, containing the voltage differences for each transition and performing the subtraction

\[
\text{codeWidths} = \text{transitions}(2:end) - \text{transitions}(1:end - 1); \quad (2.7)
\]

By then performing the root-mean-square of these values, the total amount of quantization error for the whole input range is found.
2.3.5 Nonlinearity estimation

Differential nonlinearity  The equation for calculating the DNL needs the ideal code width, as can be seen in eq.1.18. There are different approaches to arrive at this number, where one is to simply take the mean code width as was described in 2.3.2. This might be somewhat misleading, since using the mean width assumes that the differences between the code widths are spread evenly over the input curve. By analyzing the transfer curve, the range where the ADC were the most linear could be found and used to calculate the mean code width that would represent the ideal code width for that specific setting and ADC. Using this value as the ideal code width, eq.1.18 can be applied on all of the $2^N - 2$ code widths, which can be found by using eq.2.7, resulting in the DNL for all of the codes.

Integral nonlinearity  The INL for each output code can be found by performing numerical integration of the DNL, as shown in eq.1.19.

2.4 Input referred noise estimation of circuit A

In this thesis, the input referred noise are grouped into two noise sources, coming from both circuitry A or the ADC, as illustrated fig.2.17 below

![Input referred noise of the signal chain.](image)

Figure 2.17: Input referred noise of the signal chain.

In order to estimate the input referred noise of circuitry A, the noise of only the ADCs should be estimated, as explained in the sec.2.3.4 for a set of different common mode levels of the two input signals within a reasonable bound. These input referred noise levels of the ADC for the different common mode voltages should then be averaged.

To estimate the noise of the circuitry A, the input signal should be kept constant while recording a large set of outputs from the ADC. Calculating the standard deviation of this sample set will give the input referred noise of the whole signal chain. Assuming that the standard deviation is mainly due to thermal noise in circuit A and the ADC, the total noise of the measurement device can be expressed as

$$ \text{Noise}_{\text{tot}} = \sqrt{\text{Noise}_{\text{circuit A}}^2 + \text{Noise}_{\text{ADC}}^2} \quad (2.8) $$

where $\text{Noise}_{\text{ADC}}$ is the input referred noise of the ADC averaged for the different common mode levels. Rewriting eq.2.8, the noise from the circuit A can be calculated as

$$ \text{Noise}_{\text{circuit A}} = \sqrt{\text{Noise}_{\text{tot}}^2 - \text{Noise}_{\text{ADC}}^2} \quad (2.9) $$
This noise performance of circuit A should be evaluated for a set of modes in combination with a few values of the setting offset mode.
In this section, the results from the characteristics estimation will be summarized and briefly commented. Further discussion on the results from this thesis and comments on the method in general will follow in chapter 4.

As mentioned in section 2.2, all of the register settings were to be tested both separately and in combinations. The results presented here will not have any values on the Y-axis, and only show a subset of the individual register sweeps will be showed.

### 3.0.1 Common mode range

As seen in the figures 3.1, 3.2 and 3.3, the input referred noise and the lost codes due to the increase in common mode voltage of the input signals are somewhat stable and constant up until a point where they drastically increase. This "breaking point" occur at higher or lower common mode voltage levels depending on the register settings, where a change in some registers have more impact than others. The testing of the common mode range were done by sweeping all of the register values individually while keeping the others fixed. The results will be presented in plots together with tables.

Changing the value of the latch current and comparator start did little to no effect on the common mode input range in either the noise or the number of lost codes as can be seen in fig.3.1.
Figure 3.1: Sweeping values of Latch current and comparator start.
Changing the values of the filter coefficients between the pre-amplifier and the comparator did not contribute to a notable decrease in the common mode range either as can be seen in fig.3.2 below.

![Graphs](image)

**Figure 3.2**: Sweeping two of the filter coefficients.

The settings that has the most impact on the common mode range is the Pre-amp current, common mode adjust and ADC Gain. As seen in fig.3.3 (a), increasing current for the Pre-amp will result in a drastic increase in the input referred noise as well as the number of lost codes for a higher common mode voltage level. An increase in the value of the common mode adjust setting will, as described in section 1.3.1, allow the input signals to have a larger common mode level. Changing the value of the pre-amp current and the common mode adjust moves the point where the standard deviation drastically increases in terms of common mode level of the input signals while keeping it somewhat constantly low up until that point. Changing the ADC gain however will instead keep the behaviour of the curve the same and instead just increase the total amount of noise.
Figure 3.3: Sweeping values of Pre-amp current, common mode adjust and ADC gain.
3.1 Gain

In this section, the gain in terms of the input range of the ADC is presented. The mean width of a code that is also given by this estimation task, is used for estimating some of the other characteristics presented later on in this section.

3.1.1 Combinations two

The input ranges are grouped in two sets as seen in fig.3.4 where each of them corresponds to either a higher or lower value of the ADC gain setting.

There are very small deviations on the input ranges for the different combination settings, irrespective of the bandwidth given from the settings. The results can be seen as spread somewhat randomly since the different input ranges are almost identical. This holds true for both of the combination groups with a low and high gain factor.
3.1.2 Combinations three

fig.3.5 shows the input ranges of the combinations, which also are grouped in two sets depending on the value of ADC Gain.

For these combination sets, a trend can be seen more clearly. Changing the currents of both the pre-amplifier and the latch circuit cause some deviation of the input ranges for both values of ADC Gain, where one of the coefficients of the filter located in between also cause some minor deviation. These results are useful since they can, with the knowledge of which setting combination corresponds to which point in the figure, describe more clearly how the currents affect the input range.
3.1.3 Individual register sweep

The individual sweeps of the registers can be seen in fig.3.6.

*Figure 3.6: Input ranges of the ADC when sweeping through the settings individually.*
The fact that the setting for ADC Gain cause a change in the input voltage does not come as a surprise. However, sweeping through the other registers of interest individually showed that the settings for altering the current sources of the pre-amplifier and the latch circuit caused a trend of reducing the input range very slightly, while changing the timing of the comparator and the common mode adjust caused very little to no effect.

### 3.1.4 Filter

Figure 3.7 below shows the different input ranges for some of the filter settings, located in between the pre-amplifier and the latch circuit.

![Input range graph](image)

*(a) Input range.*

**Figure 3.7**: Input ranges for different filter settings.

From the figure, a trend of a slightly decreasing input range for some sets of filter coefficients can be seen. Having the values of these coefficients at hand, it can be seen as a pattern where increasing one of them while keeping the others fixed gives this behaviour. Having said that, the deviation of the input range with respect to the filter setup can be seen as negligible.

### 3.2 Offset and shift

The input referred shift given in volts and offset in terms of codes are presented in this subsection. A negative value of the input referred shift implies that the whole transfer curve is shifted to the left, as illustrated in fig.1.13, which could also imply that the offset in terms of codes will be greater than zero.

#### 3.2.1 Combinations two

The whole set of register combinations has a shift of the transfer curve present. This input referred shift are affected by the value of the ADG gain, and the result is also here grouped into two sets as can be seen in fig.3.8 (a). The offset in codes
that is a result of the input referred shift gets amplified for a higher value of ADC Gain, even though the shift changes just very slightly, as can be seen in fig.3.8 (b).

The pattern seen in the above figures can be seen as somewhat random, with the exception of the four first combinations that shows a slight trend in the shift for an increase of the comparator start register for a specific filter coefficient with a high value of the ADC gain.

### 3.2.2 Combinations three

The input referred shift and offset for the combinations three, seen in 3.9, is following a more distinct pattern. The deviation is larger between the setting combinations with a higher value of ADC Gain, which can be seen as the points closest to the bottom in the figure below.
3.2.3 Individual register sweep

fig. 3.10 and 3.11 shows the input referred shift and offset of a subset of individual register sweeps.

**Figure 3.10:** Input referred shift and offset for individual register sweeps.
3.2 Offset and shift

![Graphs showing input referred shift and offset for individual register sweeps.](image)

(a) Input referred Shift, Pre-ampCurrent.  
(b) Offset, Pre-ampCurrent.

(c) Input referred Shift, Comparator Start.  
(d) Offset, Comparator Start.

(e) Input referred Shift, common mode adjust.  
(f) Offset, common mode adjust.

**Figure 3.11:** Input referred shift and offset for individual register sweeps.

It was shown, in addition to the expected increase in shift for the setting ADC Shift, that the settings for ADC Gain and common mode adjust also altered the shift and offset, even if just very slightly, in some “linear” fashion while the contribution of the rest of the settings were negligible.
3.2.4 Filter

The shift and offset for a set of different combinations of filter coefficients are shown in fig.3.12 below, which turned out to be very similar for all of the different filter setups.

![Figure 3.12: Shift and offset for different filter settings.](image)

3.3 Input referred noise

As mentioned in the method chapter, the input referred noise is averaged over the whole input range which was mainly due to the fact that the input voltage at the transition edges between adjacent codes generated results with more noise than the input voltages at the middle of the codes. When analyzing the actual result, it was shown that the input referred noise is also dependent on the distance from the mid code, as can be seen in fig.3.13 below.

![Figure 3.13: Input referred noise distributed over the full input range of the ADC.](image)
The following section presents the input referred noise averaged over the full input range both in terms of number of codes and volts.

### 3.3.1 Combinations two

The input referred noise in terms of number of codes are grouped in two sets of combinations with either a high or low value of ADC gain, as seen in fig.3.14 below.

![Figure 3.14: Input referred noise (codes) of combinations two.](image)

Although the input referred noise shows an increasing trend which can be related to the values of the settings comparator start and the filter, it is still not that much difference between the worst and best case of any of the register combinations.
When looking at the input referred noise in terms of volts, the noise seen in fig.3.14 are multiplied with the mean code width for each of the input ranges, meaning for the high and low value of ADC gain, which results can be seen in fig.3.15.

(a) Input referred noise (volts).

(b) Zoomed, low ADC gain.

(c) Zoomed, high ADC gain.

Figure 3.15: Input referred noise (volts) of combinations two.

When comparing the figures 3.14 and 3.15, the behaviour is the same due to the fact that is has been scaled with the mean code width for that setting. However, it was seen that an increased noise in terms of codes due to a higher ADC gain actually resulted in slightly lower noise in when presenting it in terms of volts, which can be seen by looking at fig.3.18 (c) and (d).
3.3.2 Combinations three

The input referred noise for the combinations three is also showing an increase in the input referred noise, seen in fig.3.16 below.

![Graph showing input referred noise for combinations three.]

Even though the difference between the highest value of input referred noise is very small, the setting combinations with high ADC gain has a slightly larger standard deviation than the setting combinations with low ADC gain.

Figure 3.16: Input referred noise (codes) of combinations three.
The input referred noise given in volts is, however, lower for higher values of ADC gain, just as for the combinations two. These results can be seen in fig.3.17.

(a) Input referred noise.

(b) Zoomed, high ADC gain.  

(c) Zoomed, low ADC gain.

Figure 3.17: Input referred noise (volts) of combinations three.
3.3.3 Individual register sweep

The estimated input referred noise for each of the individual register sweeps can be seen in fig.3.18 and 3.19 below.

(a) Input referred noise (codes), Offset mode.

(b) Input referred noise (volts), Offset mode.

(c) Input referred noise (codes), ADC-Gain.

(d) Input referred noise (volts), ADC-Gain.

(e) Input referred noise (codes), LatchCurrent.

(f) Input referred noise (volts), LatchCurrent.

Figure 3.18: Input referred noise of individual register sweeps.
Just as described in the previous sub sections, the setting for ADC gain reduces the input referred noise relative to its mean code width while it increases when
just looking at the standard deviation in number of codes. This phenomena can be compared with the other registers which more or less exactly show the same behaviour of the standard deviation regardless if the noise is presented in volts or number of codes.

### 3.3.4 Filter

fig.3.20 shows how the input referred noise is changed by testing a set of different filter coefficients.

![Figure 3.20: Input referred noise for different filter settings.](image)

From the figure above, an optimal filter setting can be found for minimizing the input referred noise. Sweeping through some of the filter settings, a clear trend can be identified and a specific filter coefficient can be isolated of being the main contributor to this input referred noise.
3.4 Quantization noise

In this section, the quantization error averaged over the whole input range, as calculated in equation 1.17, is presented.

3.4.1 Combinations two

By looking at fig.3.21 and comparing it with fig.3.4, it can be seen that the quantization noise are correlated with the input range, where a higher input range results in a higher value of quantization noise due to the fact that the mean code with, which is the main contributor to the quantization error, is increased.

Figure 3.21: Quantization noise of combinations two.

The results are, just as for the input ranges, grouped into two sets of combinations with high and low values of ADC gain.
3.4.2 Combinations three

The same thing applies for the combinations three, where results of the quantization noise are also directly related to the input range of the ADC as can be seen by comparing fig.3.5 with 3.22 below.

(a) Quantization noise.

(b) Zoomed, gain 0.

(c) Zoomed, gain 15.

Figure 3.22: Quantization noise of combinations three.
3.4.3 Individual register sweep

fig.3.23 below shows how the different registers individually affect the quantization noise.

(a) Quantization noise, Offset mode.
(b) Quantization noise, ADCGain.

(c) Quantization noise, LatchCurrent.
(d) Quantization noise, Pre-ampCurrent.

(e) Quantization noise, Comparator Start.
(f) Quantization noise, common mode adjust.

**Figure 3.23:** Quantization noise individual register sweep.
The results from the individual register testing shows how the quantization noise relates to the input ranges.

### 3.4.4 Filter

The quantization noise depending on how the filter coefficients are set can be seen in fig.3.24.

![Quantization noise](image)

(a) Quantization noise.

**Figure 3.24:** Quantization noise for different filter settings.
3.5 INL/DNL

The data points of the DNL and INL is fitted to a polynomial which makes it easier to see the trend since simply plotting the values would result in a quite scattered plot, as can be seen in fig.3.25 where a dummy plot illustrates the process of fitting a curve to a scattered data set.

![Figure 3.25: Fitted data.](image)

The plots shown in the following sub sections are the estimated differential and integral non linearity of the ADC, where each line corresponds to one of the setting combinations. The reader should keep in mind that the results from the full input range of the ADC is cropped and distorted.
3.5.1 Combinations two

As showed in fig.3.26, the input range of the ADC has a rather constant DNL for a range of output codes. The set of combinations with ADC gain 0, marked with green, all seem to have a DNL lower than 0 for the lower and higher values while the combinations with the gain equal to 15 experience a larger spread, with the possibility of the DNL being both positive and negative for the lower and higher values.

![DNL and INL plots](image)

**Figure 3.26:** DNL and INL for combinations two.

This gives an almost identical INL for all of the register combinations with an ADC gain equal to 0 and a slight spread for those with an ADC gain of 15, as seen in fig.3.26 (b).
### 3.5.2 Combinations three

The DNL for the settings with an ADC gain of 0 is rather similar for combinations three and combinations two. However, the DNL for the settings with an ADC gain of 15 were closer to 0 for the combinations three than combinations two, as seen in fig.3.27.

![DNL and INL for combinations three](image)

**Figure 3.27:** DNL and INL for combinations three.

The INL is centered around 0 and is quite similar for all of the settings, regardless of the ADC gain, for the combinations three.
3.5.3 Individual register sweep

The figures 3.28 and 3.29 both show how the settings individually affect the DNL and INL.

**Figure 3.28:** DNL and INL for individual register sweeps.
It can be seen from the figures above that the DNL and INL of the ADC is more or less constant, irregardles of the register settings since the deviations are so small that they are negligible.
3.5.4 Filter

The non linearities of the ADC for different filter combinations can be seen in fig.3.30 below.

\[\text{(a) DNL, filter.}\]

\[\text{(b) INL, filter.}\]

\textbf{Figure 3.30:} DNL and INL for different filter combinations.
3.6 Input referred noise of circuit A

It was shown that the input referred noise did not differ that much for different common mode levels within a reasonable bound. The estimated value of all the input referred noise of circuit A is presented in the table 3.1, where the values are normalized with respect to the value of the largest standard deviation.

Table 3.1: Input referred noise of circuit A

<table>
<thead>
<tr>
<th>Circuit A mode</th>
<th>Offset mode</th>
<th>Mean Noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Low</td>
<td>0.18065</td>
</tr>
<tr>
<td>1</td>
<td>Low</td>
<td>0.20547</td>
</tr>
<tr>
<td>2</td>
<td>Low</td>
<td>0.24420</td>
</tr>
<tr>
<td>3</td>
<td>Low</td>
<td>0.29023</td>
</tr>
<tr>
<td>4</td>
<td>Low</td>
<td>0.33125</td>
</tr>
<tr>
<td>5</td>
<td>Low</td>
<td>0.41822</td>
</tr>
<tr>
<td>6</td>
<td>Low</td>
<td>0.59631</td>
</tr>
<tr>
<td>7</td>
<td>Low</td>
<td>0.94068</td>
</tr>
<tr>
<td>0</td>
<td>Mid</td>
<td>0.18093</td>
</tr>
<tr>
<td>1</td>
<td>Mid</td>
<td>0.20880</td>
</tr>
<tr>
<td>2</td>
<td>Mid</td>
<td>0.24669</td>
</tr>
<tr>
<td>3</td>
<td>Mid</td>
<td>0.29889</td>
</tr>
<tr>
<td>4</td>
<td>Mid</td>
<td>0.33956</td>
</tr>
<tr>
<td>5</td>
<td>Mid</td>
<td>0.42375</td>
</tr>
<tr>
<td>6</td>
<td>Mid</td>
<td>0.62098</td>
</tr>
<tr>
<td>7</td>
<td>Mid</td>
<td>0.99684</td>
</tr>
<tr>
<td>0</td>
<td>High</td>
<td>0.21522</td>
</tr>
<tr>
<td>1</td>
<td>High</td>
<td>0.23532</td>
</tr>
<tr>
<td>2</td>
<td>High</td>
<td>0.26217</td>
</tr>
<tr>
<td>3</td>
<td>High</td>
<td>0.31490</td>
</tr>
<tr>
<td>4</td>
<td>High</td>
<td>0.35531</td>
</tr>
<tr>
<td>5</td>
<td>High</td>
<td>0.44994</td>
</tr>
<tr>
<td>6</td>
<td>High</td>
<td>0.63816</td>
</tr>
<tr>
<td>7</td>
<td>High</td>
<td>1</td>
</tr>
</tbody>
</table>
A procedure for extracting data from the ADC by applying an input voltage sweep has been developed where an arbitrary number of different ADC settings can be tested automatically. Since the labview software and the octave scripts that was written during this thesis were proven to sucessfully being able to esti-mate the characteristics of an ADC, this report can be used as a manual for future usage testing of ADCs.

The major down side of this testing procedure is that it is rather time consuming. Performing a test that generates exercise the full input range of the ADC for one fixed combination of settings takes about 25 minutes and roughly multiplies with the number and size of the settings to be tested. This may not be optional due to the limited number of PXIe systems available in the lab, but an upside to this is that the testing is fully automated, giving the opportunity of leaving the test equipment running over nights and weekends. The time consumption of the testing procedure can be traded with the step size of the input voltages from the PXIe system together with the number of samples taken for each input voltage and setting combination.

From the results chapter, it could be seen how the registers affected the characteristics individually. It should be noted that a lot of the behaviour shown in the figures are just as expected. Some of the characteristics goes hand in hand with each other, such as the input referred shift which can almost directly translate to an offset in terms of number of codes. The same goes for the quantization noise which directly relates to the input range of the ADC, since a reduced input range will reduce the width of the codes, thus resulting in smaller quantization steps. The deviations of the input range for the different settings are in its turn related to the non linearities, i.e. some of the codes might experience a reduction in width which results in a reduction of the total input range. It was also seen that
the circuit A was the main contributor of the input referred noise in the signal chain.

This results extracted from this testing and estimation process has provided the possibility of locating optimal register values with respect to the desired characteristics of the ADC. It has also brought indications on behaviours which can be good to take into account if one would think of the idea of developing an updated version of the ADC in the future.

Having these characteristics at hand when operating the measurement device, the user can take the behaviour of the ADC into account when if the ADC would to be used in an application which has a high demand on the accuracy of how well the ADC digitally represent the incoming analog voltage.

4.1 Future work

One of the things that could be improved is to integrate the sorting of the returned data files together with the conversions to binary format in to the LabView program. This would also come with the benefit of saving some space on the hard drive of the lab computer, since binary files take up less space than text files.

The function which calculated the mean code width used for estimating the non linearities could be improved in the future to include some functionality of automatically analyzing the conversion data and locating the optimal range of the ADC where it is the most linear.

The testing of the ADC was carried out when disabling the offset removal feature in the ADC. It would be interesting to see how this feature would improve the characteristics estimated in this thesis.
Bibliography


