Simulation model of Synchronous Detection in a UV Monitor

Peter Olofsson
Simulation model of Synchronous Detection in a UV Monitor

Peter Olofsson
Abstract

A UV monitor is a part of a modular device that performs separation of protein solutions, for scientific use. It utilizes UV-light absorption to detect the type of proteins passing through it. Noise in the measurement signal affects the accuracy in the detection. For this project, a simulation model was produced in Matlab/Simulink to examine if synchronous detection could be implemented on a current hardware design, and if it could possibly improve the noise performance. Besides answering these questions, filters were designed to be used in a real-world test environment.
Acknowledgements

Special thanks to Anders Eriksson and Mikael Westin of GE Healthcare Umeå, Abdiqani A. Hirsi, Agneta Bränberg & Ulf Holmberg of Umeå University.
List of Figures

1  Overview of the chromatography process. 2
2  Example of Synchronous detection. 4
3  Example of a successive approximation ADC. 6
4  Example of a sigma-delta ADC. 7
5  Example of aliasing. 8
6  Controlling the phase of the carrier wave. 9
7  Using quadrature detection. 10
8  The complete Simulink model. 11
9  How UV detection works. 12
10 The transimpedance amplifier circuit. 13
11 The input test-signal spectrum. 15
12 First filter stage. 16
13 The signal after the first decimation stage. 17
14 Second filter stage. 18
15 The signal after the second decimation stage. 19
Contents

1 Introduction 1
  1.1 Background 1

2 Theory 3
  2.1 Noise 3
    2.1.1 Thermal Noise 3
    2.1.2 1/f Noise 3
    2.1.3 Shot Noise 3
    2.1.4 Quantization Noise 4
  2.2 Interference 4
  2.3 Synchronous Detection 4
  2.4 ADC:s 5
    2.4.1 Successive Approximation ADC:s 5
    2.4.2 Sigma-Delta ADC:s 6
  2.5 Filters 7
  2.6 Decimation 8
  2.7 AM Encoding/Decoding 9
  2.8 Processor Capacity & Filter Length 10

3 Method 11
  3.1 Simulation Model of the UV Monitor 11
    3.1.1 Digital-to-Analog Converter (DAC) 11
    3.1.2 Transconductance Amplifier 12
    3.1.3 UV-LED & Photodidode 12
    3.1.4 Transimpedance Amplifier 12
    3.1.5 Analog-to-Digital Converter 13
    3.1.6 Band-pass Filter 13
    3.1.7 AM Decoder 14


1 Introduction

1.1 Background

GE Healthcare (a subsidiary of General Electric) provides transformational medical technologies and services helping to deliver patient care to people around the world. The company provides medical imaging and information technologies, medical diagnostics, patient monitoring systems, drug discovery, biopharmaceutical manufacturing technologies and performance solutions services.

This project targets the UV monitor, which is a part of a modular system that is used to separate proteins dissolved in liquids. The liquid flows through plastic tubes, and by means of UV-light absorption, the UV monitor detects whether protein A or protein B is passing through it. The system then instructs the hardware to dispense the liquid into either container A or B. Figure 1 shows an overview of this procedure. At the testing stage for the UV monitor, a computer extracts test values to determine if the particular unit meets the desired standards.

![Diagram of UV monitor system](http://www.gelifesciences.com/gehcls_images/GELS/Related\%20Content/Files/1437474708000/litdoc29010831_20150721123116.pdf)

Figure 1:
In the chromatography process, liquids are mixed according to specified proportions. The proteins are added to the solution, and then passed through a column. Different proteins will flow through the column at different speeds, and the solution can then be analysed by several different detectors - UV-absorbance, conductivity or pH-value. These control the fraction collector, dispensing the liquid into the appropriate test tube. Source: http://www.gelifesciences.com/gehcls_images/GELS/Related\%20Content/Files/1437474708000/litdoc29010831_20150721123116.pdf
Noise in the measurement signal means that the precision of the instrument is affected, which leads to a lower degree of separation of the proteins. In practice, liquids are run through the unit several times to improve separation, but this would be less of an issue if the precision of the UV monitor could be improved.

Engineers Anders Eriksson and Mikael Westin of GE Healthcare Umeå initiated this project in February, and were looking for engineering students to do some research on the subject. Myself and Abdiqani A. Hirsi were chosen for this job.

Anders and Mikael had previously created a test environment for the UV monitor, in which a Raspberry Pi computer “bypassed” the on-board microcontroller. They also came up with the idea to use synchronous detection, to minimize the influence of noise.

This project aims to produce a simulation model of the UV monitor system with synchronous detection applied, as well as designing adequate filters for it. The simulation model will be produced in Matlab/Simulink, and the filter will be tested and implemented in Matlab. The goal is to find out if synchronous detection can be implemented on current hardware, and at the same time, provide filtering coefficients for future implementation.
2 Theory

This chapter was written to better understand the influence of noise and how it applies to the UV monitor.

2.1 Noise

There are several types of noise, with different causes and characteristics. The following is a brief description of the major types of noise:

2.1.1 Thermal Noise

Also called Johnson or Johnson-Nyquist noise, thermal noise is dependent on temperature - a hot resistor produces more noise than a cold one. Every resistance exhibits thermal noise [4]. The amount of thermal noise is constant over the frequency spectrum. Thermal noise grows with increased resistance:

$$v_n = \sqrt{4k_BT\Delta F}$$  \hspace{1cm} (2.1)

where $k_B =$ the Boltzmann constant, $T =$ temperature in degrees Kelvin, $R =$ resistance in ohms and $\Delta F =$ bandwidth in Hertz [4].

Thermal noise is caused by the fact that electrons move and collide with each other, and the higher temperature leads to more electron movement and more collisions. This is unavoidable, and thermal noise is present in all electronic components, since they all exhibit some degree of resistance.

2.1.2 1/f Noise

1/f noise or flicker noise, origins from fluctuations in resistance. The lower the frequency, the higher the noise, like the name 1/f implies. It is present in most electronic components, as a result of manufacturing imperfections [4].

The corner frequency, $f_c$, is the frequency where the 1/f noise falls below the thermal noise floor, and it's contribution to the overall noise disappears. The corner frequency varies between different components.

2.1.3 Shot Noise

When electrons crosses barriers, such as inside a diode or a transistor, they produce shot noise. This happens because the electrons all have discrete arrival times. It can be likened to raindrops falling on a roof - the raindrops hit the roof at slightly different points in time,
producing the characteristic “pitter-patter” sound. The noise is constant over the frequency spectrum [4]. Shot noise is given by the formula:

\[ i_n = \sqrt{2Iq\Delta F} \]  

where \( I = \text{DC-current} \), \( q = \text{the charge of an electron} \) and \( \Delta F = \text{bandwidth in Hertz} \) [4].

### 2.1.4 Quantization Noise

When an electronic voltage is sampled and quantized to a discrete amplitude value, there will be a quantization error, since the discrete value is not an analog of the electronic voltage. This error manifests itself as noise in the signal. In reality, the quantization error produces harmonic distortion, not noise. Through a process called dithering, noise is added to the signal prior to quantization, and this (ideally) avoids the distortion completely, at the cost of a slightly higher noise floor [6]. More on quantization noise in the section on ADC:s.

### 2.2 Interference

Other sources of unwanted noise include the AC mains frequency, 50 or 60 Hz, due to capacitive and inductive coupling. This interference can be very strong, depending on the physical design of the circuit(s). Proper shielding and grounding of the equipment is very important, not only for personal safety but also to suppress interference [1].

### 2.3 Synchronous Detection

The influence of \( 1/f \) noise can be minimized, by modulating the low-frequency signal’s amplitude with a high-frequency signal. This moves the signal of interest out of the noisy low-end of the spectrum. Above a certain frequency, the \( 1/f \) noise falls below the thermal noise floor (labelled “White noise” in the graphs). This is the corner frequency \( f_c \):

![Figure 2: Synchronous detection, showing the signal of interest moved away from the noisy region near DC. Source: Peter Olofsson](image)

As can be seen in figure 2, the noise increases quite dramatically in the near-DC region. The current technique used inside the UV monitor switches the UV-LED on and off, at a rate of 10 Hz. The signal of interest lies in the 0 - 5 Hz area.

Synchronous detection also avoids any DC-offset in the signal. DC-offset is temperature-dependent, which can cause long-term noise (drift).
2.4 ADC:s

Converting an analog voltage to digital works by sampling the signal according to a given sampling frequency, and quantize each sample to a discrete voltage value. The voltage value corresponds to a binary code. This process is called Pulse Code Modulation (PCM).

The number of bits determines the maximum dynamic range (signal-to-noise ratio). Every bit equals 6.02dB of dynamic range. An 8-bit converter will have \(2^8 = 256\) discrete voltage values, and \(6.02 \times 8 = 48.16\)dB of dynamic range. This is means that the quantization noise is 48.16dB lower than the maximum signal peaks. Dithering noise will shrink the dynamic range by about 5dB [5, 6].

The sampling frequency determines the bandwidth. If an analog signal has bandwidth \(B\), a sampling frequency \(\geq 2B\) must be used to preserve the signal. The Nyquist frequency is exactly half the sampling frequency, and the input signal must be kept strictly below this frequency [3].

Oversampling means sampling the signal \(n\) times higher than normally required, where \(n\) is an integer. This is done to allow for a more gentle anti-aliasing filter before the conversion. There are more reasons for oversampling, more on that in the section on Sigma-Delta ADC:s. A Nyquist converter is a converter that does not utilize oversampling, and thus requires a very steep anti-aliasing filter (a.k.a. brick-wall filter) [5].

2.4.1 Successive Approximation ADC:s

Successive approximation samples and holds the analog input signal, and searches in a register for a corresponding binary code by comparing the signal to a reference voltage. If the full scale of the ADC is 0 - 5 Volts, then a signal of more than 2.5 Volts equals a Most Significant Bit (MSB) value of 1. The signal is fed back into the comparator and each bit is set successively through comparison [2]. See figure 3.
Successive approximation utilizes high bit-depths, and higher bit-depths requires the circuit to generate very precise reference voltages. A 24-bit converter has to be 256 times more precise than a 16-bit converter. This is costly from a manufacturing perspective.

### 2.4.2 Sigma-Delta ADC:s

Sigma-delta ADC:s use a very low bit-depth (typically 1 to 3 bits), along with a very high sample rate (typically 64 times higher than in a Nyquist converter). A 1-bit converter has only 6.02dB of dynamic range, but sigma-delta conversion pushes the noise up in the higher frequency range, where it is less damaging. This results in a dynamic range much greater than the bit-depth would indicate [2, 5, 6].

In a sigma-delta converter, the input voltage is integrated, sampled/quantized, and then fed back and subtracted from the input voltage. The output signal can be described by:

\[ y[n] = x[n] + e[n-1] \]  

where \( x[n] \) is the input signal, \( y[n] \) is the output signal and \( e[n] \) is the quantization error. Put into words, the error of the previous sample is added to the current sample, before being quantized into a binary code (1 or 0 in this case) [2].

A feedback loop always acts as a filter, but in the sigma-delta converter, the filter only acts on the noise, creating what is called “noise shaping”. Several sigma-delta stages can be cascaded to form 2nd- and 3rd-order sigma-delta converters, which has the same effect as cascading any kind of filter - the slope gets steeper. In this case, it equals a more effective high-pass filtering of the noise, while the signal of interest stays unaffected. This explains why it is possible to attain a very high dynamic range from a 1-bit converter [5]. See figure 4 for an overview of a single sigma-delta stage.

Note that this is a simplification of the process, for practical purposes. Processing the digital signal usually requires it to be transformed into a higher bit-depth and lower sample rate,
and decimation takes care of this inside the ADC, see section on decimation below [5]. Sigma-delta conversion is cheaper and easier to implement than successive approximation, and it also consumes less power.

![Diagram of a single-stage delta-sigma ADC](image)

**Figure 4**: The feedback loop of a single-stage *delta-sigma* ADC. Source: Peter Olofsson

### 2.5 Filters

Filters are an enormous subject, and there are many types available to us. Fortunately, the evolution of digital technology has simplified the process of filter design (and implementation) considerably. The following applies to low- and high-pass filters, as well as band-pass and notch filters.

To narrow down our choice of filters, we first have to settle for Infinite vs Finite Impulses Response (IIR vs FIR). FIR filters are a natural choice for digital signal processing, because they are unconditionally stable, compared to IIR which are prone to instability (controlled or uncontrolled oscillations). In addition to that, FIR filters are phase-linear in the passband, unlike IIR [3].

Filters used for anti-aliasing must sufficiently attenuate all frequencies from Nyquist (half the sample rate) and upwards. Sufficient attenuation will be determined by the quantization noise - it would mean that for a bit-depth of 16 bits, the noise floor will be 96dB below the signal, and thus require 96dB of attenuation to completely avoid aliasing. Any signal above Nyquist will be folded back and added on top of the signal of interest, in other words the signal is aliased. (However, the aliasing may fall below the noise floor, if proper filtering is applied before sampling) [3]. See figure 5 for an example of aliasing.

---
It should be added that aliasing is a kind of distortion of the signal that cannot be undone - once a signal is affected by aliasing, no amount of filtering can remove it. Noise is also folded into the signal, but it doesn’t produce aliasing - it decreases the signal-to-noise ratio. Although this may not be a problem, it is still irreversible, like aliasing.

This is the formula for calculating aliasing frequencies:

\[ | \pm f_a \pm f_N | \]

where \( f_a \) = the frequency above Nyquist and \( f_N \) the Nyquist frequency. Of course, any frequency that ends up outside the bandwidth can be ignored.

### 2.6 Decimation

Decimation is the practice of reducing the sample rate (downsampling) by integer division. The same rules apply to decimating as to sampling - any signal above Nyquist (of the post-decimation signal) will be folded back into the passband, possibly creating unrecoverable distortion of the signal of interest [3].

The actual reason for decimation is that, with a lower sample rate, the DSP calculations becomes much simpler, as the processor has less samples-per-second to process. In addition, the length of the filter itself is reduced - decimation by 2 (half the original sample rate) reduces the filter order by 50%. In a nutshell - more time to perform fewer calculations.

The optimum type of FIR filter for anti-aliasing purposes is the Equiripple, (a.k.a. Parks-McClellan). The name equi-ripple stems from the fact that the ripple that occurs is evenly distributed between the passband and stopband. The equiripple filter also offers control over the degree of ripple [8].
2.7 AM Encoding/Decoding

AM encoding works by simply multiplying the signal of interest with a higher carrier frequency, creating an amplitude-modulated signal. This moves the original signal from its original frequency-band to the frequency band of the carrier. To extract the original signal from the modulated signal, the modulated signal is multiplied with the carrier frequency. The carrier and the carrier component of the modulated signal must be in phase for optimum signal-to-noise ratio. Decoding of the AM-modulated signals renders the original, baseband signal as well as a doubled carrier frequency, both at half the level of the modulated signal:

\[ \sin(x) \times \sin(x) = \sin^2(x) = \frac{1 - \cos(2x)}{2} = \frac{1}{2} - \frac{1}{2}\cos(2x) \]  

(2.4)

The demodulation turns the modulating factor \( \sin(x) \) (the 110 Hz carrier) into a factor \( \frac{1}{2} \) and another factor \( -\frac{1}{2}\cos(2x) \), so the original signal is multiplied by these factors, instead of the carrier wave. However, if there is any DC present in the AM-modulated signal, the AM-decoding will not remove the carrier frequency completely or at all. If there’s a DC component of 1, so that

\[ \sin(x) \times (\sin(x) + 1)) = \sin^2(x) + \sin(x) = \frac{1 - \cos(2x)}{2} + \sin(x) = \frac{1}{2} - \frac{1}{2}\cos(2x) + \sin(x) \]  

(2.5)

the carrier signal \( \sin(x) \) will persist. Each of these factors will multiply with the original signal, so that with a DC component of 1, the original signal is modulated once again by the carrier.

Because of phase shift in the signal chain of the AM-modulated signal, the phase of the carrier frequency has to be adjusted. Keeping the signals in phase at the demodulation stage can be done by a simple delay stage.

Figure 6: The carrier wave \( c(t) \) is kept in phase with the carrier component of the AM-modulated signal using a simple delay stage. Source: Anders Eriksson
Another way is to use quadrature detection - this method is phase-sensitive, and does not rely on both signals being in phase.

Figure 7: Using quadrature detection, in the demodulation stage, the AM-modulated signal is split into two signals. One is multiplied with the carrier wave $c(t)$, and the other with the time-derivate of $c(t)$. Both signals are then low-pass filtered and added together, creating phase-sensitive AM-detection. Source: Anders Eriksson

Quadrature detection is very practical, since it always ensures optimum S/N-ratio, whether the delay is static or varies (like in the case of inserting different filters in the signal chain, since different filters have different phase responses). See figures 6 and 7 for a comparison of the two different methods of AM detection.

### 2.8 Processor Capacity & Filter Length

A microprocessor (or a microcontroller in the case of the UV monitor) has limited processing power. Because of this, the length of the filters must be kept as short as possible, while still providing effective filtering. The filter length is equal to the number of filter taps. The number of taps is equal to the filter order - 1. A higher sample rate equals more taps, as do steeper filter slopes. Increased stop-band attenuation usually leads to more taps, but not necessarily [3].
3 Method

The project was initialized by developing a model in Simulink, to get a better understanding of how noise affects the system. Starting off simple, it gradually developed into a more advanced system requiring the use of “sub-systems”, for easier management and better overview. Therefore, it’s necessary to look at each sub-system separately, to get an idea of the interior workings. Figure 8 shows the complete Simulink model.

Later on, focus was shifted towards coding a simplified Matlab model, to better control and understand the decimation and filter parameters. Different filters were tested and generated, for use in the aforementioned real-world hardware test-bench.

3.1 Simulation Model of the UV Monitor

![Simulink model](image)

**Figure 8:** The Simulink model, with all its sub-systems hidden inside discrete blocks. Source: Peter Olofsson

What follows is a brief description of each component in the system, along with closeup views of the system.

3.1.1 Digital-to-Analog Converter (DAC)

The digital-to-analog converter outputs a 110 Hz sine wave generated inside the microcontroller. This is the carrier signal, that modulates the amplitude of the absorption signal. A DC-bias is added to the signal in order to drive the UV-LED. The DAC operates at a sample rate of 2640 Hz, because it has to generate a relatively smooth sine wave without a recon-
The choice of carrier frequency for the implementation of synchronous detection was set to 110 Hz, as this keeps it in-between multiples of any 50- or 60 Hz AC frequencies. A higher frequency could have been chosen, but this would not improve upon the S/N significantly. It would however require a lot more processing power, and since the goal is ultimately to implement synchronous detection on the current hardware design, higher frequencies (meaning higher sample rates) should be avoided.

### 3.1.2 Transconductance Amplifier

The transconductance amplifier converts the voltage from the DAC output to a current to control the UV-LED, as its emission of light is linear to current, not voltage.

### 3.1.3 UV-LED & Photodiode

The Ultraviolet LED is a current-controlled device. The photodiode is a light-sensing device, converting light to current. The LED illuminates the liquid proteins, as they pass through the system, through a clear tube in front of a mirror. The transmitted light is measured by the sample detector, and the reflected light is measured by the reference detector. The absorption signal $\text{ABS}$ is given by $\text{ABS} = \log_{10}\left(\frac{R}{S}\right)$, where $R$ is the reference signal and $S$ is the sample signal. This forms an optic coupling between the diodes, and is labelled as such in the diagram. Figure 9 shows the UV detection technique used in the UV monitor.

![Figure 9: UV detection](image)

**Figure 9:** UV detection, showing how the UV-LED emits light through a semi-transparent mirror and through the flow-cell. The photodiodes senses the reference signal and the sample signal. Source: Peter Olofsson

### 3.1.4 Transimpedance Amplifier

The transimpedance amplifier converts the (tiny) photodiode output current to a voltage. Calculating the TIA noise can be simplified if we omit the frequency-dependent noise (1/f), which only becomes an issue at very low frequencies. Since the current is indeed small, the sensitivity to noise is very high. Since the feedback resistor of this amplifier circuit has to be 500 Mohms, the only choice is to use a thick-film resistor. The thick-film resistor unfortunately has a higher 1/f noise than thin-film. Figure 10 shows the transimpedance
amplifier circuit.

![Amplifier Circuit Diagram]

**Figure 10:** The *transimpedance* amplifier circuit, converting the small photodiode current to a voltage, ready for ADC conversion. Source: Peter Olofsson

### 3.1.5 Analog-to-Digital Converter

The analog-to-digital converter does just what it says. Noise added by the ADC is dependent on the bit-depth, more (effective) bits equal less noise. By effective, we mean the number of bits in actual use - if the input signal amplitude is very small, less bits are utilized and noise increases [6].

### 3.1.6 Band-pass Filter

The band-pass filter attenuates DC from the signal, which would otherwise interfere with the AM decoding. It also attenuates 50/60Hz AC hum, and noise from the TIA and ADC. The most important part for this particular filter is to block DC from the modulated signal. If it fails to do so, the AM decoder will not be able to decode it properly. In addition to DC-blocking, we want the filter to attenuate everything except the modulated signal.
3.1.7 AM Decoder

The AM decoder multiplies the modulated signal with the 110 Hz carrier, in order to obtain the original signal. These two signals must be in phase. This was implemented by using quadrature detection - see section on AM Encoding/Decoding in the Theory chapter. This method proved to be highly convenient when trying out different band-pass filters, because each filter has a different phase response. To manually set the delay to compensate for phase response would have been impractical.

3.1.8 Low-pass Filter

Like the band-pass filter, the low-pass filter also attenuates other noise sources in the signal chain. The low-pass filter has to attenuate all frequencies above 5 Hz, but especially the 220 Hz by-product, that is created at the AM decoding stage.

3.2 Moving On

The first weeks of the project were spent producing the Simulink model, and as time went by, the focus shifted towards the low-pass filter. The original sample rate of 2640 Hz was lowered to 660 Hz. At first, dummy test signals were used to adjust filter parameters of both the low-pass and band-pass filters, containing 3, 50, 60, 110 and 220 Hz signals as well as noise. The 3 Hz signal represents the absorption signal, which would vary between 0 to 3 Hz, typically. The 50 and 60 Hz signals represent the two AC mains frequencies used around the world, and the 110 and 220 Hz signals represent the carrier frequency and its by-product from the demodulation stage. Later on, test signals were recorded from the UV monitor/Raspberry Pi. These recordings only contained a 220 Hz and baseband signal - the carrier was unmodulated. As can be seen in figure 11, the baseband signal is a DC, carrying no UV absorption signal.
3.3 Design of the Low-pass Filter

In order to reduce the data rate and implementation cost, it was decided to apply decimation to the low-pass filter. The decimation was distributed between two stages. First, a low-pass filter followed by downsampling by 11, and then another low-pass filter followed by downsampling by 6. In other words, the sample rate was lowered from 660 to 60 Hz after the first stage, and then from 60 to 10 Hz at the last stage. Below is the magnitude and impulse response of the first filter. Notice that the stopband starts at 50 Hz - the point behind this was to avoid AC hum to contaminate the signal. Figure 12 shows the frequency response of the first low-pass filter.
Figure 12: The frequency response of the first low-pass filter. Note that the stop-band attenuation is only 40dB here, but it is only the first of two cascaded filters, forming one effective filter. Source: Peter Olofsson

This signal was then fed in to the first decimation stage, see figure 13:
Figure 13: The signal after the first decimation stage, showing a frequency range of 0 - 30 Hz, as the sample rate has been decimated by a factor 11, from 660 Hz down to 60 Hz. The 20 Hz signal comes from aliasing of the 220 Hz signal, but it is attenuated by almost 100 dB, and thus of little concern. Source: Peter Olofsson

The second filter prepared the signal for the second and final decimation stage, see figure 14:
Figure 14: The frequency response of the second low-pass filter. Thanks to decimation, this filter can be made very effective, while still keeping the filter length relatively short. Source: Peter Olofsson

The 60 Hz sample rate could then be reduced to 10 Hz, see figure 15.
Figure 15: The 60 Hz signal has now been decimated by a factor 6 down to a 10 Hz sample rate, making the bandwidth only 5 Hz, or actually strictly less than 5 Hz, but this is perfectly sufficient for its application. Source: Peter Olofsson
4 The Results

Synchronous detection can most likely be implemented using the onboard microcontroller of the UV monitor, since it’s possible to substantially lower the sample rates and use decimation, and in turn reduce the calculation requirements.

Filters have been developed, and the coefficients have been applied to the UV monitor/Raspberry Pi test-bench.

As the test-signal was fed into Matlab, it was discovered that the noise was a bit higher than predicted. Again, this could be due to inferior components in the UV monitor circuit. No 50 Hz AC interference could be seen, however.

4.1 Discussion

Implementation of synchronous detection does not necessarily improve upon the current design of the UV monitor.

The primary source of noise in the measurement signal is believed to originate from the 500 Mohm thick-film resistor in the transimpedance amplifier. Increasing the feedback resistance would result in a lower noise from the TIA, because:

\[ V_{out} = -I_pR_f \]

\[ v_n = \sqrt{4k_BT\Delta F} \]

In other words, TIA amplification increases with R, while the thermal noise in the feedback resistor increases with the square root of R.

The UV monitor circuit used for testing had previously been rejected at the manufacturing tests, showing inferior values. This could mean that one or more components are faulty, so the test-bench for this project may be at a disadvantage. However, that could be seen as an advantage as well, because it is desirable to create a noise-suppressing system that fights all kinds of noise, regardless of its origin.

Interference from AC mains could not be detected in the test-signal. Presence of AC interference was just a logical assumption when creating the simulation model in Matlab/Simulink, prior to any testing with real-world signals.

Decimation was used for the low-pass filter only. Decimation could not be used for the band-pass filter because the sample rate was already at the limit - 660 Hz yields a Nyquist frequency of 330 Hz. The modulated signal at 110 Hz would have to be demodulated to extract the information signal, and this would generate a 220 Hz signal as well. If the sample rate was already decimated to 330 Hz at that stage, leading to a Nyquist frequency of 165 Hz, the 220 Hz signal would have been aliased.
Actually, the reason the 2640 Hz sample rate was lowered to 660 Hz at the ADC was that it was found to be unnecessarily high - only 220 Hz needed to be preserved. Higher sample rate leads to longer filters, which we wanted to avoid. The sample rate at the DAC was kept at 2640 Hz, since it had to approximate a smooth sine wave through a stepped output signal, without filtering.

The band-pass filter and first low-pass filter are implemented using a 660 Hz sample rate, and therefore require more processing power than the second low-pass filter, operating at only 60 Hz. They should be made as steep as possible while taking the available processing power into account.

4.2 Future Investigations

*Polyphase filtering* is a technique that takes advantage of the fact that FIR filters are symmetrical. This means that the first and last coefficients are identical, the 2nd and 2nd last coefficients are identical, the 3rd and 3rd last coefficients are identical...and so forth. In the convolution process, these can be joined together and significantly reduce the number of calculations required [7].

*Noble Identity* is another concept that can reduce calculation requirements. It is done by switching the order of decimation and filtering, so that decimation is done before filtering, instead of vice-versa. The filters can then be implemented at a lower sample rate, reducing the filter length and increasing the available time to perform calculations.

The effect of noise folding should be examined. At the output transimpedance amplifier, there is an RC-filter (low-pass) that has its cutoff frequency set to the original 2640 Hz sample rate of the ADC. The sample rate of the ADC was lowered to 660 Hz however, and this was not taken into account. This means that the bandwidth of the TIA output is much wider than it should be, and while no aliasing effects could be noticed, a wider bandwidth contains more noise. Folding of this noise could in theory lead to a lower signal-to-noise ratio.

Although the feedback resistance of the transimpedance amplifier is already very high, an even higher resistance is likely to increase the signal-to-noise ratio. This should definitely be examined.
References


