Implementation Considerations for Active Noise Control in Ventilation Systems

Marcus Asteborg
Niklas Svanberg

Department of Signal Processing
Blekinge Institute of Technology
Box 520
SE - 372 25 Ronneby
Sweden
This thesis is submitted to the School of Engineering at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical Engineering. The thesis is equivalent to 2 x 10 weeks of full time studies.

Contact Information:
Authors:
Marcus Asteborg, 820430-0357
Address: Nebulosavägen 27, 175 60 Järfälla, Sweden
E-mail: asteborg@kth.se

Niklas Svanberg, 830604-3350
Address: Minervavägen 22b, 371 41 Karlskrona, Sweden
E-mail: nisv03@student.bth.se

University advisor and examiner:
Sven Johansson
Department of Signal Processing, BTH

University advisor:
Martin Larsson
Department of Signal Processing, BTH

Department of Signal Processing
Blekinge Institute of Technology
Box 520
SE - 372 25 Ronneby
Sweden

Internet : www.bth.se/tek/ask
Phone : +46 457 38 50 00
Fax : +46 457 271 25
Abstract

The most common method to attenuate noise in ventilation systems today is passive silencers. For these to efficiently attenuate frequencies below 400 Hz such silencers need to be large and a more neat solution to attenuate low frequencies is to use active noise control (ANC). The usage of ANC in ventilation systems is well known and there are several commercial products available. ANC is not, however, used on a wide basis due to its often high price and poor performance.

Since the price is an important factor in ANC systems the expensive laboratory filters and the amplifier that is currently used in the experimental setup at Blekinge Institute of Technology (BTH) need to be replaced with cheaper ones, but without too much performance loss. For easier implementation in ventilation systems the placement of the reference microphone is important, the shorter distance from the anti-noise loud speaker the easier the ANC system is to implement. But if the distance is so small that the ANC system is no longer causal the performance will be decreased and if the reference microphone is close enough to pick up acoustic feedback from the anti-noise loud speaker the performance will also be decreased. In this thesis the expensive laboratory filters will be exchanged to cheaper alternatives, power and total harmonic distortion (THD) measurements will be done on the amplifier that is driving the loud speaker and the reference microphones position will be investigated with measurements on the group delay of the system and the acoustic feedback between the loud speaker and the reference microphone.
# Contents

## 1 Introduction
- 1.1 Purpose ................................................................. 1
- 1.2 Limitations ............................................................. 1

## 2 Active Noise Control
- 2.1 Introduction ........................................................... 3
- 2.2 Adaptive Filters ....................................................... 5
  - 2.2.1 Steepest Descent Adaptive Filter ............................... 6
- 2.3 Algorithms .............................................................. 7
  - 2.3.1 Least-Mean-Square Algorithm ................................. 7
  - 2.3.2 Normalized LMS .................................................. 8
  - 2.3.3 Filtered-X LMS ................................................... 9
- 2.4 Coherence .............................................................. 11
- 2.5 Causality .............................................................. 13

## 3 The Experimental Setup
- 3.1 Ventilation System .................................................. 15
- 3.2 Sensors ................................................................. 17
  - 3.2.1 The Microphones ................................................. 17
  - 3.2.2 Microphone Setup .............................................. 17
- 3.3 Actuator ............................................................... 18
  - 3.3.1 Loud speaker element ......................................... 18
  - 3.3.2 Amplifier ......................................................... 18
- 3.4 The Controller ....................................................... 19
  - 3.4.1 Digital Signal Processor ..................................... 19
  - 3.4.2 Input/Output ................................................... 19
- 3.5 Other Hardware ..................................................... 22
  - 3.5.1 Filters .......................................................... 22
  - 3.5.2 Signal Analyzer ................................................. 22
- 3.6 Software, Code and Setup ......................................... 25
  - 3.6.1 Forward Path Estimation .................................... 25
  - 3.6.2 Active Noise Control ......................................... 26
# 4 Analog Filters

4.1 Introduction .................................................. 29
4.2 Response of Filters ............................................ 30
4.3 Passive versus Active Filters .................................. 31
   4.3.1 Passive Filters ........................................... 31
   4.3.2 Active Filters ........................................... 31
   4.3.3 Summary .................................................. 33
   4.3.4 Poles and Zeros .......................................... 34
4.4 Analog Filters for Control Systems ......................... 36
   4.4.1 Anti-aliasing Filters .................................... 36
   4.4.2 Reconstruction Filter ................................... 38

# 5 Design and Implementation

5.1 Requirements ................................................ 41
5.2 Filter Characteristics and Architecture ................. 42
5.3 Design ......................................................... 43
   5.3.1 Butterworth Filter ........................................ 43
   5.3.2 Chebyshev Type 1 Filter ................................ 44
   5.3.3 Elliptic Filter ............................................ 45
   5.3.4 Summary .................................................. 46
5.4 Implementation ............................................... 47
5.5 Testing ......................................................... 50
   5.5.1 SPICE Simulation ......................................... 50
   5.5.2 Realization ............................................... 51
   5.5.3 Summary .................................................. 52

# 6 Performance using Different Filters

6.1 Kemo VBF10M .................................................. 53
   6.1.1 Performance .............................................. 53
6.2 5th Order Filters ............................................. 54
   6.2.1 Performance .............................................. 55
   6.2.2 Summary .................................................. 58
6.3 10th Order .................................................... 59
   6.3.1 Design and Implementation ............................ 59
   6.3.2 Performance .............................................. 62
   6.3.3 Summary .................................................. 65
6.4 Maxim 280 .................................................... 66
   6.4.1 Implementation of Maxim 280 ......................... 66
   6.4.2 Performance .............................................. 69
   6.4.3 Summary .................................................. 71
6.5 Summary ....................................................... 71

# 7 Amplifier

7.1 Measurements and requirements ........................... 73
   7.1.1 Total Harmonic Distortion ............................. 74
7.2 QSC USA 370 .................................................. 75
   7.2.1 Performance .............................................. 75
   7.2.2 Total Harmonic Distortion ............................. 76
7.3 Kemo MO34 .................................................... 77
<table>
<thead>
<tr>
<th>CONTENTS</th>
<th>vii</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3.1 Performance</td>
<td>77</td>
</tr>
<tr>
<td>7.3.2 Total Harmonic Distortion</td>
<td>78</td>
</tr>
<tr>
<td>7.4 Sonic-T Amp</td>
<td>79</td>
</tr>
<tr>
<td>7.4.1 Performance</td>
<td>79</td>
</tr>
<tr>
<td>7.4.2 Total Harmonic Distortion</td>
<td>80</td>
</tr>
<tr>
<td>7.5 Summary</td>
<td>81</td>
</tr>
<tr>
<td>8 Confidential</td>
<td>83</td>
</tr>
<tr>
<td>9 Conclusions and Further Work</td>
<td>85</td>
</tr>
<tr>
<td>9.1 Conclusions</td>
<td>85</td>
</tr>
<tr>
<td>9.2 Further work</td>
<td>86</td>
</tr>
<tr>
<td>Bibliography</td>
<td>87</td>
</tr>
<tr>
<td>A MATLAB scripts</td>
<td>89</td>
</tr>
<tr>
<td>A.1 Analog filterdesign</td>
<td>89</td>
</tr>
<tr>
<td>A.1.1 Butterworth</td>
<td>89</td>
</tr>
<tr>
<td>A.1.2 Chebyshev type 1</td>
<td>90</td>
</tr>
<tr>
<td>A.1.3 Elliptic</td>
<td>91</td>
</tr>
<tr>
<td>B Filters</td>
<td>93</td>
</tr>
<tr>
<td>B.1 10th order</td>
<td>93</td>
</tr>
<tr>
<td>B.1.1 Calculations</td>
<td>93</td>
</tr>
<tr>
<td>B.2 Maxim 280</td>
<td>97</td>
</tr>
<tr>
<td>B.2.1 Trimmed Values, 5th Order</td>
<td>97</td>
</tr>
<tr>
<td>B.2.2 Trimmed Values, 10th Order</td>
<td>97</td>
</tr>
<tr>
<td>C C Code for Active Noise Control</td>
<td>99</td>
</tr>
<tr>
<td>C.1 Forward Path Estimation</td>
<td>99</td>
</tr>
<tr>
<td>C.2 Active Noise Control, Normalized FXLMS</td>
<td>102</td>
</tr>
<tr>
<td>C.3 Active Noise Control, Partial-update FXLMS</td>
<td>106</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Illustration of ANC system where P denotes the plant.</td>
<td>3</td>
</tr>
<tr>
<td>2.2</td>
<td>The concept of feedforward control</td>
<td>4</td>
</tr>
<tr>
<td>2.3</td>
<td>The concept of feedback control</td>
<td>4</td>
</tr>
<tr>
<td>2.4</td>
<td>Block diagram of an adaptive filter</td>
<td>5</td>
</tr>
<tr>
<td>2.5</td>
<td>Block diagram of LMS algorithm</td>
<td>7</td>
</tr>
<tr>
<td>2.6</td>
<td>Block diagram of FXLMS algorithm</td>
<td>9</td>
</tr>
<tr>
<td>2.7</td>
<td>Block diagram of system identification</td>
<td>10</td>
</tr>
<tr>
<td>2.8</td>
<td>Single-input-single-output system with noise at the output.</td>
<td>11</td>
</tr>
<tr>
<td>2.9</td>
<td>Linear time invariant system</td>
<td>12</td>
</tr>
<tr>
<td>2.10</td>
<td>(a) Power Spectral Density (PSD) measured at the error sensor. Solid line shows the attenuation when the ANC system is on. Dash dot line shows when the ANC system is off. (b) Coherence measurement between the reference sensor signal and the error sensor signal.</td>
<td>12</td>
</tr>
<tr>
<td>2.11</td>
<td>Group delays in system</td>
<td>13</td>
</tr>
<tr>
<td>3.1</td>
<td>The ventilation system in the research laboratory at Blekinge Institute of Technology (BTH).</td>
<td>15</td>
</tr>
<tr>
<td>3.2</td>
<td>Power Spectral Density (PSD) at the error sensor. Solid line shows the attenuation when the ANC system is off. Dash dot line shows when the ANC system is on.</td>
<td>16</td>
</tr>
<tr>
<td>3.3</td>
<td>Circuit scheme for microphone icp-boxes.</td>
<td>17</td>
</tr>
<tr>
<td>3.4</td>
<td>T-duct microphone setup [7].</td>
<td>18</td>
</tr>
<tr>
<td>3.5</td>
<td>T-duct speaker setup</td>
<td>18</td>
</tr>
<tr>
<td>3.6</td>
<td>Conversion from analog-to-digital and digital-to-analog domain.</td>
<td>20</td>
</tr>
<tr>
<td>3.7</td>
<td>(a) shows the two tones before the ADC and (b) shows the two tones after the ADC, DSP and DAC</td>
<td>21</td>
</tr>
<tr>
<td>3.8</td>
<td>Flowcharts of the forward path estimation program. (a) shows the main routine and (b) the interrupt routine.</td>
<td>25</td>
</tr>
<tr>
<td>3.9</td>
<td>Forward path estimation setup.</td>
<td>26</td>
</tr>
<tr>
<td>3.10</td>
<td>Flowcharts of the active noise control program. (a) shows the main routine and (b) the interrupt routine.</td>
<td>27</td>
</tr>
<tr>
<td>3.11</td>
<td>Active noise control setup.</td>
<td>28</td>
</tr>
<tr>
<td>4.1</td>
<td>Dashed line shows the ideal frequency response and the solid line shows the true frequency response, for each of the four common filter groups, (a) lowpass, (b) highpass, (c) bandpass, and (d) bandstop.</td>
<td>29</td>
</tr>
<tr>
<td>4.2</td>
<td>Circuit schemes for passive filters, where (a) is a lowpass filter and (b) is a highpass filter. The filter input is to left and output to the right.</td>
<td>31</td>
</tr>
<tr>
<td>4.3</td>
<td>Circuit schemes for active filters, where (a) is a one pole pair lowpass and (b) is a two pole pair lowpass.</td>
<td>32</td>
</tr>
</tbody>
</table>
4.4 Poles of a second order Butterworth filter in the s-plane, imaginary part \((j\omega)\) on y-axis, and real part \((\sigma)\) on x-axis ................................................................. 34
4.5 Solid line shows a sinusoid with frequency of 100 Hz, and sampled with \(f_s = 80\) Hz samples/sec in time domain ................................................................. 36
4.6 Magnitude of the frequency response of an anti-aliasing filter and the aliased characteristics used to calculate the required cutoff rate [17]. .................................................... 37
4.7 Shows the output of digital-to-analog converter (DAC) in left figure, and the smoothed out signal after the reconstruction filter in the right figure. ........................................ 39

5.1 Low pass filter requirements, (a) is for Butterworth, (b) is for Chebyshev type 1 and Elliptic. 41
5.2 General architecture for Sallen-Key of a two-pole filter section [18]. ................................. 42
5.3 Simulated responses of 5th order Butterworth lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees. ............. 43
5.4 Simulated responses of 4th order Chebyshev type 1 lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees. ............ 44
5.5 Simulated responses of 3rd order Elliptic lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees. ............. 45
5.6 Phase response of Butterworth, Chebyshev type 1 and Elliptic filters; solid line is Butterworth, semi dashed line is Chebyshev type 1 and dashed line is Elliptic. ......................... 46
5.7 Circuit scheme of 5th order Butterworth lowpass filter. .................................................. 50
5.8 Simulated filter response in SPICE of 5th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees. ................................................................. 50
5.9 Circuit scheme of a unity buffer amplifier. ................................................................. 51
5.10 Real responses of built 5th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees. .... 51
5.11 Phase response of built 5th order Butterworth filter vs Kemo filter; the solid line is the built 5th order filter and and the dashed line is the Kemo filter. ................................. 52
5.12 Frequency response of built 5th order Butterworth filter versus Kemo filter; the solid line is the built 5th order filter and the dashed line is the Kemo filter. ......................... 52

6.1 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with 128 filter coefficients when the ANC system is on. Dash dot line shows the attenuation with 256 filter coefficients when the ANC system is on. ................................................................. 54
6.2 Active noise control setup with 5th or 10th order filter. .................................................. 55
6.3 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 5th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. ................................................................. 56
6.4 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 5th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. ................................................................. 57
6.5 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 5th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. ................................................................. 58
6.6 Circuit scheme of 10th order Butterworth lowpass filter using Sallen-Key architecture. . . 59
LIST OF FIGURES

6.7 Simulated filter response in SPICE of 10th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees. ................................................................. 60
6.8 Real responses of built 10th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees. .................................................. 60
6.9 The solid line shows the frequency response of the 10th order filter, the dashed line shows the frequency response of the 5th order filter and the dash dotted line shows the frequency response Kemo filter. .................................................................................. 61
6.10 The solid line shows the phase response of the 10th order filter, the dashed line shows the phase response of the 5th order filter and the dash dotted line shows the phase response kemo filter. .................................................................................. 61
6.11 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. .................................................................................. 62
6.12 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. .................................................................................. 63
6.13 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on. .................................................................................. 64
6.14 Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the 5th order filters when the ANC system is on. .................................................................................. 65
6.15 Circuit scheme of 5th order Butterworth low pass filter using a Maxim 280 circuit. .. 67
6.16 Circuit scheme of two cascaded Maxim 280 Butterworth lowpass circuits. ............... 68
6.17 The solid line shows the frequency response of the maxim 280 circuit, the dashed line shows the frequency response of the 10th order filter and dash dotted line shows the frequency response of the Kemo filter. .................................................................................. 68
6.18 The solid line shows the phase response of the maxim 280 circuit, the dashed line shows the phase response of the 10th order filter and dash dotted line shows the phase response of the Kemo filter. .................................................................................. 68
6.19 Power Spectral Density (PSD) measured at the error microphone when the ANC system is on. Solid line shows the attenuation when using a 5th order filter as anti-aliasing filter on the reference microphone. Dash dot line shows the attenuation when using a 5th order Maxim filter as anti-aliasing filter on the reference microphone. .................................................................................. 69
6.20 Power Spectral Density (PSD) measured at the error microphone when the ANC system is on. Solid line shows the attenuation when using a 10th order filter as anti-aliasing filter on the reference signal. Dash dot line shows the attenuation when using a 10th order Maxim filter as anti-aliasing filter on the reference signal. .................................................................................. 70
7.1 The setup of total harmonic distortion measurements. ................................................. 74
7.2 Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using QSC USA 370 as power amplifier. Solid line shows when ANC system is off, and dashed line shows when ANC system is on. ................................................. 75
7.3 Power Spectrum measured at the microphone, $THD = 0.313\%$. ................................ 76
7.1 Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using Kemo as power amplifier. Solid line shows when ANC system is off, dashed line shows when using Kemo M034 amplifier (middle line), and dashed dot line shows when using QSC USA 370 Amplifier (lower line), when ANC system is on.  

7.5 Power Spectrum measured at the microphone, THD = 0.613%. 

7.6 Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using Sonic-T as power amplifier. Solid line shows when ANC system is off, dashed line shows when using Sonic-T amplifier, and dashed dot line shows when using QSC USA 370 Amplifier, when ANC system is on. 

7.7 Power Spectrum measured at the microphone, THD = 0.310%. 

B.1 Circuit scheme with trimmed values on 5th order filter based on Maxim 280 circuit. 

B.2 Circuit scheme with trimmed values on 10th order filter based on Maxim 280 circuit. 
List of Tables

2.1 Theoretically maximum attenuation possible with different coherence. ................. 12
3.1 Analog daughter module specification. ................................................................. 21
3.2 Kemo filter specification. ....................................................................................... 22
3.3 Settings for signal analyzer for PSD measurements. .............................................. 22
3.4 Settings for signal analyzer for FRF measurements. ............................................. 23
3.5 Settings for signal analyzer for group delay measurements. .............................. 23
3.6 Settings for signal analyzer for coherence measurements. ................................. 24

6.1 Settings for measurements in Fig. 6.1. ................................................................. 53
6.2 Settings for measurements in Fig. 6.3. ................................................................. 55
6.3 Settings for measurements in Fig. 6.4. ................................................................. 57
6.4 Settings for measurements in Fig. 6.5. ................................................................. 58
6.5 Settings for measurements in Fig. 6.11. ............................................................... 62
6.6 Settings for measurements in Fig. 6.12. ............................................................... 63
6.7 Settings for measurements in Fig. 6.13. ............................................................... 64
6.8 Settings for measurements in Fig. 6.14. ............................................................... 64

7.1 Settings for the system ............................................................................................ 73
7.2 Settings on the signal analyzer for THD measurements. ..................................... 74
7.3 QSC USA 370 specification. .................................................................................. 75
7.4 Kemo M034 specification. .................................................................................... 77
7.5 Sonic-T amp specification. ..................................................................................... 79
7.6 Summary of total harmonic distortion. ................................................................. 81
Chapter 1

Introduction

This report describes our thesis which has been performed at the Department of Signal Processing (ASB) at Blekinge Institute of Technology (BTH). The thesis is on C-level and extend of 10 points. The active noise control (ANC) system that has been used is the result of five earlier theses [1, 2, 3, 4, 5] and two conference papers [6, 7].

1.1 Purpose

The ANC system has already been developed in earlier theses and the performance is good, the next step is to exchange the laboratory equipment to cheaper and smaller products to be able to use the ANC system as a commercial product.

First the analog laboratory filters that are used as anti-aliasing and reconstruction filters should be replaced. A filter specification is given and MATLAB should be used to calculate the filter order, simulate different types of filters and finally be used to calculate the filter coefficients. Then the filters should be implemented, simulated in SPICE and finally built and be tested in the ANC system.

The power from the amplifier that is driving the loud speaker should be measured and a low cost amplifier should be bought and tested with the ANC system and be compared to two other amplifiers.

In the last part the placement of the reference microphone is tested. Measurements on how much acoustic feedback that is present should be done and also the causality of the ANC system should be tested.

1.2 Limitations

The purpose with the work presented in the thesis has not been to construct a complete product, since there is still much that can be done in the research project until a commercial product is possible. Therefore the results of this thesis are the basis for further research in this project to complete it as a commercial product.
Chapter 2

Active Noise Control

2.1 Introduction

Active noise control (ANC) is when a secondary source is used to generate a secondary sound field to cancel the primary sound field generated by the primary noise source [8]. How much of the primary noise that gets canceled is dependent on how accurate the amplitude and the phase shift are of the secondary sound field.

Paul Lueg came up with the first idea of an ANC system and got a US patent in 1936 on his feedforward control system [8]. It was not adaptive or really applicable at that time, but the general idea with using a reference microphone and speaker to cancel out a primary source was created. Most of the work until the seventies was theoretical, but when B. Widrow presented his adaptive algorithms and the digital signal processing development exploded so did the research in active noise control [9]. The filtered-X LMS (FXLMS) algorithm that B. Widrow presented in 1981 was the beginning for the practical implementation of ANC systems with adaptive filters.

An active noise control system is usually based on four main components [9]

- The plant or control path: This is the physical system between the actuator and the error.
- Sensors: This is usually a microphone or an accelerometer to pick up the reference signal and error signal.
- Actuators: The second source is often a shaker or a loud speaker.
- The controller: This controls the actuator using the signals picked up by the sensors.

This is illustrated in Fig. 2.1 and the same denotations will be used through the report.

![Illustration of ANC system](image)

Figure 2.1: Illustration of ANC system where P denotes the plant.
The ANC system is dependent either on feedforward control or feedback control. Feedforward is a control system with a reference sensor that senses the noise generated by the primary source before it passes the actuator. After the actuator another sensor (the error sensor) senses the residual noise after control. See Fig. 2.2.

![Feedforward Control Diagram](image)

**Figure 2.2: The concept of feedforward control**

Feedback control was introduced by Harry F. Olson and May in 1953 [8]. The main difference between feedback and feedforward control is that no reference sensor is used in feedback control, see Fig. 2.3. This causes the feedback system to not be as stable as the feedforward system [8].

![Feedback Control Diagram](image)

**Figure 2.3: The concept of feedback control**

ANC systems can be an effective solution when low frequency cancellation is wanted and when the passive solution is too large and/or expensive to implement [8]. ANC is most useful at low frequencies, but where can it be used and under what circumstances? ANC works best in simple one-dimensional systems, for example a duct when the wavelength is long compared to the diameter of the duct.
But ANC systems can be implemented in other applications also, here are some examples.

- **Automotive**: Cars, vans, truck and military vehicles.
- **Appliances**: Air-condition ducts, refrigerators and washing machines.
- **Industrial**: Fans, air ducts, chimneys, transformers, blower’s office cubicle partitions.
- **Transportation**: Airplanes, boats, ships, helicopters and motorcycles.
- **Headset**: Headsets used in helicopters and portable music player headphones.

Today research in ANC has for example focus on noise reduction in ventilation systems also called heating, ventilating and air-conditioning (HVAC). Mostly because it has a wide area of use in the industry and good implementation properties as mention earlier [8]. Since the prices are dropping on DSP equipment, the prices have drastically fallen on ANC systems and are definitely a product that will be more common in ventilation systems in the future.

### 2.2 Adaptive Filters

Adaptive filters are usually digital but can also be analog filters that adapt its performance based on the input signal, which can be used in a numerous of applications. The adaptive filter coefficients are updated with help of an algorithm, some of those are presented in following sections. See Fig. 2.4 for a block diagram of a standard digital adaptive filter.

![Figure 2.4: Block diagram of an adaptive filter](image)

In Fig. 2.4, $x(n)$ is the reference signal, $y(n)$ the estimated signal, $d(n)$ the desired signal, and $e(n)$ the error signal.

In designing a finite impulse response (FIR) adaptive filter, the goal is to find the filter coefficients in the FIR filter that minimize the quadric function at time $n$ [10]:

$$
\hat{\xi} = E\{e^2(n)\} = E\{(d(n) - y(n))^2\} \quad (2.1)
$$

In Eq. (2.1), $E\{\cdot\}$ is the expected value, $d(n)$ the desired response signal and $y(n)$ denotes the output signal from the adaptive filter.

A FIR Wiener filter of order $P-1$ ($P$ is no. of coefficients), give the starting equations:

$$
y(n) = \sum_{l=0}^{P-1} w(l)x(n-l) \quad (2.2)
$$
and
\[ e(n) = d(n) - \sum_{l=0}^{P-1} w_n(l)x(n-l) \]  
(2.3)

By deriving \( \xi \) of Eq. (2.1) with respect to \( w_n(l) \) and set the expression equal to zero results in
\[ E\{e(n)x(n-l)\} = 0 \]  
(2.4)

By substituting Eq. (2.3) into Eq. (2.4) and rearranging terms, gives
\[ E\{d(n)x(n-k)\} - \sum_{l=0}^{P-1} w_n(l)E\{x(n-l)x(n-k)\} = 0 \]  
(2.5)

Finally if \( x(n) \) and \( d(n) \) are jointly WSS(Wide Sens Stationary), then
\[ E\{x(n-l)x(n-k)\} = r_x(k-l) \]  
and
\[ E\{d(n)x(n-k)\} = r_{dx}(k) \]

which in the end forms the Wiener-Hopf equation
\[ \sum_{l=0}^{P-1} w_n(l)r_x(k-l) = r_{dx}(k) \]  
(2.6)

In Eq. (2.6) \( w_n \) is the optimal filter coefficients, \( r_x \) is the autocorrelation of the input signal \( x(n) \), and \( r_{dx} \) is the cross-correlation between input signal \( x(n) \) and the desired signal \( d(n) \).

Can now be written as the optimal filter coefficients
\[ w_{opt} = R_x^{-1}r_{dx} \]  
(2.7)

Eq. (2.7) expressed in matrix form
\[
\begin{pmatrix}
w_{opt}(0) \\
w_{opt}(1) \\
\vdots \\
w_{opt}(P-1)
\end{pmatrix} =
\begin{pmatrix}
r_x(0) & r_x(1) & \cdots & r_x(P-1) \\
r_x(1) & r_x(0) & \cdots & r_x(P-2) \\
\vdots & \vdots & \ddots & \vdots \\
r_x(P-1) & r_x(P-2) & \cdots & r_x(0)
\end{pmatrix}^{-1}
\begin{pmatrix}
r_{dx}(0) \\
r_{dx}(1) \\
\vdots \\
r_{dx}(P-1)
\end{pmatrix}
\]

2.2.1 Steepest Descent Adaptive Filter

Steepest descent is an iterative method that searches for a solution to minimize the quadric function Eq. (2.1). The idea is simple, let \( w_n \) be a vector estimate that minimizes the mean-square error \( \xi(n) \) at time \( n \), and at time \( (n+1) \), the last estimate of \( w_n \) is corrected by a new estimate, that is directing \( w_n \) closer to the desired solution, see Eq. (2.8).
\[ w_{n+1} = w_n - \mu \nabla \xi(n) \]  
(2.8)
In Eq. (2.8) \( \mu \) is the step size in direction of maximum descent down the quadric error surface, and \( \nabla \xi(n) \) is the gradient vector for the least square error. The algorithm works as follows:

1. Initialize \( w_0 \).
2. Calculate \( \nabla \xi(n) \).
3. Update Eq. (2.8).
4. Go back to step 2 and redo the process.

Drawbacks with this algorithm is:

- Slow convergence, since many calculations must be performed.
- Need autocorrelation matrix \( R_x(n) \) of the input signal \( x(n) \) and cross-correlation matrix \( r_{dx}(n) \) between the desired signal \( d(n) \) and the input signal \( x(n) \) to be known, which is not always the case according to [10].

### 2.3 Algorithms

#### 2.3.1 Least-Mean-Square Algorithm

The Least Mean Square (LMS) [10] algorithm is a stochastic gradient algorithm, that was invented by Bernard Widrow in the 1970, because of a problem with the deepest descent method. The problem was that in general \( d(n) \) and \( x(n) \) are unknown, and therefore since deepest descent needs the exact knowledge of the gradient vector in each direction, it could not be used. Fig. 2.5 shows a block diagram of how the adaptive filter works with the algorithm.

![LMS Algorithm Diagram](image)

**Figure 2.5:** Block diagram of LMS algorithm

In Fig. 2.5, \( x(n) \) is the reference signal, \( y(n) \) the estimated signal, \( d(n) \) the desired signal, and \( e(n) \) the error signal.

The LMS algorithm is defined by following equations [10]:

\[
y(n) = \sum_{i=0}^{P} w_i(i)x(n-i)
\]

(2.9)
where \( P + 1 \) is the filter length. The error signal value is computed by
\[
e(n) = d(n) - y(n)
\] (2.10)
and the weight vector update equation is
\[
w_{n+1} = w_n + \mu e(n)x(n)
\] (2.11)

The correct choice of the step size has a great importance of stability and convergence performance for the LMS algorithm, it should comply with Eq. (2.12) [10]
\[
0 < \mu < \frac{2}{(P + 1)E[x^2(n)]}
\] (2.12)
where \( E[x^2(n)] \) is the power in the reference signal and can be estimated as
\[
\frac{1}{N} \sum_{i=0}^{N-1} x^2(n-i)
\] (2.13)

### 2.3.2 Normalized LMS

NLMS is a normalized version of the original LMS. In unstationary processes the reference signal changes in power, which might cause LMS to become unstable. NLMS solves this problem by normalizing the step size according to Eq. (2.14). If the reference signal is large, the step size becomes small and vice versa. Eq. (2.11) is changed to [10]:
\[
w_{n+1} = w_n + \beta \frac{x(n)}{\|x(n)\|^2} e(n)
\] (2.14)
where \( \|x(n)\| \) is the \( L_2 \)-norm, that effects the step size in a negative gradient direction and reduces the sensitivity of the algorithm. The NLMS algorithm converge if the new step size variable \( \beta \) is
\[
0 < \beta < 2
\] (2.15)
If \( \|x(n)\| \) becomes a small value, the algorithm may begin to diverge. This is solved by adding a small positive value \( \varepsilon \) to Eq. (2.14)
\[
w_{n+1} = w_n + \beta \frac{x(n)}{\varepsilon + \|x(n)\|^2} e(n)
\] (2.16)
2.3.3 Filtered-X LMS

FXLMS is an algorithm based on the LMS algorithm and is used in applications, where the filter works as an active controller of a dynamic system. Using the LMS algorithm for such applications would make the system unstable due to the filtering that the forward path introduces. FXLMS compensates for the physical forward path by filtering the reference signal with an estimate of it, as described in Fig. 2.6. See page 10 for a description on how to estimate the forward path.

![Block diagram of FXLMS algorithm](image)

Figure 2.6: Block diagram of FXLMS algorithm

In Fig. 2.6, \( x(n) \) is the reference signal, \( x_c(n) \) is \( x(n) \) filtered by an estimate of the forward path \( c \), \( y(n) \) is the output signal of the adaptive filter \( w(n) \), \( y_c(n) \) is the estimated signal filtered by the forward path, \( d(n) \) is the desired signal, and \( e(n) \) is the error signal. FXLMS is defined by [8]:

\[
y(n) = \sum_{i=0}^{p} w_n(i) x(n - i) \quad (2.17)
\]

\[
e(n) = d(n) - y_c(n) \quad (2.18)
\]

\[
x_c(n) = \sum_{j=0}^{f} C^*_j x(n - j) \quad (2.19)
\]

which gives

\[
w_{n+1} = w_n + \mu e(n)x_c(n) \quad (2.20)
\]

The step size \( \mu \) can be written as

\[
0 < \mu < \frac{2}{(P + 1 + \delta)E[x_c(n)^2]} \quad (2.21)
\]

where \( \delta \) is a delay caused by the forward path.
Forward Path

An estimation model of the forward path usually called C, is needed to be done for the FXLMS algorithm. The forward path represents the physical path between the actuator input and the error sensor output. The forward path includes the delays, that the electrical equipment introduces such as the lowpass filters, analog-to-digital converters (ADC), digital-to-analog converters (DAC), amplifier and actuator. To estimate the forward path, denoted C in Fig. 2.7, a noise signal $x(n)$ is generated and excites the system through the actuator and the desired signal $d(n)$ is obtained from the error sensor.

$$y(n) = \sum_{i=0}^{P} w_i x(n - i) \quad (2.22)$$

The error signal is given by

$$e(n) = d(n) - y(n) \quad (2.23)$$

Update the filter coefficients using LMS described by Eq. (2.11) and continue doing these steps until the power of $e(n)$ becomes close to zero.

The filter $w$ in Fig. 2.7 has then converged to a model of the forward path.
2.4. Coherence

The coherence function indicates how much of the output signal $y(t)$ that is linear dependent of the signal $v(t)$ [11], see Fig. 2.8.

![Figure 2.8: Single-input-single-output system with noise at the output.](image)

The ordinary coherence is limited between $0 \leq \gamma^2(f) \leq 1$ this can also be seen in Eq. (2.24) which states the ordinary coherence function.

$$\gamma^2(f) = \frac{G_{vv}(f)}{G_{yy}(f)}$$ (2.24)

To use the coherence function in practice an estimator is needed. $G_{vv}(f)$ is the auto spectrum of the signal $v(t)$ and $G_{yy}(f)$ is the auto spectrum of the output signal $y(t)$. $G_{vv}(f)$ can also be noted as.

$$G_{vv}(f) = |H_{yx}(f)|^2G_{xx}(f)$$ (2.25)

In this derivation the $H_1$ estimator in Eq. (2.26) is used.

$$H_1(f) = \frac{G_{yx}(f)}{G_{xx}(f)}$$ (2.26)

$G_{yx}(f)$ is the cross spectrum between the output signal $y$ and the input signal $x$ and $G_{xx}(f)$ is the auto spectrum of the input signal. The $H_1$ estimator assumes there is noise at the output and the $H_2$ estimator assumes the noise is at the input. Substitute $H_{yx}(f)$ in Eq. (2.25) with the $H_1$ estimator in Eq. (2.26).

$$G_{vv}(f) = \frac{|G_{yx}(f)|^2}{G_{xx}(f)}$$ (2.27)

Finally substitute $G_{vy}(f)$ in Eq. (2.24) with Eq. (2.27) and an estimate of the ordinary coherence function $\hat{\gamma}^2(f)$ is obtained.

$$\hat{\gamma}^2_{yx} = \frac{|G_{yx}(f)|^2}{G_{xx}(f)G_{yy}(f)} = \frac{|G_{yy}(f)|^2}{G_{xx}(f)G_{yy}(f)}$$ (2.28)

The coherence function in (2.28) was derived with the $H_1$ estimator. But the coherence function is not dependent of the model of the noise which the $H_1$ and $H_2$ states, since the coherence function will be the same wherever the noise is added [11]. The coherence is important in an ANC system because the noise shall be correlated with the noise at the reference sensor and at the actuator. In other words the noise at the actuator shall be the same as at the reference sensor but delayed $\delta_t$ seconds.

If the duct is described as a linear time invariant system it could look like Fig. 2.9 where the $x(t)$ can be assumed to be the reference sensor and $y(t)$ is the error sensor. The optimal impulse response $h(t)$ should of course only be a delay but in practical systems this is often not the case. The disturbances can occur by many reasons but the most usual in a
ventilation system that provides bad coherence is different flow patterns and turbulence at the sensors. Because of this, a good setup for the sensors should be used to minimize the coherence dips. Other types of disturbance can be electrical noise from cables and equipment, resonance from the duct and disturbance from other equipments in the surroundings.

To estimate how much the coherence dip will limit the attenuation in the ANC system Eq. (2.29) could be used to calculate the theoretical maximum attenuation [8].

$$A(f) = -10 \log_{10}(1 - \gamma_{xy}^2)[dB]$$

(2.29)

See table 2.1 for examples.

<table>
<thead>
<tr>
<th>Coherence, $\hat{\gamma}_{xy}^2$</th>
<th>Maximum attenuation [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>10</td>
</tr>
<tr>
<td>0.99</td>
<td>20</td>
</tr>
<tr>
<td>0.999</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 2.1: Theoretically maximum attenuation possible with different coherence.

In Fig. 2.10 the two plots show the relationship between the ANC system performance and the coherence. The coherence is measured between the reference microphone and the error microphone in the ANC system.

This clearly shows that the coherence 'dips' results in less attenuation.
2.5 Causality

Causality in an ANC system is that the electrical delay should be shorter than the acoustical delay. To attenuate the noise, the anti-noise must be generated and have been excited to the plant before the noise has passed the attenuator. If this criterion is not fulfilled the ANC system will have lower performance broad banded noise applications. If the source is periodic or narrow banded the source can be reduced due to the periodicity and predictability of the narrowband signal [8]. Since only broad band noise will be used in this thesis the causality constraint is important and will be taken into consideration. The acoustic delay between the reference sensor and the actuator can be calculated with

\[
\delta_A = \frac{L}{v}
\]

where \( L \) is the distance in meters between the reference sensor and the actuator, \( \delta_A \) is the acoustic delay in seconds in the same distance and \( v \) is speed of sound. The electrical delay can be expressed with [8]

\[
\delta_E = \delta_W + \delta_{T1} + \delta_{T2}
\]

where \( \delta_W \) is the group delay of the adaptive filter in the (DSP) digital signal processor, \( \delta_{T1} \) is the delay of the anti aliasing filter on reference sensor and the (ADC) analog-to-digital converter, \( \delta_{T2} \) is the delay of the (DAC) digital-to-analog converter, the reconstruction filter, the amplifier and the actuator. The illustration of this can be seen in Fig. 2.11.

![Figure 2.11: Group delays in system](image)

To fulfill the causality constraints the total electrical delay \( \delta_E \) should be smaller than the acoustic delay \( \delta_A \)

\[
\delta_A \geq \delta_E
\]

Use Eq. (2.30) to calculate the smallest distance between the sensor and the actuator.

\[
L_{\text{min}} \geq v\delta_f
\]

where \( v \) is the speed of sound.
Chapter 3

The Experimental Setup

Since this thesis topic is about implementation considerations rather than the implementation itself this chapter will be a brief introduction to the experimental setup. This includes an overview over the current system, different parts, performance, settings, a brief introduction how to use the system and also some comments and thoughts about the current implementation.

3.1 Ventilation System

Fig. 3.1 shows a schematic illustration of the ventilation system setup that has been used. The reference sensor is placed approximately 6 meters from the actuator. Earlier research has concluded this to be a good placement due to causality and feedback issues. The placement of the reference sensor will be investigated in chapter 8, since in a commercial product a short propagation path is desired, as an easy and flexible implementation is needed. In other terms the planning of the system can be difficult if the propagation path is 6 meters since the ventilation systems are planned after building structures not the other way around.

![Figure 3.1: The ventilation system in the research laboratory at Blekinge Institute of Technology (BTH).](image-url)
The system is built with duct parts manufactured by Lindab AB called SR315 where 315 indicate the diameter of the duct in mm. A passive silencer has been used since earlier research has shown that it cancels standing waves and thus much better coherence is achieved [2]. The passive silencer is a sound absorbing damper called SLU100 also manufactured by Lindab AB. This passive silencer should attenuate frequencies down to approximately 300 Hz but as seen in Fig. 2.10, where the dash dotted line denotes the PSD when the ANC system is off, the passive damper do not attenuate the frequencies between 300 to 600 Hz is as good as higher frequencies. From this plot the following conclusions can be made. Even if the passive damper should attenuate approximately 40 dB over 300 Hz this is not achieved with the current system. The ANC system should be configured so it attenuates all the way up to where the passive damper fully attenuates instead of cutting at 400 Hz. Otherwise a peak will emerge between where the ANC system stop attenuate and where the passive damper attenuates, see Fig. 3.2. If the system instead had been configured to cutting at 640 Hz the peak would have been attenuated as in Fig. 2.10. The ANC system should be designed after which passive silencer that should be used and not after a static specification of 300 to 400 Hz, or choose a passive silencer after a good performing ANC system.

The fan that is used in the system is called CK315 and is a standard axial fan which has a speed of 2370 rpm and 11 fan blades. The air flow is controlled with a draught valve, but in this thesis only one constant flow of 3.2 m/s has been used. The fan is mounted in another room to minimize the disturbances of the measurements.
3.2 Sensors

The reference sensor is used to sense the sound pressure caused by the fan and the error sensor should sense the pressure after the actuator. Either some sort of microphone or an accelerometer could be used. In this system only a microphone sensor has been tested so there are no results how good an accelerometer would have worked. Due to the price of accelerometers, it was never considered as an alternative to the much cheaper microphone. Because of this the sensors in the system are microphones, so after this section when the microphones are mentioned it directly refers to the sensors. Two microphones are used according to a feedforward setup in Fig 2.2. A reference microphone is placed upstream in the duct and one error microphone directly after the actuator. See Fig. 3.1.

3.2.1 The Microphones

The microphones that are currently used in the system are cheap conductor microphones. The microphones are powered by a ICP box designed at BTH that uses two AA batteries to give an operating voltage of $3\text{V}$. See Fig. 3.3 for the circuit scheme of the ICP that is used for the microphones.

![Circuit scheme for microphone icp-boxes.](image)

In earlier theses it has been concluded that this cheap microphone works approximately as well as the expensive ones with sensitivity above $30\text{ mV/Pa}$ [3].

3.2.2 Microphone Setup

One of the usual problems when implementing an ANC in ventilation systems is the mounting of the microphones. If not a proper setup is chosen significant coherent drops will occur due to turbulent flow of noise, and as mention in section 2.4 this results in reduced attenuation. Two papers are the result of research at the department in this area [6, 7]. The latter presents the T-duct installation that is currently used in the system. See Fig. 3.4 for an illustration of the microphone setup.
CHAPTER 3. THE EXPERIMENTAL SETUP

Figure 3.4: T-duct microphone setup [7].

The installation is done in a normal T-duct from Lindab AB, a net is mounted over the vent hole and a thin layer of wadding is placed over the net. The microphone is placed on the wadding and the opening is closed to prevent air leakage.

3.3 Actuator

The actuators role in an ANC system is to transform the analog signal to mechanical pressure. Different loud speaker elements were investigated in [4] and the current element was chosen there, the element performs well but it might be replaced in the future due to price/performance issues.

3.3.1 Loud speaker element

The loud speaker element is mounted in a 90° angle to the duct, and is placed in a speaker box that fits to a T-duct. See Fig. 3.5 In [4] measurements were performed on three different elements, to compare the total harmonic distortion (THD). The result showed that a low total harmonic distortion was desirable to achieve better performance with the ANC system.

3.3.2 Amplifier

Since the controller cannot power the loud speaker an amplifier is needed. The amplifier that is used in the system is called USA 370 and manufactured by QSC. In chapter 7 it will be investigated how much power that is really needed for our system and also two other amplifiers will be tested. See table 7.3 for the specification for the amplifier.
3.4 The Controller

The controller is the core of the ANC system and in this system a digital signal processor (DSP) is used.

3.4.1 Digital Signal Processor

A DSP is a microprocessor that is optimized for numerically operations at high rates. The first DSP was produced by Intel in the late seventies and in the beginning of the eighties Texas Instrument put their first DSP on the market. It was called TMS32010 and was a 16-bit fixed-point processor. After that many other companies released their own developed DSP with architectures and instruction sets for different applications.

One difference between DSP processors is the arithmetic formats they use. There are fixed-point and floating-point processors and they both have pros and cons dependent on which application they should be used for. The fixed-point processors are normally 16-bit or 24-bit devices which stores the numbers in the same precision. This clearly limits the dynamic range of the DSP and also introduces round off errors. But the fixed-point processors are cheaper and faster than floating-point processors and are used in most commercial high-volume products such as modems, digital cameras and audio players. The floating-points processor is usually 32-bit and therefore covers a wider dynamic range. This is ideal for controller applications where a wide dynamic range is important. They are easier to program since no scaling has to be considered. But as a result of the more complex architecture than fixed-point processors they are more expensive and then also have a higher power consumption.

The DSP used in this project is a TMS320C32 which is a floating point processor from the late eighties. It has a capacity of 33.3 MHz or 16.7 millions of instructions per second (MIPS). It is mounted on a ISA board which is connected to a PC.

The speed of the DSP is important because it is directly related to how many instructions that can be performed in one period $T$. How long the period time $T$ is dependent on the sampling frequency more about this in section 3.4.2. But some important guidelines about the performance are that the processor must be able to finish all the operations during one period before the next sample is taken. That is

$$t_o + t_p < T \quad (3.1)$$

where $t_p$ is the processing time and $t_o$ is the overhead of input-output (I/O) operations [12]. According to

$$T = \frac{1}{f_s} \quad (3.2)$$

were $f_s$ is the sampling frequency, that clearly shows that the processor is strictly dependent on the sampling frequency which means the higher sampling frequency, the fewer operations can be performed.

3.4.2 Input/Output

To connect the DSP to the analog real-world an analog-to-digital converter (ADC) is needed to sample the signals on the input, on the output a digital-to-analog converter (DAC) is needed. See Fig. 3.6.

The relation between $t$ and $n$ is given by

$$t = nT \quad (3.3)$$
where \( n \) is the sample, \( t \) is the time in seconds and \( T \) is the sampling period given by Eq. (3.2) were \( f_s \) is the sampling frequency used by the ADC and the DAC. The sampling frequency need to fulfill the Nyquist rate which states the minimum sampling rate see

\[
fs = 2fm
\]  

(3.4)

Observe that Nyquist only states the minimum sampling rate for an ideal case when \( fm \) is the maximum frequency in the signal. In a control system the \( fs \) is dependent on the anti-aliasing filters and the dynamic range of the ADC, more about this later in chapter 4.

To represent the signal digitally it must be encoded into binary numbers, this is normally called quantization or encoding. The word length of the ADC decides how many values that can be represented. For example if the ADC has the word length of \( B \) bits the number of values can be calculated with \( 2^B \). To calculate the difference between the quantization levels following equation can be used \[12\],

\[
q = \frac{V_{pp}}{2^B}
\]  

(3.5)

where \( q \) is the voltage resolution and \( V_{pp} \) is the input voltage peak to peak. \( V_{pp} \) is given in the manual to I/O card, if the input range is \( \pm 3 \) volt then \( V_{pp} = 6 \) volt. If the value of \( x(n) \) is not exact on any level it will be rounded of to the nearest value. This introduces a quantization error which is the difference between the exact value and the value that it is rounded of to, and can be calculated with according to \[12\].

\[
\sigma_e^2 = \frac{q^2}{12}
\]  

(3.6)

From Eq. (3.6) it can easily be seen that a larger word length results in smaller input quantization error. The quantification error \( \sigma_e^2 \) can be compared to the input signal. This is called signal-to-quantization-noise ratio (SQNR) and can be calculated as \[12\].

\[
SQNR = 10\log_{10} \left( \frac{\sigma_x^2}{\sigma_e^2} \right)
\]  

(3.7)

To have the highest possible ratio it is important to have the power of the input signal, \( \sigma_x^2 \), as large as possible and as small quantization error , \( \sigma_e^2 \), as possible. It can be seen from Eq. (3.7) that SQNR is proportional to \( \sigma_x^2 \). To maximize the SQNR is therefore important to amplify the amplitude of the signal to \( V_{pp} \). The sensor signal \( x'(n) \) is amplified with a gain factor \( g \) as seen in following equation,

\[
x(t) = gx'(t)
\]  

(3.8)

where \( x(t) \) is the amplified input signal. The amplitude of the sensor signal \( x'(t) \) should be amplified so it matches the \( V_{pp} \) ADC value. In this system the amplification has been done manually but in a finished product there has to be some sort of active gain control (AGC) that automatically amplifies the input signal to \( V_{pp} \) of the ADC.

---

**Figure 3.6**: Conversion from analog-to-digital and digital-to-analog domain.
3.4. THE CONTROLLER

AM/D16SA Burr-Brown Analog Daughter Module

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analog Input Voltage Range</td>
<td>±3 [V]</td>
</tr>
<tr>
<td>S/N ratio</td>
<td>90 [dB]</td>
</tr>
<tr>
<td>Input Impedance</td>
<td>20.000 [Ω]</td>
</tr>
<tr>
<td>Bits on each channel</td>
<td>16</td>
</tr>
</tbody>
</table>

Table 3.1: Analog daughter module specification.

The I/O to the DSP is handled by a daughter module called AM/D16SA from Burr-Brown that is connected to the same ISA card as the DSP. See table 3.1 for specifications.

The dynamic range of the ADC can be calculated with Eq. (3.9)

\[ D \approx 6B \]  

where D is the dynamic range in dB and B is number of bits of the ADC [13]. Since the daughter module has 16 bits on each channel this gives a dynamic range of 96 dB. The dynamic range of the input channel can also be measured with the two-tone test. Two signal generators were used, a lowpass filter and the DSP card with the daughter module. One signal generator produced a sine-wave at 1000 Hz and the other signal generator one at 2000 Hz and the signals were added together. The new signal was filtered through a lowpass filter with cut off frequency of 1100 Hz to attenuate the 2000 Hz signal down approximately 86 dB. The signal was then run through the DSP with a program that sent the samples through, to see if the 2000 Hz sinusoid with the low power could be recognized after it has passed the DSP. See Fig. 3.7 for the result.

![Figure 3.7](attachment:figure.png)

Figure 3.7: (a) shows the two tones before the ADC and (b) shows the two tones after the ADC, DSP and DAC

As Fig. 3.7 shows the ADC has at least a dynamic range of 86 dB since the 2000 Hz sinusoid is detected.
CHAPTER 3. THE EXPERIMENTAL SETUP

3.5 Other Hardware

Other equipment that is also used in the system is filters and a signal analyzer.

3.5.1 Filters

The filters that are used as anti-aliasing filters and reconstruction filter is laboratory filters from Kemo Inc. See table 3.2 for filter specification.

<table>
<thead>
<tr>
<th>Kemo VBF10</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. Channels</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Gain</td>
</tr>
<tr>
<td>Input Coupling</td>
</tr>
<tr>
<td>Input Type</td>
</tr>
</tbody>
</table>

**Table 3.2:** Kemo filter specification.

The Kemo filters have an attenuation of 120dB/octave see Fig. 5.12 for frequency response. The filter theory will be discussed in chapter 4.

3.5.2 Signal Analyzer

The signal analyzer is used to do all the measurements and as a random noise generator. The signal analyzer is a Hewlett Packard 35670A. The settings for PSD measurements can be seen in table 3.3.

**Table 3.3:** Settings for signal analyzer for PSD measurements.
The settings for FRF measurements can be seen in table 3.4.

**Table 3.4: Settings for signal analyzer for FRF measurements.**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inst Mode</strong></td>
<td>FFT ANALYSIS</td>
</tr>
<tr>
<td><strong>Meas data</strong></td>
<td>FREQ RESP 2/1</td>
</tr>
<tr>
<td><strong>Trace Coord</strong></td>
<td>dB MAGNITUDE</td>
</tr>
<tr>
<td>X-AXIS LOG</td>
<td></td>
</tr>
<tr>
<td><strong>Freq</strong></td>
<td>START 0 [Hz]</td>
</tr>
<tr>
<td></td>
<td>STOP 1600 [Hz]</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>FRONT END CH* SETUP AC</td>
</tr>
<tr>
<td>XCDR UNIT CH* SETUP</td>
<td>OFF</td>
</tr>
<tr>
<td><strong>Avg</strong></td>
<td>NUMBER AVERAGES 100</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>RANDOM NOISE</td>
</tr>
<tr>
<td><strong>LEVEL</strong></td>
<td>0 dBVrms</td>
</tr>
</tbody>
</table>

The settings for group delay measurements can be seen in table 3.5.

**Table 3.5: Settings for signal analyzer for group delay measurements.**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inst Mode</strong></td>
<td>FFT ANALYSIS</td>
</tr>
<tr>
<td><strong>Meas data</strong></td>
<td>FREQ RESP 2/1</td>
</tr>
<tr>
<td><strong>Trace Coord</strong></td>
<td>GROUP DELAY</td>
</tr>
<tr>
<td><strong>Freq</strong></td>
<td>START 0 [Hz]</td>
</tr>
<tr>
<td></td>
<td>STOP 400 [Hz]</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>FRONT END CH* SETUP AC</td>
</tr>
<tr>
<td>XCDR UNIT CH* SETUP</td>
<td>OFF</td>
</tr>
<tr>
<td><strong>Avg</strong></td>
<td>NUMBER AVERAGES 100</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>RANDOM NOISE</td>
</tr>
<tr>
<td><strong>LEVEL</strong></td>
<td>1V</td>
</tr>
</tbody>
</table>
The settings for coherence measurements can be seen in table 3.6.

Signal Analyzer Settings for Coherence

<table>
<thead>
<tr>
<th>Inst Mode</th>
<th>FFT ANALYSIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>NUMBER AVERAGES 100</td>
</tr>
<tr>
<td></td>
<td>OVLDR REJ ON</td>
</tr>
<tr>
<td>Meas data</td>
<td>COHERENCE 2/1</td>
</tr>
<tr>
<td>Trace Coord</td>
<td>dB MAGNITUDE</td>
</tr>
<tr>
<td>Freq</td>
<td>START 0 [Hz]</td>
</tr>
<tr>
<td></td>
<td>STOP 400 [Hz]</td>
</tr>
<tr>
<td></td>
<td>RESOLUTN (LINES) 800</td>
</tr>
<tr>
<td>Input</td>
<td>FRONT END CH* SETUP COUPLING AC</td>
</tr>
<tr>
<td></td>
<td>XCDR UNIT CH* SETUP ON</td>
</tr>
</tbody>
</table>

**Table 3.6:** Settings for signal analyzer for coherence measurements.
3.6 Software, Code and Setup

The graphical user interface (GUI) that is used to control the DSP is called UDL and was developed at BTH in the middle of the nineties. In earlier theses complaints were made on UDL but during this project it worked well and felt robust. The code that has been used was developed in an earlier project course in adaptive signal processing and it is similar to the code used in earlier theses with some minor modifications. It is divided into two parts, forward path estimation and active noise control. They could easily have been in the same program and controlled with UDL, but since manual settings still have to be done on the system there was no point for this. The signal analyzer has been used as a random noise generator, this can also be done with the DSP and should be implemented so no reconnections have to be done.

In the following sections a short introduction is made to how the system works and how the setup looks like for forward path estimation and active noise control. The settings that are presented are according to earlier work.

3.6.1 Forward Path Estimation

To estimate the forward path the system identification method with LMS is used. See Fig. 3.8 for a flowchart of the forward path estimation code, the code can be found in appendix C.1.

![Flowchart](image)

**Figure 3.8:** Flowcharts of the forward path estimation program. (a) shows the main routine and (b) the interrupt routine.

The standard settings that have been used is a sampling frequency of 6 kHz for the ADC and DAC which then is decimated down to 1 kHz for the DSP. The frequency range that the ANC system should attenuate is 0 - 400 Hz so all the filters is setup for 400 Hz as cutoff frequency.
The setup for forward path estimation can be seen in Fig. 3.9.

The signal analyzer is used to generate random noise between 0 - 400 Hz as a reference signal, this will also be used to excite the system. The noise is amplified to the $V_{pp}$ specification of $\pm3$ volt to use the whole dynamic range of the ADC. The Kemo filters are used to amplify the signals and the levels are checked with an oscilloscope. The forward path estimate program is used to estimate the filter coefficients. When the coefficients have converged, they are exported and saved as a vector in a header file. The standard length of the forward path estimate that has been used is 128 coefficients. This is enough to make a proper model of the forward path.

### 3.6.2 Active Noise Control

In the active noise control code a minor change has been done that neither of the other theses has. When the filter coefficients are updated the unmodified code looked like this:

```c
for(cnt=0;cnt<no_coeffs;cnt++)
    w_array[cnt] = w_array[cnt] + (mu*e*xc_array[cnt]);
```

This gives $2L$ multiplications and $L$ additions every iteration where $L$ denotes the length of the filter. If the first multiplication is moved out of the loop since $e$ and $mu$ are constant during the whole loop the following loop is achieved.

```c
emu = mu*e;
for(cnt=0;cnt<no_coeffs;cnt++)
    w_array[cnt] = w_array[cnt] + (emu*xc_array[cnt]);
```

A very simple optimization and instead of $2L$ multiplications, only $L+1$ multiplications are needed. Since the DSP is only optimized for filtering operations and not multiply and add
operations because there is a lot of memory transfer which usually takes as many cycles as
the multiplication. This means that the filter coefficients update is the most time consuming
part in the active noise control program. So if the sampling frequency is increased, fewer filter
coefficients can be used. But if both a higher sampling frequency and a higher number of filter
coefficients are needed the partial-update FXLMS algorithm can be used. Instead of updating
all the coefficients in one iteration only some of them are updated. In this way a long filter still
can be used, where the only difference is that the convergence time is longer dependent on how
many blocks the filter is divided into. So if the filter is divided into 4 blocks the convergence
time is 4 times longer than if it would have been updated in one iteration. See appendix C.3
for the implementation of the partial-update FXLMS. See Fig. 3.10 for a flowchart of the active
noise control code, the code based on normalized FXLMS algorithm can be found in appendix
C.2.

![Flowchart of the active noise control program](image)

**Figure 3.10:** Flowcharts of the active noise control program. (a) shows the
main routine and (b) the interrupt routine.

The same settings on the filters and the sampling frequency should be used here as in the
forward path estimate otherwise the conditions is changed and the estimate is not an accurate
model of the forward path anymore.
The setup for active noise control can been seen in Fig. 3.11.

![Active noise control setup](image)

**Figure 3.11:** Active noise control setup.

The input signals should be amplified to ±3 volts and the header file with the forward path estimate should be included when compiling the program. The step size is achieved with try and failure technique but normally the optimal step size is around $1 \cdot 10^{-13}$ for FXLMS and for the normalized FXLMS the stepsize beta is dependent on which filter length that is used. The length of the adaptive filter is 256 coefficients. Usually the leaky FXLMS is used but in this thesis only normalized FXLMS and partial-update FXLMS is used. When the program is started first a sample from the reference microphone is taken. It is filtered with the adaptive filter $w$ and the output $y$ is put out on the output for the anti-noise loud speaker. Then the filter coefficients are updated before next samples collected. The power-spectral-density (PSD) is measured at the error microphone with the ANC system off and when it is on, so the difference easily can be seen.
Chapter 4

Analog Filters

4.1 Introduction

Analog filters are continuous-time filters that can be implemented with resistors, capacitors, amplifiers or specialized elements, etc. as passive and active filters, which have been used for a long time in electrical engineering [14]. Electronic filters exist in many different types and ways that they can be grouped into, such as lowpass, highpass, bandpass and bandstop filters. Ideal filters have identical gain at all frequencies in its passband and zero gain at all frequencies outside its passband.

See Fig. 4.1, where the dashed line shows the ideal filter in the stopband and the solid line is a normal approximation of the response in the stopband(s) and passband.

![Diagram of filter types](image)

**Figure 4.1:** Dashed line shows the ideal frequency response and the solid line shows the true frequency response, for each of the four common filter groups, (a) lowpass, (b) highpass, (c) bandpass, and (d) bandstop.

Each of these filters seen in Fig. 4.1 have their way to affect the signals in a certain frequency band. And by usage of a special filter type, as Butterworth, Chebyshev type 1 and 2, Bessel, etc, gives a certain characteristics to the filter because of the placement of poles and zeros.
4.2 Response of Filters

One usual method to measure and identify the response of an analog filter is by using the frequency response function, which is defined as the ratio between the spectrum of the output signal and the spectrum of the input signal over all frequencies. The frequency response of a filter using Fourier-transform is defined as

\[ H(j\omega) = \frac{Y(j\omega)}{X(j\omega)} \]  

(4.1)

where \( Y(j\omega) \) is the output signal from the filter, \( X(j\omega) \) is the input signal to the filter, and \( H(j\omega) \) is the quota of the input and output signals. Eq.(4.1) can be given in dB according to

\[ 20 \cdot \log(|H(j\omega)|) \]  

(4.2)

Another method to measure and identify the response of an analog filter with the ratio between the input and output signals, just as frequency response, is the transfer function. It uses the Laplace-transform instead of Fourier-transform, which is defined as

\[ H(s) = \frac{T(s)}{N(s)} \]  

(4.3)

where \( T(s) \) and \( N(s) \) are polynomials in the s-plane, which means that \( H(s) \) is a rational function in the s-plane.

The last important response of a filter is the phase response, which is the angle of the output relative to the input over all the signal frequencies. Phase response is defined as

\[ \phi(\omega) = \arg\{H(j\omega)\} = \arctan\left(\frac{H_i(j\omega)}{H_r(j\omega)}\right) \pm \pi \]  

(4.4)

where \( H_i(j\omega) \) and \( H_r(j\omega) \) are the imaginary part and the real part of \( H(j\omega) \), respectively.
4.3 Passive versus Active Filters

Analog filters can be either passive or active, which will be discussed in the following sections.

4.3.1 Passive Filters

Passive filters are built with resistors, capacitors and inductors, where resistors and capacitors forms RC circuits, inductors and capacitors forms LC circuits. The passive filters do not consume any power which makes such circuits more useful in low-power applications and is used in more specialized applications, as high frequency filter or where a large range between the background noise and maximum signal level is needed [16]. Fig. 4.2 shows two passive circuit schemes. Fig. 4.2(a) shows the RC circuit scheme for a lowpass filter and Fig. 4.2(b) shows the scheme for a highpass filter. Only the lowpass filter will be discussed below.

![RC circuit scheme for a lowpass filter](a)

![RC circuit scheme for a highpass filter](b)

Figure 4.2: Circuit schemes for passive filters, where (a) is a lowpass filter and (b) is a highpass filter. The filter input is to left and output to the right.

The capacitors reactance in the RC filter reduces as the frequency increases, when using circuit (a) in Fig. 4.2. And the reactance is 90° out of phase with the resistance. At the cutoff frequency, where the filters output is $1/\sqrt{2}$ of the input voltage (-3dB), the phase is behind with 45° in lowpass filters and as the capacitive reactance reduces, the phase goes towards 90°. The attenuation is $n \cdot 6\text{dB/octave}$, where $n$ is the filter order also known as number of poles, which is discussed further in section 4.3.4.

4.3.2 Active Filters

Active filters are built with active components as transistors and operational amplifiers, but also with use of resistors and capacitors. See Fig. 4.3 for an active filter scheme of one pole pair labelled (a) and two pole pair labelled (b). Both filter configurations are lowpass filters in Sallen-Key architecture, and using the non-inverting port as input.
Figure 4.3: Circuit schemes for active filters, where (a) is a one pole pair lowpass and (b) is a two pole pair lowpass.

The performance of the active filter can be restricted, due to the gain and frequency range limitations of the op-amps, and the signal amplitude can be limited because of the output slew rate and voltage power supply of the op-amps [16]. When using op-amps the gain in the passband can be > 1, depending on which resistor values have been put on $R_4$ and $R_3$ in Fig. 4.3. The gain is calculated as in Eq. (4.5) [14].

$$\frac{R_4 + R_3}{R_4}$$  \hspace{1cm} (4.5)

The attenuation in the stopband is $n \cdot 6dB/octave$, where $n$ is the filter order, which is number of poles, recall Fig. 4.3.
4.3. PASSIVE VERSUS ACTIVE FILTERS

4.3.3 Summary

A summary of the advantages and disadvantages using passive and active filters are written below:

**Passive Filters**

**Advantage**
- Do not consume any power.
- Have large dynamic and high frequency range.

**Disadvantage**
- Cannot generate a gain larger than one, since a purely passive circuit cannot add energy to the signal.
- Need many topologies for inductive elements, which are generally expensive and is not usually available in precise values.

**Active Filters**

**Advantage**
- Can generate a gain larger than one, according to Eq. (4.5)
- Higher order filters can easily be cascaded, since each op-amp can be second order.
- The filters are smaller in size as long as no inductors are used, which make it very useful as integrated circuits (IC)

**Disadvantage**
- The op-amps add noise and harmonic distortion to the signals.
- Low frequency range, 0.1 Hz - 50 MHz[15].
4.3.4 Poles and Zeros

Poles and zeros position in the s-plane decides the characteristics of a filter. If a filter should be able to be implemented practically, the numerator polynomial degree must be lower than or equal to the denominator polynomial degree. The numerator $z$ represents the zeros and the denominator $p$ represents the poles as

$$H(s) = \frac{N(s)}{T(s)} = G \cdot \frac{(s-z_1)(s-z_2)\cdots(s-z_K)}{(s-p_1)(s-p_2)\cdots(s-p_L)} \quad K \leq L$$  \hspace{1cm} (4.6)

where $G$ is the amplitude gain of the filter, $K$ and $L$ are the degrees of the polynomials.

In a stable filter the poles are general on or inside the unity circle in quadrant two or three on the s-plane, while the zeros can be anywhere in the s-plane. The s-plane is a two-dimensional coordinate system.

For each complex zero $z$ or pole $p$, there must be one complex conjugate for $z^*$ and $p^*$, since the numerator and denominator only can have real coefficients in the transfer function. Both complex conjugate poles, zeros and real poles, zeros can exist in the s-plane. In Fig. 4.4, the poles of a second order Butterworth filter in the s-plane are shown.

There exists no visual zeros in a Butterworth filter because of the denominator has higher degree than the numerator and the transfer function has then $(L-K)$ zeros in infinity ($s = \infty$), because of the transfer function goes towards zero. General for poles of an analog filter is [15]

$$\text{no. of poles} = \text{no. of finite zeros} + \text{no. of infinite zeros} \quad (s = \infty)$$

The G factor is constant overall "gain" value, but the magnitude and phase function can be calculated from Eq. (4.3) by changing $s$ to $j\omega$, which gives the frequency response [15]

$$H(j\omega) = G \cdot \frac{(j\omega-z_1)(j\omega-z_2)\cdots(j\omega-z_K)}{(j\omega-p_1)(j\omega-p_2)\cdots(j\omega-p_L)}$$  \hspace{1cm} (4.7)
4.3. PASSIVE VERSUS ACTIVE FILTERS

where the poles and zeros can be replaced by $a_i + jb_i$, which gives

$$(j\omega - a_i - jb_i) = -(a_i + j(\omega - b_i)) = r_i \cdot e^{j\phi_i}$$

where

$$r_i = \sqrt{a_i^2 + (\omega - b_i)^2}$$

$$\phi_i = \arctan \left( \frac{\omega - b_i}{-a_i} \right)$$

which makes Eq. (4.7) a lot simpler

$$H(j\omega) = G \cdot \frac{r_{z1}e^{j\phi_{z1}} \cdot r_{z2}e^{j\phi_{z2}} \cdots r_{zK}e^{j\phi_{zK}}}{r_{p1}e^{j\phi_{p1}} \cdot r_{p2}e^{j\phi_{p2}} \cdots r_{pL}e^{j\phi_{pL}}} \quad (4.8)$$

By adding the angles from the zeros and subtract the angles from the poles the phase can be calculated as

$$\phi(\omega) = \arg\{G\} + \phi_{z1} + \phi_{z2} + \ldots + \phi_{zK} - (\phi_{p1} + \phi_{p2} + \ldots + \phi_{pL}) \quad (4.9)$$

The magnitude function $|H(j\omega)|$, shows how the output signal amplitude changes to ratio of the input signal amplitude as a function of the frequencies. $|H(j\omega)|$ can be calculated as

$$|H(j\omega)| = |G \cdot \frac{r_{z1} \cdot r_{z2} \cdots r_{zK}}{r_{p1} \cdot r_{p2} \cdots r_{pL}}| \quad (4.10)$$
4.4 Analog Filters for Control Systems

In control systems it is important to have good filters that do not give aliasing and as small delay as possible. With a bad setting on the sampling frequency, aliasing will occur and the filters will not be able to attenuate a signal as good as hoped, and then the reconstruction of a signal could become a completely different. This is described in the following sections.

4.4.1 Anti-aliasing Filters

Aliasing arises if a signal is sampled with a sampling frequency less than \(2 \cdot f_{\text{max}}\), as described in Eq. (3.4). For simplicity consider a sinusoid of \(f_0 = 100\) Hz, that is sampled with a rate of \(f_s = 80\) samples/sec, which would give aliasing because of the Nyquist criteria in Eq. (3.4) is not fulfilled. In Fig. 4.5, the sinusoid is plotted in continuous-time domain over nine periods (solid line), and the sampled points of the sinusoid is indicated as rings (dashed line).

With the current sampling frequency of 80 samples/sec, it would remake the 100 Hz sinusoid to a 20 Hz sinusoid, which was not wanted, because of under-sampling. One easy way to avoid this problem is over sampling of a signal, which means that the sampling frequency should be

\[
f_s \gg 2 \cdot f_{\text{max}}
\]

(4.11)

To avoid aliasing, anti-aliasing filters can be used. When deciding the sampling frequency and/or the filter characteristics in a control system using analog filters, there are some things to consider, first look at Fig. 4.6.
Here \( f_1 \) is the cutoff frequency and \( f_s \) is the sampling frequency. If the measured signal has a reasonably flat spectrum up to about half the sample rate \( f_s/2 \), then the spectrum of the signal passed to the analogue-to-digital converter is given by the magnitude of the filter response itself up to \( f_s/2 \). Above half the sample frequency, the signal is aliased, with continuous-time signals at \( f_s/2 + \Delta f \) appearing in the sampled signal at \( f_s/2 - \Delta f \), so that a signal at \( f_s - f_1 \) will alias down to a frequency of \( f_1 \). The frequency components of the continuous-time signals at \( f_s/2 \) will thus appear in the sampled signal below \( f_s/2 \), with an amplitude proportional to the dashed line in Fig. 4.6 which is called the aliased characteristics [17].

When designing a filter the cutoff frequency \( f_1 \), and number of dB attenuation must be set, where the signal in a certain frequency range up to \( f_1 \) shall be digitalized and thereafter attenuated D number of dB in a certain frequency range with a constant fall off rate of R dB/octave. With Fig. 4.6 the relationships between the sampling rate \( (f_s) \) and the fall off rate R of the filter can be calculated so that the aliased part falls below the dynamic range D of the required signal. As seen, the filter must have a cut-off frequency at \( f_1 \) and have attenuated the signal at frequency \( f_s - f_1 \) by D dB. By knowing the constant fall off rate R dB/octave the filter response will fall by D dB in D/R octaves, from \( f_1 \) to \( f_s - f_1 \) which give the following equation [17]

\[
\frac{f_s - f_1}{f_1} = 2^\frac{D}{R} \quad (4.12)
\]

And for a given fall off rate R, the sampling frequency \( f_s \) can be calculated as

\[
f_s = \left(1 + 2^\frac{D}{R}\right) \cdot f_1 \quad (4.13)
\]

By knowing the sampling frequency, the fall off rate can be calculated as

\[
R = \frac{D}{\log_2 \left[\frac{(f_s - f_1)}{f_1}\right]} \quad (4.14)
\]

In an adaptive feedforward control system, there are two signals, the reference signal and error signal, which need to be converted to digital form, and have generally dynamic range requirements.
CHAPTER 4. ANALOG FILTERS

Anti-aliasing Filter for the Reference Microphone

The reference signal is used to drive the loud speaker and is filtered by the control filter. It is important to maintain the dynamic range of the reference signal within the desired frequency range of the control system, since otherwise the aliased parts will appear as sensor noise in reference signal and thereby reduce the possible attenuation. It is also important to prevent any aliased parts from being sent out by the loud speaker.

If an attenuation of A dB is required at the stopband of a filter, then the dynamic range D in Eq. (4.12), can be replaced by A, to be able to calculate the characteristics of the filter required to prevent any loss of performance, but it results in a less strict filter than required to prevent audibility of the aliased parts. To prevent the aliased parts from being audible, the filter is required to have fallen off A dB at $f_s/2$. In this case the Eq. (4.12) is [17]

$$\frac{\left(\frac{f_s}{2}\right)}{f_1} = 2^A \quad (4.15)$$

and for a given fall off rate $R$, the sampling frequency $f_s$ can be calculated as

$$f_s = \left(2^{\left(\frac{A}{R}+1\right)}\right) \cdot f_1 \quad (4.16)$$

By knowing the sampling frequency, the fall off rate can be calculated as

$$R = \frac{A}{\log_2\left(\frac{f_s}{2f_1}\right)} \quad (4.17)$$

Anti-aliasing Filter for the Error Microphone

The worst that could happen at the error sensor is when the disturbance has multiple harmonics and the sampling rate ($f_s$) is an integer multiple of the fundamental frequency $f_0$. When, or if this happens, harmonics of the disturbance will maybe appear exactly at $f_1$ and $f_s - f_1$, and those harmonics are going to be jointed so that the estimate of the harmonic amplitude at $f_1$ is affected by the aliased signal at $f_s - f_1$.

If the disturbance is instead random, then the signal at $f_s - f_1$ would be disjointed with the signal at $f_1$ and the aliasing would have no effect on the cross-correlation between the sampled reference signal and the sampled error signal, assumed that the reference signal is not aliased.

The aliasing will have the effect of adding noise to the sampled error signal, which may give unexpected behavior in the adaptive algorithm that being used. If using LMS algorithm, then the filter coefficients will effect to the increased values of random noise, because of the aliasing, and further causes the mean-square error to increase.

4.4.2 Reconstruction Filter

What is reconstruction? Reconstruction is when a time-discrete signal is converted back to continues-time. The output of a digital-to-analog converter (DAC) looks typically as a series of stair-steps, which the lowpass reconstruction filter smooths out (in frequency domain this corresponds to removing the harmonics above the cut-off frequency of the reconstruction filter) to reconstruct the analogue signal corresponding to the digital time sequence, which is illustrated in Fig. 4.7. A DAC usually has an inbuilt zero-order hold device which will be mentioned more in following text.
4.4. ANALOG FILTERS FOR CONTROL SYSTEMS

Figure 4.7: Shows the output of digital-to-analog converter (DAC) in left figure, and the smoothed out signal after the reconstruction filter in the right figure.

When considering the performance of the filter it is good to do so in the frequency domain, and when passing the sampled signal through the zero-order hold the signal is effectively convolved by an impulse response of unit amplitude and has the duration of one sample period $T$, recall Eq. (3.2). The effect of the zero-order hold in frequency domain is then the same as a linear filter, which frequency response is given by the Fourier transform of the before mentioned impulse response. It can also be mentioned that the linear phase part of the frequency response gives a delay of half a sample period ($T/2$) which will contribute to the sampled systems total delay. The complete continues-time frequency response of the zero-order hold is obtained by [17]

$$T_c(j\omega) = T \cdot \text{sinc}\left(\frac{\omega T}{2}\right) \cdot e^{(-j\omega T/2)} \quad (4.18)$$

Zero-order hold does suppress the frequency parts of the original sampled signal at frequencies around $\omega T \approx 2\pi$, which corresponds to $f \approx f_s$. Since the sampled signal spectrum is periodic, frequencies in the original spectrum around $f_s/4$, $\omega T \approx \pi/2$, will appear at $3f_s/4$, $\omega T \approx 3\pi/2$ also. Thereby is the original spectrum frequencies attenuated 1 dB, and the others 10 dB, when passing through the zero-order hold. Reconstruction filters are sometimes called anti-imaging filters, because all frequency parts above $f_s/2$, $\omega T \approx \pi$ are images of the required spectrum and should effectively be attenuated by this filter.

The specification of reconstruction filter will depend on the spectrum of the input signal to the digital-to-analog converter $u(n)$ and the degree of attenuation required in the frequency images. If these frequency images are needed to be attenuated $A$ dB, the filter must have a decay on the order of $(A - 10)$ dB, and a fall off rate of $R$ dB/octave after the cutoff frequency of $f_c = f_s/x$, where $f_s$ is the value of calculated sampling rate of anti-aliasing filters and $x$ is the over sampling variable, which give following equation

$$\frac{3f_s/4}{f_c} = 2^{(A-10)\pi} \quad (4.19)$$

then the fall off rate can be calculated as

$$R = \frac{(A-10)}{\log_2\left(\frac{f_s/4}{f_c}\right)} \quad (4.20)$$
Chapter 5

Design and Implementation

5.1 Requirements

The design of low-pass filters which should be used as anti aliasing and reconstruction filters had the following requirements:

- At least 30 dB attenuation in the stop band.
- Linear phase.

Where linear phase is needed to delay all frequencies equal in the pass band.

Analysis of Butterworth, Chebyshev type 1, and Elliptic filter types should be done, to decide which type is most suitable for implementation, depending on the system requirements.

![Mag (dB)](fpass, fstop, f(Hz))

![Mag (dB)](fpass, fstop, f(Hz))

Figure 5.1: Low pass filter requirements, (a) is for Butterworth, (b) is for Chebyshev type 1 and Elliptic.

In Fig. 5.1, \( A_{pass} \) is the 3 dB limit, \( P_{\text{ripple}} \) is the passband ripple (1 dB), \( S_{\text{ripple}} \) is the stopband ripple (1 dB), \( A_{stop} \) is the attenuation requirement in the stop band, \( f_{pass} \) is the frequency at end of pass band, set to 400 Hz, \( f_{stop} \) is the frequency at the beginning of stop band, the zone between \( f_{pass} \) and \( f_{stop} \) is the transition zone.
5.2 Filter Characteristics and Architecture

If an ideal low-pass filter existed, it would completely eliminate signals above the cutoff frequency, and perfectly pass signals below the cutoff frequency [18]. A large number of different filter types have been created from the ideal idea mentioned above, trying to get an optimum filter for different application areas.

In 1930 Stephen Butterworth presented his filter, which is one of the simplest and most common filter type today. Butterworth filters are frequently used since they are easy to implement and not due to good characteristics [15]. It is a maximally-flat-magnitude-response filter, which has maximum flatness in the passband and 3 dB attenuation at the cutoff frequency, and has about 6 dB/order attenuation above the cutoff frequency.

Chebyshev type 1 filters are designed to have a ripple in the passband. Higher ripple in the passband gives higher attenuation in the stopband, and a steeper slope above the cutoff frequency, which means higher and faster attenuation in shorter frequency range. It has 1 dB attenuation at cutoff frequency.

In 1933 Wilhelm Cauer presented a filter which today is known as both Cauer and Elliptic filter. This filter is designed to have a ripple in both passband and stopband, since it would handle filter specification more efficient. This means that an elliptic filter can have lower filter order than any other filter type and still follow the attenuation specifications of a filter.

The architecture that has been used to implement the lowpass filter is called Sallen-Key. This was chosen because of its simplicity compared to other known architectures as Multiple Feedback (MFB) and State Variable, where the latter is for precision performance. A circuit diagram for Sallen-Key is illustrated in Fig. 5.2.

![Figure 5.2: General architecture for Sallen-Key of a two-pole filter section [18].](image)

Sallen-Key architecture uses the non-inverting port as input, which normally is set for unity gain operation. This gives a very accurate unity gain filter in the passband. Pole-pair Q-value below three is specific good for Sallen-Key and only need one op-amp to build a two-pole filter section.
5.3 Design

The three filter types, Butterworth, Chebyshev type 1, and Elliptic that were described theoretically in the previous section, are going to be simulated in MATLAB in this section. The magnitude and phase function of each filter type are going to be presented in following subsections, with the requirements written in section 5.1. The result of the following analysis decides which filter type that is going to be implemented.

5.3.1 Butterworth Filter

Butterworth with its flat magnitude in the passband, and low steepness after the cutoff frequency, requires to be a 5th order, to fulfil the attenuation requirement of 30 dB, see Fig. 5.3 for filter responses.

![Butterworth Lowpass Filter](image)

**Figure 5.3:** Simulated responses of 5th order Butterworth lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees.
5.3.2 Chebyshev Type 1 Filter

Chebyshev type 1 with ripple of 1 dB in the passband and with steeper fall off rate after the cutoff frequency, then Butterworth. It requires only to be a 4th order, to be able to follow attenuation requirement of 30 dB, see Fig 5.4 for the filter response.

![Chebyshev Type 1 Lowpass Filter](image)

**Figure 5.4:** Simulated responses of 4th order Chebyshev type 1 lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees.
5.3.3 Elliptic Filter

Elliptic or Cauer filter, with 1 dB ripple in the passband and stopband. It should have the same steepness after the cutoff frequency as Chebyshev type 1. A 3rd order filter is needed to fulfil the attenuation requirement of 30 dB, see Fig. 5.5 for the filter response.

![Elliptic Filter Response](image)

**Figure 5.5:** Simulated responses of 3rd order Elliptic lowpass filter; the upper plot shows the filter response in dB and the lower plot shows the phase function in degrees.
5.3.4 Summary

As seen in previous subsections, the three filter types have different characteristics and thereby give different filter orders, Butterworth 5th order, Chebyshev type 1, 4th order, and Elliptic 3rd order. All three filters met the requirements on attenuation in the stopband, but not the linear phase requirement, as can be seen more accurate in Fig. 5.6 below.

![Figure 5.6: Phase response of Butterworth, Chebyshev type 1 and Elliptic filters; solid line is Butterworth, semi dashed line is Chebyshev type 1 and dashed line is Elliptic.](image)

The Elliptic filter does not look linear at all, Chebyshev type 1 looks somewhat linear, but Butterworth met the requirement of linear phase best, and is thereby decided to be implemented.
5.4 Implementation

A lowpass Butterworth filter of 5th order using Sallen-Key architecture is going to be used. When implementing, first the resistors and capacitors part of the Sallen-Key architecture must be calculated. The procedure of calculating those is described in the following text.

Start with writing out the five poles (2 pairs and one alone) below on the form [18]

\[ \sigma_p \pm i\omega_p \]  \hfill (5.1)

The poles can easily be given, by using function buttap in MATLAB, an example on that can be seen below.

\[
[z, p, k] = \text{buttap}(5);
\]

\[
p = -0.3090 + 0.9511i
\]

\[
-0.3090 - 0.9511i
\]

\[
-0.8090 + 0.5878i
\]

\[
-0.8090 - 0.5878i
\]

\[-1.0000
\]

which rewritten on Eq. (5.1) form is

1. \(-0.8090 \pm 0.5878i\)
2. \(-0.390 \pm 0.9511i\)
3. \(-1\)

The quadric function is

\[ N(s) = (s + \sigma_p - i\omega_p) \cdot (s + \sigma_p + i\omega_p) = s^2 + 2s\sigma_p + \sigma_p^2 + \omega_p^2 \] \hfill (5.2)

and gives following quadric functions on the five poles

1. \(s^2 + 1.618s + 1\)
2. \(s^2 + 0.618s + 1\)
3. \(s + 1\)

respective quality values denoted Q are calculated according to Eq. (5.3) for complex-pole pairs [18]

\[ Q = \frac{\sqrt{Re^2 + |Im|^2}}{2Re} \] \hfill (5.3)

where Re is the real part of the complex-pole pair, and Im is the imaginary part. The respective quality values of the complex-pole pairs and the single pole is

1. 1.618
2. 0.618
3. 1
Next a relationship between the resistors and capacitors need to be derived, which called m and n. The quality value Q can be written as [18]

$$Q = \sqrt{\frac{m \cdot n}{m+1}} \tag{5.4}$$

Refract $m \cdot n$ to one side

$$Q^2 (m+1)^2 = m \cdot n$$

The equation can be rewritten as

$$m + 2 + \frac{1}{m} = \frac{n}{Q^2} \tag{5.5}$$

Let

$$\left(\frac{n}{Q^2} - 2\right) = x \tag{5.6}$$

and now Eq. (5.5) can now be written as

$$m^2 - xm + 1 = 0 \tag{5.7}$$

With the relationship derived, the component values can be calculated.

The constant n which is the relationship between the two capacitors in the pole pair cases. It should be chosen with respect to values of the capacitors as exists to buy. And also to keep the resistor values to a few-thousand ohms according to [18]. This is why one of the roots in Eq. (5.7) is thrown away. The Q value also has a curtain influence the value of n, the higher Q the higher n should be.

Following bold numbers 1,2 and 3, corresponds to the three different Q-values, which can be seen on previous page.

1. Choose $n = 22$ and calculate $x$ according to Eq. (5.6), which give $x \approx 6.40$. The solution to Eq. (5.7) is
   - $m_1 = 6.24$
   - $m_2 = 0.16$

   where $m_1$ is thrown away. Choose a value of the first capacitor

   $$C_1 = 10nF$$

   and with the relationship [18]

   $$C_2 = C_1 \cdot n \tag{5.8}$$

   gives

   $$C_2 = C_1 \cdot n = 220nF$$

   See Fig. 5.2 for Sallen-Key architecture. $f_{pass}$ is set to 400 Hz, and the frequency scaling factor denoted FSF, is calculated as

   $$FSF = \frac{1}{\sqrt{Re^2 + |Im|^2}} \tag{5.9}$$

   where Re is the real part of the complex-pole pair, and Im is the imaginary part. FSF is $\approx 1$. From this the resistor values can be calculated

   $$R_2 = \frac{1}{2\pi \cdot C_1 \cdot FSF \cdot f_{pass} \cdot \sqrt{mn}} \tag{5.10}$$
which give \[ R_2 = 21.2k\Omega \]

and with the relationship [18]

\[ R_1 = R_2 \cdot m \]  \hspace{1cm} (5.11)

gives

\[ R_1 = R_2 \cdot m_2 = 3.4k\Omega \]

2. Choose \( n = \frac{100}{72} \) and calculate \( x \) according to Eq. (5.6), which give \( x \approx 9.90 \). The solution to Eq. (5.7) is

- \( m_1 = 9.80 \)
- \( m_2 = 0.10 \)

where \( m_1 \) is thrown away. Choose a value of the first capacitor

\[ C_1 = 22nF \]

which with Eq. (5.8) gives

\[ C_2 = C_1 \cdot n = 100nF \]

\( f_{pass} \) is set to 400 Hz, FSF is \( \approx 1 \) according to Eq. (5.9), and Eq. (5.10) gives

\[ R_2 = 25.55k\Omega \]

which with Eq. (5.11) gives

\[ R_1 = R_2 \cdot m_2 = 2.55k\Omega \]

3. This is not a pole pair, and thereby it is calculated a lot easier, see equation below [15]

\[ \frac{1}{RC} = 2\pi \cdot f_{pass} \]  \hspace{1cm} (5.12)

Choose

\[ C = 22nF \]

and refract R from Eq. (5.12), which give

\[ R = 18.1k\Omega \]

As a short summary, the calculated values for the three stages are

1. \( C_1 = 10nF, C_2 = 220nF, R_1 = 3.4k\Omega, R_2 = 21.2k\Omega. \)
2. \( C_1 = 22nF, C_2 = 100nF, R_1 = 2.55k\Omega, R_2 = 25.55k\Omega. \)
3. \( C = 22nF, R = 18.1k\Omega. \)
5.5 Testing

5.5.1 SPICE Simulation

With the component values for a 5th order Butterworth lowpass filter known, the filter is implemented in OrCad\(^1\) and simulated in SPICE. Fig. 5.7 shows the circuit scheme, but observe that the stages in previous section 5.4 are reordered with the single pole first, then lowest to highest Q-value of the two pole pairs, since it is a rule of thumb to place the pole pair with lowest Q-value first and then after increasing Q-value [15].

![Circuit scheme of 5th order Butterworth lowpass filter.](image)

The amplifiers are based on the TL081 circuit since this circuit is good for audio applications, see [19] for more information. The simulated response from SPICE of the circuit shown in Fig. 5.7 is shown in Fig. 5.8.

![Simulated filter response in SPICE of 5th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees.](image)

\(^{1}\)Newer version of old PSpice
5.5. Testing

5.5.2 Realization

The simulated response looks good and also looks familiar with the simulated response in MATLAB in section 5.3.1, and is thereby decided to be implemented in the real world. To avoid the problem that the circuits load the source voltage, a buffer unity amplifier is added before and after the lowpass filter circuits [20], see Fig. 5.9, for a circuit diagram of a buffer. As a buffer, the \( \mu A741 \) amplifier circuit has been used, see [21] for more information.

\[ \text{Figure 5.9: Circuit scheme of a unity buffer amplifier.} \]

Since the resistors and capacitors used for constructing the filters have a tolerance of error on \( \approx \pm 10\% \), trim potentiometers have been used to be able to adjust cutoff frequency \( (f_{\text{stop}}) \) to its supposed value of -3 dB. In Fig. 5.10 the measured frequency- and phase responses of the built 5th order butterworth filter are illustrated.

\[ \text{Figure 5.10: Real responses of built 5th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees.} \]
5.5.3 Summary

After having built and adjusted the 5th order Butterworth filter, the attenuation outcome of the responses of the filter works as believed.

Fig. 5.11 shows a comparison in phase between the built filter and the expensive Kemo filters, that the built filters should be able to replace, note that the latter is about 23rd order filter.

![Phase response of built 5th order Butterworth filter vs Kemo filter](image)

The phase looks still linear which is very good as required by the filter specifications. In Fig. 5.12 the frequency response is shown between the build filter and a Kemo filter.

![Frequency response of built 5th order Butterworth filter versus Kemo filter](image)

At 800 Hz (1 octave above the cutoff frequency) the built filter attenuates the signals almost 29.5 dB, while the Kemo filter attenuates the same in just one fourth octave. The performances of the built filters is discussed in chapter 6.
Chapter 6

Performance using Different Filters

In this chapter the performance when using different filters as anti-aliasing- and reconstruction filters in the ANC system are investigated and illustrated. The performance when using a higher sampling rate than 1 kHz, and using 5th or 10th order of the filters are also presented in this chapter.

6.1 Kemo VBF10M

When the ANC system was developed the Kemo filters were used because they can amplify the signal up to 70 dB and since they have such steep fall-of rate, aliasing is never a problem even when a sampling rate of approximately $2f_{\text{max}}$ (highest frequency of interest) is used. But they are lab equipment and can of course not be used in a finished product due to their price and size.

6.1.1 Performance

The setup is the same as presented in chapter 3 and can be seen in Fig. 3.11. The measurements are started after five minutes to ensure that the filter coefficients in the ANC algorithm has converged. See Fig. 6.1 for the performance of the ANC system with the Kemo filters. The system setup that has been used is presented in table 6.1.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Normalized FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>128 and 256</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>6000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>1000 [Hz]</td>
</tr>
<tr>
<td>Step size $\beta$</td>
<td>0.001 and 0.008</td>
</tr>
</tbody>
</table>

Table 6.1: Settings for measurements in Fig. 6.1.

As seen in Fig. 6.1 128 coefficient clearly gives lower attenuation since it not enough coefficients to model the propagation path of 6 meters. These measurements will be used as a reference to be compared with when the measurements on the other filters are done.
6.2 5th Order Filters

Many measurements have been done with the 5th order filters and the ones that are presented in this section are used in the last stage of the thesis work. Some of the problems with the hardware and the setup have given many different results but in the end the expected results were achieved.

One problem that was encountered was that the reconstruction filter had some distortion which was brought out through the speaker. Since the noise from the reconstruction filter was added after the DSP created the anti-noise, the lowest level of attenuation that was achieved could not go below the noise level from the reconstruction filter. This problem was with the Maxim 280 circuits and not with the filters based on the op amps. So the Maxim 280 filters were used as anti-aliasing filters on the reference and error signals where they did not add any noise. Another mistake that was done was that the filters were used before the amplification of the signal. Since the op amplifiers give some total harmonic distortion (THD) this also was amplified and corrupted the input signals which lead to lower performance. But if the filters were used after the amplification this was not a problem. These problem lead to that 10th order filters were also constructed since conclusions were made that 5th order filters were not steep enough. The Kemo filters were setup with bypass response so no filtering is performed only amplification of the signal. In other words the filters were put before the Kemo filters that were used for amplification of the input signal. Since the op amplifiers give some total harmonic distortion (THD) this also was amplified and corrupted the input signals which lead to lower performance.

The implementation of the 5th order filters were presented in chapter 5 but in the last measurements only one of those filters were used as a reconstruction filter and the two anti-aliasing filters were based on the Maxim 280 circuit. See Fig. 6.15 for the circuit scheme. The implementation of the Maxim 280 circuit will be presented in section 6.4 of this chapter.
6.2.1 Performance

The setup that is used can be seen in Fig. 6.2.

![Figure 6.2: Active noise control setup with 5th or 10th order filter.](image)

First the system is tested with a sampling frequency of 1000 Hz. The system setup that is used for this measurement are presented in table 6.2.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Normalized FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>6000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>1000 [Hz]</td>
</tr>
<tr>
<td>Stepsize $\beta$</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Table 6.2: Settings for measurements in Fig. 6.3.
CHAPTER 6. PERFORMANCE USING DIFFERENT FILTERS

See Fig. 6.3 for the result when using two max280 5th order circuits as anti-aliasing filters and one constructed 5th order as reconstruction filter, compared with the performance of the Kemo filters when using 256 filter coefficients in the algorithm.

\[ f_s = \left(2^{\left(\frac{40}{30}+1\right)}\right) \cdot 400 \]

which gives that the sampling frequency should be approximately \( f_s = 2015 \). But a higher sampling frequency for the DSP means shorter sampling period \( T \) for the DSP to work with. This means that the number of instructions that is done every iteration must be decreased. To get the system to be stable and use the normalized FXLMS algorithm the number of filter coefficients needed to be decreased down to 128 which also means that the system performs worse according to Fig. 6.1.

The system was tested with the higher sampling frequency 2142 Hz and with 128 coefficients in the adaptive filter.
See Fig. 6.4 for the result when using two max280 5th order circuits and one constructed 5th order, compared with the performance of the Kemo filters when using 128 filter coefficients in the algorithm.

![Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 5th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on.](image)

The system setup that is used for this measurement are presented in table 6.3.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Normalized FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>128</td>
</tr>
<tr>
<td>( f_s ) on ADC</td>
<td>30,000 Hz</td>
</tr>
<tr>
<td>( f_s ) on DSP</td>
<td>2142 Hz</td>
</tr>
<tr>
<td>Stepsize ( \beta )</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Table 6.3: Settings for measurements in Fig. 6.4.

As seen in Fig. 6.4 the attenuation at the higher frequencies are much better since no aliasing disturbs the signals. What also can be seen is that the performance is worse than with the Kemo filters using 128 and is therefore much worse than with 256 coefficients.

To be able to use 256 filter coefficients and have a higher sampling frequency partial-update FXLMS [8] needed to be implemented. Since no normalization is used fewer calculations were carried out every iteration. The algorithm results in that only a block of the filter is updated every iteration which means less calculation time.
The results can be seen in Fig. 6.5 for the case when using two max280 5th order circuits and one constructed 5th order, compared with the performance of the Kemo filters when using 256 filter coefficients in the algorithm.

![Figure 6.5: Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 5th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on.](image)

The settings that is used for this measurement is presented in table 6.4.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Partial-update FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>No. Blocks</td>
<td>4</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>30.000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>2142 [Hz]</td>
</tr>
<tr>
<td>Stepsize $\mu$</td>
<td>$1 \cdot 10^{-13}$</td>
</tr>
</tbody>
</table>

**Table 6.4: Settings for measurements in Fig. 6.5.**

As Fig. 6.5 shows the over-sampling prevents the aliasing and with the use of 256 filter coefficients the attenuation is as good as the reference measurement with the Kemo filters. Normally the measurements are taken five minutes after the system has started, but since four blocks were used this means that the convergence time is four times longer which means that the measurement started after 20 minutes. This is the only negative factor with partial-update FXLMS algorithm that the convergence time is longer but since the ANC system in the ventilation duct hardly is a time critical it does not matter.

### 6.2.2 Summary

As the measurement in Fig. 6.5 shows, the 5th order filter can replace the expensive Kemo filters. Some modifications had to be done in the algorithm like partial update FXLMS to be
able to use higher sampling frequency but since the only loss was convergence time it cannot be considered as a problem. The pros with using a 5th order filter is that it is cheaper than using higher order and it is also have shorter group delay. More about group delay in chapter 8.

6.3 10th Order

In earlier measurements when the performance of the 5th order filters were poor the decision were made that a higher filter order probably were needed. This should hopefully work out as a direct replacement to the Kemo filters so no changes in the algorithm were needed. The 10th order was chosen since the Maxim 280 circuit easy could be cascaded into a 10th order filter.

6.3.1 Design and Implementation

Since the filter type already is decided to be Butterworth, any simulation in MATLAB is not necessary. But simulation in SPICE is still going to be done and the result is shown in this section. In Fig. 6.6 the circuit scheme is shown for a 10th order Butterworth filter using Sallen-Key architecture.

The calculations for this filter order can be seen in appendix B.1.1. There are voltage buffers, as can be seen in Fig. 5.9, before and after the circuit in Fig. 6.6.
The SPICE simulation can be seen in Fig. 6.7, which gives an expected graph.

![Figure 6.7: Simulated filter response in SPICE of 10th order Butterworth low-pass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees.](image)

The results obtained in the SPICE simulations were good. As before the -3 dB limit was not at expected position in the frequency domain (at cutoff frequency), because of the margin of error for the resistors and capacitors was approximately ±10%. Once again trim potentiometers were used to calibrate the -3 dB limit to the wanted cutoff frequency. See Fig. 6.8 for the frequency and phase response of the filter.

![Figure 6.8: Real responses of built 10th order Butterworth lowpass filter; the upper plot shows the filter frequency response in dB and the lower plot shows the phase function in degrees.](image)
Finally the phase and frequency responses of this filter are compared with the Kemo filter and 5th order Butterworth filter, See Fig. 6.9 and Fig. 6.10.

**Figure 6.9:** The solid line shows the frequency response of the 10th order filter, the dashed line shows the frequency response of the 5th order filter and the dash dotted line shows the frequency response Kemo filter.

**Figure 6.10:** The solid line shows the phase response of the 10th order filter, the dashed line shows the phase response of the 5th order filter and the dash dotted line shows the phase response Kemo filter.
6.3.2 Performance

The setup that is used can be seen in Fig. 6.2. First the system is tested with a sampling frequency of 1000 Hz. Fig. 6.11 shows the result when using the 10th order filter. The figure also shows the result for the Kemo filters. In the experiment 256 coefficients were used in the adaptive filter.

![Figure 6.11: Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on.](image)

The system setup that are used for this measurement are presented in table 6.5.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Normalized FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>6000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>1000 [Hz]</td>
</tr>
<tr>
<td>Stepsize $\beta$</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Table 6.5: Settings for measurements in Fig. 6.11.

In Fig. 6.11 the steeper fall-of rate shows very clearly if it is compared to the 5th order graph in Fig. 6.3. The aliasing begins at 370 Hz instead of 300 Hz the graph also follows the Kemo graph and the results are at least comparable to each other. Since the 10th order filters have a steeper fall-off rate than the 5th a lower sampling frequency $f_s$ than 2142 Hz is probably needed. As in the 5th order section Eq. (4.16) is used to calculate the $f_s$. Set $A = 40$ dB, the falloff rate for the 10h order filters are approximately $R = 55$ dB and the cutoff frequency is $f_1 = 400$ Hz. This gives the following

$$ f_s = \left(2^{\frac{40}{55} + 1}\right) \cdot 400 $$

which gives the sampling frequency should be $f_s = 1324$ Hz.
Since it is already known that if the number of coefficients are reduced to 128 the performance is lower there is no reason to try this configuration. Instead the partial-update FXLMS is used directly and the sampling frequency are chosen to be 1500 Hz. Fig. 6.12 shows the result when using the 10th order filter. The figure also shows the result for the Kemo filters. In the experiment 256 coefficients were used in the adaptive filter.

The system setup that are used for this measurement is presented in table 6.6.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Partial-update FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>No. Blocks</td>
<td>2</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>30,000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>1500 [Hz]</td>
</tr>
<tr>
<td>Stepsize $\mu$</td>
<td>$1 \cdot 10^{-13}$</td>
</tr>
</tbody>
</table>

Table 6.6: Settings for measurements in Fig. 6.12.

As seen in Fig. 6.12 the attenuation is worse than a sampling frequency of 1 kHz was used and the reason for this is unknown. So a last measurement is done with the sampling frequency at 2142 Hz.
Fig. 6.13 shows the result when using the 10th order filter. The figure also shows the result for the Kemo filters. In the experiment 256 coefficients were used in the adaptive filter.

![Image of Figure 6.13: Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the Kemo filters when the ANC system is on.]

The system setup that are used for this measurement is presented in table 6.7.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Partial-update FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>No. Blocks</td>
<td>4</td>
</tr>
<tr>
<td>$f_s$ on ADC</td>
<td>30.000 [Hz]</td>
</tr>
<tr>
<td>$f_s$ on DSP</td>
<td>2142 [Hz]</td>
</tr>
<tr>
<td>Stepsize $\mu$</td>
<td>$1 \cdot 10^{-13}$</td>
</tr>
</tbody>
</table>

Table 6.7: Settings for measurements in Fig. 6.13.

As seen in Fig. 6.13 the results are similar to the 5th order filter with the same system setup.
6.3. 10TH ORDER

This can be seen in Fig. 6.14.

Figure 6.14: Power Spectral Density (PSD) measured at the error microphone. Solid line shows when the ANC system is off. Dashed line shows the attenuation with the 10th order filters when the ANC system is on. Dash dot line shows the attenuation with the 5th order filters when the ANC system is on.

6.3.3 Summary

Here the results showed that a 10th order filter was clearly a better option if no changes to the system are made. Less aliasing disturbs the signals because of the steeper fall-off rate but the 10th order filter does not do the job perfectly. The calculation showed that a sampling frequency at 1300 Hz should be enough so 1500 Hz was tried with partial-update FXLMS algorithm. The results were poor but when the sampling frequency was increased further up to 2142 Hz the results was good. So if the system setup wants to use the 1000 Hz sampling frequency the 10th order filter can work but with some aliasing present at the higher frequency. If the sampling frequency is increased the results were the same as at the 5th order filters.
CHAPTER 6. PERFORMANCE USING DIFFERENT FILTERS

6.4 Maxim 280

Since it is very time consuming to implement filters with operational amplifiers especially the 10th order which uses seven operational amplifiers and a lot of other components. The accuracy of the components value often differs with up to ±10% which means the filters have to be calibrated. To get a faster and easier implementation the Maxim 280 circuits could be used. They have a Butterworth response and only need a few extra components. Some calibration must be done but it is not as time consuming as with the filters based on operational amplifiers.

6.4.1 Implementation of Maxim 280

Every equation given in this section is taken from Maxim 280 datasheet [22]. The Maxim 280 circuit has an internal clock frequency of 140 kHz, but can also use an external. To get a maximally flat amplitude response the clock frequency should be 100 times the desired cutoff frequency. With cutoff frequency \( f_c = 400 \) Hz would then give

\[
 f_{osc} = f_c \cdot 100 = 40\text{kHz}
\]

But since the clock frequency \( f_{osc} \) can vary by ±62.5%, an external trim potentiometer could be used, which reduces the variation to ±19.5%. The clock frequency can also be calculated as

\[
 f_{osc} = 140 \cdot 10^3 \cdot \left( \frac{33 \cdot 10^{-12}}{33 \cdot 10^{-12} + C_{osc}} \right) \quad \text{[Hz]} 
\]

where 33\( \mu F \) is an internal capacitor and \( C_{osc} \) can be calculated as

\[
 C_{osc} = \left( \frac{140 \cdot 10^3}{f_{osc}} - 1 \right) \cdot 33 \cdot 10^{-12} \quad \text{[F]} 
\]

Using Eq. (6.2), the capacitor used to control the clock frequency became 82.5\( \mu F \). When using an external trim potentiometer \( C_{osc} \) should be doubled, which give a real world value of 150\( \mu F \). The trim potentiometer was recommended to be 50kΩ.

When using each Maxim 280 circuit by itself, the following Eq. (6.3)

\[
 f_c = \frac{1}{1.62} = \frac{1}{2\pi RC} 
\]

should be used to calculate appropriate values for resistor R and capacitor C on the external one pole passive filter, which is used as part of a feedback loop for the whole filter circuit. Fig. 4.2(a) shows a circuit scheme for a one pole passive lowpass filter, which is placed at the input to the maxim 280 circuit in Fig. 6.15. Since there are limited values for capacitors, it was chosen to be 0.1\( \mu F \).

From Eq. (6.3) the resistor value becomes, since \( C = 0.1\mu F \), and \( f_c = 400\text{Hz} \)

\[
 R = \frac{1.62}{f_c^2 2\pi 0.1 \cdot 10^{-6}} = 6.445\text{kΩ} 
\]
This setup can be seen in Fig. 6.15, but no SPICE simulation could be done because of limited circuit library in the computer program used.

![Figure 6.15: Circuit scheme of 5th order Butterworth low pass filter using a Maxim 280 circuit.](image)

When cascading two Maxim 280 circuits to get higher order of filters, two methods for cascading exist.

The first method uses the unbuffered output of the first circuit which directly drives the next stage input. By using this method the second circuit loads the first one which is bad. It can be corrected by making the resistor value $R_2$ of the passive pole filter 117 times bigger than $R_1$. The second method uses the buffered output on the first circuit which corrects the loading problem more easily, and both resistor values can be similar. The second method was chosen to implement the 10th order Butterworth filter using Maxim 280 circuits. To calculate the first one pole filter Eq. (6.4) should be used and Eq. (6.5) for second one pole filter.

\[
\frac{f_c}{1.59} = \frac{1}{2\pi R_1 C} \quad (6.4)
\]

\[
\frac{f_c}{1.64} = \frac{1}{2\pi R_2 C} \quad (6.5)
\]

Once again the capacitor was chosen to $0.1\mu F$, which gave

$R_1 = 6.326k\Omega$

and

$R_2 = 6.525k\Omega$
As before, no simulation in SPICE could be done because of limited library in the program, but the circuit scheme can be seen in Fig. 6.16.

Figure 6.16: Circuit scheme of two cascaded Maxim 280 Butterworth lowpass circuits.

Trim potentiometers were again used to adjust the cutoff frequency of the filter (-3 dB point). See Fig. 6.17 and 6.18 for frequency and phase responses of the maxim circuit compared to the home made 10th order filter and the Kemo filters.

Figure 6.17: The solid line shows the frequency response of the maxim 280 circuit, the dashed line shows the frequency response of the 10th order filter and dash dotted line shows the frequency response of the Kemo filter.
Figure 6.18: The solid line shows the phase response of the Maxim 280 circuit, the dashed line shows the phase response of the 10th order filter and dash dotted line shows the phase response of the Kemo filter.

6.4.2 Performance

Since only two Maxim 280 circuits were available during this thesis, one 10th order filter was built and then two 5th order filters. To compare them to the filters based on the operational amplifiers they were used as anti-aliasing filters on the reference signal. Kemo filters were used on the error signal and as a reconstruction filter.

In Fig. 6.19 and 6.20 the reference signal filter has been changed to the Maxim 280 filter to see if it makes any difference to the performance.
CHAPTER 6. PERFORMANCE USING DIFFERENT FILTERS

**Figure 6.19:** Power Spectral Density (PSD) measured at the error microphone when the ANC system is on. Solid line shows the attenuation when using a 5th order filter as anti-aliasing filter on the reference microphone. Dash dot line shows the attenuation when using a 5th order Maxim filter as anti-aliasing filter on the reference microphone.

**Figure 6.20:** Power Spectral Density (PSD) measured at the error microphone when the ANC system is on. Solid line shows the attenuation when using a 10th order filter as anti-aliasing filter on the reference signal. Dash dot line shows the attenuation when using a 10th order Maxim filter as anti-aliasing filter on the reference signal.

In these two cases Kemo filters were used as anti-aliasing filter at the error microphone signal and reconstruction filter to the loud speaker.
6.4.3 Summary

The 5th order filters perform equally but as seen in Fig. 6.20 the performance is slightly better with the 10th order Maxim circuit than with the filter based on operational amplifiers. This is probably because the Maxim circuit has better frequency response than the trimmed operational amplified filters. This can clearly be seen in Fig. 6.17.

6.5 Summary

In the first section with 5th order filters it was clearly shown that a sampling frequency that was slightly over $2f_c$ (cutoff frequency) was not enough. To use a 5th order filter it is recommended to use over sampling but then either the length of the adaptive filter must be shorter, faster DSP or partial-update FXLMS is needed. If none of these are possible a 10th order filter should be used but there is still some aliasing on the higher frequencies. However this means that the filters costs more and greater group-delay is obtained. If it is possible to use the 5th order filter it is should be used since both the price and the group-delay is halved. To this system a 5th order implementation based on the Maxim 280 with a sampling frequency of approximately $5f_c$ Hz and a partial-update FXLMS algorithm should be used. Since the only loss is convergence time when using partial-update FXLMS. The convergence time cannot be considered as a important problem compared to cost and group delay in the system.
Chapter 7

Amplifier

A power amplifier is used in the system to amplify the anti-noise signal which is fed to the anti-noise speaker. It is important that the power amplifier enhances the signal as good as possible without adding any distortions to the signal, which may alter the quality of the sound signal. Such an example on distortion is total harmonic distortion (THD).

The original power amplifier used in the system had a maximum output power of 185 W. The THD and the performance of the ANC system using the original amplifier and two others, Kemo M034 40 W and Sonic-T 18 W, were investigated. The results are presented in this chapter.

7.1 Measurements and requirements

The power needed for the anti loud speaker to function can be calculated with part of Ohms law \[20\]

\[
P = \frac{U^2}{R} \tag{7.1}
\]

where \(P\) is the power in Watt (W), \(U\) is the voltage \((V_{rms})\), and \(R\) is the resistor (impedance) in ohm (\(\Omega\)). When using the original setup as seen in Fig. 3.1 the loud speaker needs the following amount of power to produce anti noise with a sufficient amplitude

\[
P = \frac{0.1^2}{4} = 2.5mW
\]

Where \(R = 4 \Omega\) is the impedance of the loud speaker and \(U = 0.1 V_{rms}\) is measured when the ANC system was running at the output of the power amplifier. In [23] they tried attenuate an average noise level of 125 dB, and was not using more than 10 watts.

When doing the performance measurements of the different power amplifiers, the settings in table 7.1 was used.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Normalized FXLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. coefficients</td>
<td>256</td>
</tr>
<tr>
<td>(f_s) on ADC</td>
<td>6000 [Hz]</td>
</tr>
<tr>
<td>(f_s) on DSP</td>
<td>1000 [Hz]</td>
</tr>
<tr>
<td>Stepsize (\beta)</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Table 7.1: Settings for the system
CHAPTER 7. AMPLIFIER

7.1.1 Total Harmonic Distortion

Total Harmonic Distortion (THD), which is measured on the three power amplifiers mentioned before, are presented in this chapter. When audio signals travels through electric circuits, it suffers some distortion, i.e. harmonics arises.

A measure of how much an equipment distorts the signal is the total harmonic distortion (THD). The THD can be calculated by comparing the power of the harmonics to the power of the harmonics and the fundamental as [4]

\[
THD = \sqrt{\frac{H_2^2 + H_3^2 + \ldots + H_N^2}{H_1^2 + H_2^2 + H_3^2 + \ldots + H_N^2}}
\]  

(7.2)

where \(H_1^2\) is power of the fundamental frequency and \(H_2^2\) to \(H_N^2\) is the power of the harmonics.

The THD measurements presented in this chapter have been done with the setup seen in Fig. 7.1, and used the settings in table 7.2 on the signal analyzer. The sound pressure level (SPL) at the microphone has been 80 dB.

![Setup Diagram](image)

**Figure 7.1:** The setup of total harmonic distortion measurements.

In Fig. 7.1 the loud speaker is the same used in the ventilation system described in section 3.3.1, the microphone is a high precision, expensive one from PCB Piezotronics Inc., the signal analyzer is the same that is described in section 3.5.2, and the power amplifier is one of the three amplifiers named, QSC USA 370, Kemo M034 and Sonic-T.

<table>
<thead>
<tr>
<th>Signal Analyzer Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fundamental frequency</td>
</tr>
<tr>
<td>Frequency range</td>
</tr>
<tr>
<td>Frequency lines</td>
</tr>
<tr>
<td>Window</td>
</tr>
<tr>
<td>No. of harmonics</td>
</tr>
<tr>
<td>No. of averages</td>
</tr>
</tbody>
</table>

**Table 7.2:** Settings on the signal analyzer for THD measurements.

The signal analyzer used in this thesis only compares the fundamental frequency to the number of harmonics used for calculating the THD. When using seven harmonics Eq. (7.2) would become

\[
THD = \frac{\sqrt{H_2^2 + H_3^2 + H_4^2 + H_5^2 + H_6^2 + H_7^2}}{H_1}
\]
7.2 QSC USA 370

QSC is a company which produces high quality audio products. The amplifier USA 370 is powerful amplifier with 185 W at each channel with low THD. See table 7.3 for specification given from [24].

<table>
<thead>
<tr>
<th>QSC USA 370</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Power  4Ω</td>
</tr>
<tr>
<td>Power Bandwidth  5 – 60,000 [Hz]</td>
</tr>
<tr>
<td>Distortion  4Ω</td>
</tr>
<tr>
<td>Power Consumption  4.4 [Aac] @ 120V</td>
</tr>
<tr>
<td>Frequency Response  20 – 20,000 [Hz]</td>
</tr>
<tr>
<td>Input Impedance  20k [Ω] balanced, 10k [Ω] unbalanced</td>
</tr>
</tbody>
</table>

Table 7.3: QSC USA 370 specification.

7.2.1 Performance

Fig. 7.2 shows the performance when using the original setup, and using the QSC USA 370 as power amplifier.

Figure 7.2: Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using QSC USA 370 as power amplifier. Solid line shows when ANC system is off, and dashed line shows when ANC system is on.
7.2.2 Total Harmonic Distortion

Fig. 7.3 shows how much total harmonic distortion the QSC USA 370 power amplifier gives, when using the same loud speaker used in ventilation system.

The total harmonic distortion of this power amplifier and the loud speaker gives is about 0.313%.

Figure 7.3: Power Spectrum measured at the microphone, THD = 0.313%.
7.3 Kemo MO34

The brand Kemo that produces this amplifier is not the same that produces the Kemo VBF10M filters. Their products is mostly electronic kits, modules, devices and cases, in the lower price-range. The Kemo M034 is a low cost amplifier and only a potentiometer, cooling element and a power supply is needed. See table 7.4 for specifications, no THD specification was given in the datasheet [25] so this is currently unknown.

<table>
<thead>
<tr>
<th>Kemo MO34</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Power 4Ω</td>
</tr>
<tr>
<td>Frequency Response</td>
</tr>
<tr>
<td>Operating voltage</td>
</tr>
<tr>
<td>Sensitivity</td>
</tr>
</tbody>
</table>

**Table 7.4:** Kemo MO34 specification.

The circuit scheme for the amplifier can be seen in the data sheet [25].

7.3.1 Performance

Fig. 7.4 shows the performance when using the original setup, and using the Kemo M034 as power amplifier.

![Figure 7.4: Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using Kemo as power amplifier. Solid line shows when ANC system is off, dashed line shows when using Kemo M034 amplifier (middle line), and dashed dot line shows when using QSC USA 370 Amplifier (lower line), when ANC system is on.](image)

Here an obvious deterioration of the attenuation can be seen, when using the Kemo M034 compared to the QSC USA 370 power amplifier in the system.
7.3.2 Total Harmonic Distortion

Fig. 7.5 shows how much total harmonic distortion the Kemo M034 power amplifier gives, when using the same loud speaker used in ventilation system.

![Power Spectrum](image)

**Figure 7.5:** Power Spectrum measured at the microphone, $THD = 0.613\%$.

The total harmonic distortion of this power amplifier and the loud speaker is about $0.613\%$, which is almost twice as much as the original one.
7.4 Sonic-T Amp

Sonic Impact is a company that makes various electrical gadgets often very cheap. They have not developed the Sonic-T amp, they have simply made the plastic casing for a digital amplifier from Tripath called TA2024. It has very low THD and can give out 9W driven with batteries. This amplifier is very popular since it produces good audio quality at a low price. See table 7.5 for specifications given from [26].

<table>
<thead>
<tr>
<th>Sonic-T amp</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Power 4Ω</td>
<td>9 [W]</td>
</tr>
<tr>
<td>Operating voltage 8.5 - 14 [V]</td>
<td></td>
</tr>
<tr>
<td>Distortion 4Ω</td>
<td>THD 0.03% @ 1000 [Hz]</td>
</tr>
<tr>
<td>Frequency Response</td>
<td>20 – 20,000 [Hz]</td>
</tr>
</tbody>
</table>

Table 7.5: Sonic-T amp specification.

7.4.1 Performance

Fig. 7.6 shows the performance when using the original setup, and using the Sonic-T as power amplifier.

Figure 7.6: Power Spectral Density (PSD) measured at the error microphone when the ANC system is on, using Sonic-T as power amplifier. Solid line shows when ANC system is off, dashed line shows when using Sonic-T amplifier, and dashed dot line shows when using QSC USA 370 Amplifier, when ANC system is on.

Here the Sonic-T power amplifier gives the same attenuation as the QSC USA 470 power amplifier, which is used in the original setup.
7.4.2 Total Harmonic Distortion

Fig. 7.7 shows how much total harmonic distortion the Sonic-T power amplifier gives, when using the same loud speaker used in ventilation system.

![Power Spectrum](image)

**Figure 7.7:** Power Spectrum measured at the microphone, \( THD = 0.310\% \).

The total harmonic distortion of this power amplifier and the loud speaker is about 0.310\%, which is about the same as the original one.
7.5 Summary

Three power amplifiers have been tried for their performance in the ventilation system, and the results have shown that the original power amplifier QSC USA 370 and the Sonic-T power amplifier give the best performance. The QSC is a high quality and expensive product using 185 W per channel, while the Sonic-T is a cheap product with only 9 W (14 W when not using batteries) and still gives the same performance. As calculated in section 7.1 the power that is needed for the loud speaker to be able to function is about 2.5 mW, which is measured when the ANC system is on, with the QSC as power amplifier.

The Kemo M034 power amplifier performance was not really what could have been hoped for, since it gave 3 - 10 dB less attenuation over the whole frequency range, compared to the other two power amplifiers.

Why does this quite cheap 40W amplifier give worse performance? As mentioned in the beginning of this chapter, any distortion added to the sound signal may alter its quality. On each of the three power amplifiers measurements have been done to understand how much total harmonic distortion (THD) the power amplifier and the loud speaker give together. With original used power amplifier QSC USA 370 the THD is about 0.313%, which is the same as from the Sonic-T amplifier. The Kemo M034 on the other hand gives about 0.613% THD, which is twice as much as the other two, this explains why it cannot give the same performance in the system as the other two power amplifiers. See table 7.6 for measurement results.

<table>
<thead>
<tr>
<th>THD Summary</th>
<th>$f_0$ [Hz]</th>
<th>Power amplifier</th>
<th>THD [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>QSC USA 370</td>
<td>0.313</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>Kemo M034</td>
<td>0.613</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>Sonic-T</td>
<td>0.310</td>
<td></td>
</tr>
</tbody>
</table>

Table 7.6: Summary of total harmonic distortion.

As noticed above, a power amplifier and the loud speaker need to have low total harmonic distortion. THD is one thing that must be taken into consideration when choosing a future replacement for the current QSC USA 370 power amplifier and not only the power since it was shown that 9 W was enough and the measurements shown the same.
Chapter 8

Confidential
Chapter 9

Conclusions and Further Work

9.1 Conclusions

The goal of this research project is that the ANC system should become a commercial product with good performance and to a low price. The results with the filters showed that the expensive laboratory filters from Kemo could be exchanged to low cost 10th order or 5th order filters but with minor changes in the system setup, such as increased sampling frequency and partial-update algorithm. The 5th order was chosen, because of lower cost to implement, a lower group delay and no performance loss compared to the 10th order filters.

The amplifier measurements showed that the power was not an important factor, instead the THD was important. If the THD was too large the performance was decreased and this should not be an issue, since cheap low cost amplifiers with low THD are available. One example is the tripath TA2024b that is used in the Sonic-T amp, this digital amplifier gave us similar performance to the QSC USA 370 amplifier and is in the same price range as the cheap Kemo M034 amplifier.

The feedback measurements showed that no feedback was present as long as the passive damper was used and this is good since no feedback cancellation needs to be implemented in the controller.

The only constraint for a short propagation path was then the causality of the ANC system. The measurements of the equipment showed that the loud speaker introduced the longest delay and this should be investigated further. The results from the group delay were similar to the results that were achieved in the positioning of the reference microphone. The Measurements showed that approximately a propagation path of 3.15 meters needs to be used.

The results on the position of the reference microphone are a drawback since the optimal length of the propagation path should be approximately 1.3 meters which is not even near to being casual. So here a choice has to made, a lower performance and an easier ANC system implementation with a short propagation path, or a higher performance ANC system with longer propagation path which then leads to a more complicated implementation.
9.2 Further work

Before the ANC system can be a commercial product there are some things that should be investigated and evaluated. Some examples for thesis projects are:

- More amplifiers should be tested to find a low cost solution with low THD.
- Investigate if it is possible to decrease the group delay on the loud speaker.
- Automatic gain control (AGC) because the ADC converters should use the whole dynamic range so input signal should always be at the ±3V. This is currently adjusted manually by the Kemo filters but in a commercial product this must be done automatically.
- Automatic forward path estimate before the ANC system goes online and the DSP should generate the random noise. The AGC part should be done first since the input power to microphones can differ a lot. Maybe also an online forward path estimate should be implemented and tested.
- Implementation of a faster DSP so the sampling rate could be higher and the ordinary FXLMS could be used instead of partial-update FXLMS. The current DSP is a 17 year old model and new models that are faster are available for the same price as the one that is currently used.
- Different ventilation system setups, for example the bend closer to the passive dampers and more ventilation duct after the error microphones and then evaluate how this affects the performance of the ANC system.
Bibliography


Appendix A

MATLAB scripts

A.1 Analog filter design

A.1.1 Butterworth

function [n, h, f] = abuttdesign(Fpass, Fstop, A_stop, Fs)

% [n, h, f] = abuttdesign(Fpass, Fstop, A_stop, Fs)
%
% n - filter order
% h - frequency function
% f - frequency vector
% Fpass - Passband frequency [Hz]
% Fstop - Stopband frequency [Hz]
% A_stop - Stopband ripple in [dB]
% Fs - Sampling frequency [Hz]
% (c) Marcus Asteborg, Niklas Svanberg, 2006

A_pass = 3; % In dB
wp_n = (Fpass*2*pi)/Fs; % Calculates the normalized frequency
ws_n = (Fstop*2*pi)/Fs;
[n,Wn] = buttord(wp_n,ws_n,A_pass,A_stop,'s'); % Calculates the order
[b,a] = butter(n,Wn,'low','s');
[h,w] = freqs(b,a);
f = (w*Fs/(2*pi));
subplot(2,1,1), semilogx(f,20*log10(abs(h)));
title(['Butterworth Lowpass Filter',',Order = ',num2str(n)])
ylabel('Magnitude [dB]
xlabel('Frequency [Hz]
grid on
subplot(2,1,2), semilogx(f,(angle(h)*180)/pi);
ylabel(texlabel('Phase [Degrees]'))
xlabel('Frequency [Hz]
grid on
A.1.2 Chebyshev type 1

function [n, h, f] = acheby1design(Fpass, Fstop, A_pass, A_stop, Fs)

% [n, h, f] = acheby1design(Fpass, Fstop, A_pass, A_stop, Fs)
% %n  - filter order
% %h  - frequency function
% %f  - frequency vector
% %Fpass - Passband frequency [Hz]
% %Fstop - Stopband frequency [Hz]
% %A_pass - Passband ripple [dB]
% %A_stop - Stopband ripple [dB]
% %Fs  - Sampling frequency [Hz]
% (c) Marcus Asteborg, Niklas Svanberg, 2006

wp_n = (Fpass*2*pi)/Fs; % Calculates the normalized frequency
ws_n = (Fstop*2*pi)/Fs;
[n,Wp] = cheb1ord(wp_n,ws_n,A_pass,A_stop,'s'); % Calculates the order
[b,a] = cheby1(n,A_pass,Wp,'low','s');
[h,w] = freqs(b,a);
f = (w*Fs/(2*pi));
subplot(2,1,1), semilogx(f,20*log10(abs(h)));
title(['Chebychew 1 Lowpass Filter', 'Order = ', num2str(n)])
ylabel('Magnitude [dB]')
xlabel('Frequency [Hz]')
grid on;
subplot(2,1,2), semilogx(f,(angle(h)*180)/pi);
ylabel(texlabel('Phase [Degrees]'))
xlabel('Frequency [Hz]')
grid on;
A.1.3  Elliptic

function [n, h, f] = aellipdesign(Fpass, Fstop, A_pass, A_stop, Fs)

% [n, h, f] = aellipdesign(Fpass, Fstop, A_pass, A_stop, Fs)
% 
% n - filter order
% h - frequency function
% f - frequency vector
% Fpass - Passband frequency [Hz]
% Fstop - Stopband frequency [Hz]
% A_pass - Passband ripple [dB]
% A_stop - Stopband ripple [dB]
% Fs - Sampling frequency [Hz]
% (c)Marcus Asteborg, Niklas Svanberg, 2006

wp_n = (Fpass*2*pi)/Fs; %Calculates the normalized frequency
ws_n = (Fstop*2*pi)/Fs;
[n,Wp] = ellipord(wp_n,ws_n,A_pass,A_stop,'s'); %Calculates the order
[b,a] = ellip(n,A_pass,A_stop,Wp,'low','s');
[h,w] = freqs(b,a);
f = (w*Fs/(2*pi));
subplot(2,1,1), semilogx(f,20*log10(abs(h)));
title(['Elliptic Lowpass Filter', 'Order = ', num2str(n)])
ylabel('Magnitude [dB]')
xlabel('Frequency [Hz]')
grid on
subplot(2,1,2), semilogx(f,(angle(h)*180)/pi);
ylabel(texlabel('Phase [Degrees]'))
xlabel('Frequency [Hz]')
grid on
Appendix B

Filters

B.1 10th order

B.1.1 Calculations

Start with writing out the five pole pairs below on the form [18]

\[ \sigma_p \pm i \cdot \omega_p \]  \hspace{1cm} (B.1)

The poles can easily be given, by using function buttap in MATLAB, an example on that can be seen below.

\[ [z, p, k] = \text{buttap}(10); \]

\[ p \]

\[-0.1564 + 0.9877i \]
\[-0.1564 - 0.9877i \]
\[-0.4540 + 0.8910i \]
\[-0.4540 - 0.8910i \]
\[-0.7071 + 0.7071i \]
\[-0.7071 - 0.7071i \]
\[-0.8910 + 0.4540i \]
\[-0.8910 - 0.4540i \]
\[-0.9877 + 0.1564i \]
\[-0.9877 - 0.1564i \]

which rewritten in Eq. (B.1)

1. \(-0.1564 \pm i \cdot 0.9877\)
2. \(-0.4540 \pm i \cdot 0.8910\)
3. \(-0.7071 \pm i \cdot 0.7071\)
4. \(-0.8910 \pm i \cdot 0.4540\)
5. \(-0.9877 \pm i \cdot 0.1564\)

The Quadric function is

\[ N(s) = (s + \sigma_p - i \cdot \omega_p) \cdot (s + \sigma_p + i \cdot \omega_p) = s^2 + 2 \cdot s \cdot \sigma_p + \sigma_p^2 + \omega_p^2 \]

and gives following quadric function with the ten poles
1. \( s^2 + 3.20 \cdot s + 1 \)
2. \( s^2 + 1.10 \cdot s + 1 \)
3. \( s^2 + 0.7071 \cdot s + 1 \)
4. \( s^2 + 0.5612 \cdot s + 1 \)
5. \( s^2 + 0.5062 \cdot s + 1 \)

respective quality values denoted \( Q \) are calculated according to Eq. (B.2) [18]

\[
Q = \frac{\sqrt{\text{Re}^2 + |\text{Im}|^2}}{2\text{Re}}
\]  

(B.2)

where \( \text{Re} \) is the real part of the complex-pole pair, and \( \text{Im} \) is the imaginary part. The respective quality values of the complex-pole pairs is

1. 3.20
2. 1.10
3. 0.7071
4. 0.5612
5. 0.5062

The derived relationship between the resistors and capacitors done in chapter 5, is summarized as

\[
m^2 - xm + 1 = 0
\]  

(B.3)

where \( x \) is

\[
x = \left( \frac{n}{Q^2} - 2 \right)
\]  

(B.4)

The constant \( n \) which is the relationship between the two capacitors in the pole pair cases. It should be chosen with respect to values of the capacitors as exists to buy. And also to keep the resistor values to a few-thousand ohms according to [18]. This is why one of the roots in Eq. (B.3) is thrown away. The Q value also has a curtain influence the value of \( n \), the higher Q the higher \( n \) should be.

Following bold numbers 1,2,3,4 and 5, corresponds to the five different Q-values, which can be seen on previous page.

1. Choose \( n = \frac{100}{\sqrt{2}} \) and calculate \( x \) according to Eq. (B.4), which give \( x \approx 2.43 \). The solution to Eq. (B.3) is

- \( m_1 = 1.92 \)
- \( m_2 = 0.51 \)

where \( m_1 \) is thrown away. Choose a value of the first capacitor

\[ C_1 = 2.2nF \]

and with relationship [18]

\[ C_2 = C_1 \cdot n \]  

(B.5)
B.1. 10TH ORDER

\[ C_2 = C_1 \cdot n = 100nF \]

See Fig. 5.2 for Sallen-Key architecture. \( f_{pass} \) is set to 400 Hz, and the frequency scaling factor denoted FSF, is calculated as

\[ FSF = \sqrt{Re^2 + |Im|^2} \]  

(B.6)

where \( Re \) is the real part of the complex-pole pair, and \( Im \) is the imaginary part. FSF is \( \approx 1 \). From this the resistor values can be calculated

\[ R_2 = \frac{1}{2 \pi \cdot C_1 \cdot FSF \cdot f_{pass} \cdot \sqrt{mn}} \]  

(B.7)

which give

\[ R_2 = 37.51k\Omega \]

and with the relationship [18]

\[ R_1 = R_2 \cdot m \]  

(B.8)

gives

\[ R_1 = R_2 \cdot m_2 = 19.18k\Omega \]

2. Choose \( n = 22 \) and calculate \( x \) according to Eq. (B.4), which give \( x \approx 16.18 \). The solution to Eq. (B.3) is

- \( m_1 = 16.12 \)
- \( m_2 = 0.06 \)

where \( m_1 \) is thrown away. Choose a value of the first capacitor

\[ C_1 = 10nF \]

which with Eq. (B.5) gives

\[ C_2 = C_1 \cdot n = 220nF \]

\( f_{pass} \) is set to 400 Hz, FSF is \( \approx 1 \) according to Eq. (B.6), and Eq. (B.7) gives

\[ R_2 = 34.0k\Omega \]

which with Eq. (B.8) gives

\[ R_1 = R_2 \cdot m_2 = 2.11k\Omega \]

3. Choose \( n = 10 \) and calculate \( x \) according to Eq. (B.4), which give \( x \approx 18.0 \). The solution to Eq. (B.3) is

- \( m_1 = 17.95 \)
- \( m_2 = 0.06 \)

where \( m_1 \) is thrown away. Choose a value of the first capacitor

\[ C_1 = 22nF \]

which with Eq. (B.5) gives

\[ C_2 = C_1 \cdot n = 220nF \]

\[ C_2 = C_1 \cdot n = 220nF \]
\( f_{\text{pass}} \) is set to 400 Hz, FSF is \( \approx 1 \) according to Eq. (B.6), and Eq. (B.7) gives
\[
R_2 = 24.23k\Omega
\]
which with Eq. (B.8) gives
\[
R_1 = R_2 \cdot m_2 = 1.35k\Omega
\]

4. Choose \( n = \frac{100}{22} \) and calculate \( x \) according to Eq. (B.4), which gives \( x \approx 12.43 \). The solution to Eq. (B.3) is
\[
\begin{align*}
& m_1 = 12.35 \\
& m_2 = 0.08
\end{align*}
\]
where \( m_1 \) is thrown away. Choose a value of the first capacitor
\[
C_1 = 22nF
\]
which with Eq. (B.5) gives
\[
C_2 = C_1 \cdot n = 100nF
\]
\( f_{\text{pass}} \) is set to 400 Hz, FSF is \( \approx 1 \) according to Eq. (B.6), and Eq. (B.7) gives
\[
R_2 = 29.81k\Omega
\]
which with Eq. (B.8) gives
\[
R_1 = R_2 \cdot m_2 = 2.41k\Omega
\]

5. Choose \( n = \frac{100}{22} \) and calculate \( x \) according to Eq. (B.4), which gives \( x \approx 15.74 \). The solution to Eq. (B.3) is
\[
\begin{align*}
& m_1 = 15.68 \\
& m_2 = 0.06
\end{align*}
\]
where \( m_1 \) is thrown away. Choose a value of the first capacitor
\[
C_1 = 22nF
\]
which with Eq. (B.5) gives
\[
C_2 = C_1 \cdot n = 100nF
\]
\( f_{\text{pass}} \) is set to 400 Hz, FSF is \( \approx 1 \) according to Eq. (B.6), and Eq. (B.7) gives
\[
R_2 = 33.59k\Omega
\]
which with Eq. (B.8) gives
\[
R_1 = R_2 \cdot m_2 = 2.14k\Omega
\]

As a short summary, the calculated values for the five stages are
1. \( C_1 = 2.2nF, C_2 = 100nF, R_1 = 37.51k\Omega, R_2 = 19.18k\Omega \).
2. \( C_1 = 10nF, C_2 = 220nF, R_1 = 2.11k\Omega, R_2 = 34.0k\Omega \).
3. \( C_1 = 22nF, C_2 = 220nF, R_1 = 1.35k\Omega, R_2 = 24.23k\Omega \).
4. \( C_1 = 22nF, C_2 = 100nF, R_1 = 2.41k\Omega, R_2 = 29.81k\Omega \).
5. \( C_1 = 22nF, C_2 = 100nF, R_1 = 2.14k\Omega, R_2 = 33.59k\Omega \).
B.2 Maxim 280

B.2.1 Trimmed Values, 5th Order

Figure B.1: Circuit scheme with trimmed values on 5th order filter based on Maxim 280 circuit.

B.2.2 Trimmed Values, 10th Order

Figure B.2: Circuit scheme with trimmed values on 10th order filter based on Maxim 280 circuit.
Appendix C

C Code for Active Noise Control

C.1 Forward Path Estimation

/* Forward path estimate with LMS
(c) Marcus Asteborg, Niklas Svanberg, 2006 */
#include <hkrc32.h> /* Processor specific */
#include <lsiio.h> /* LSI I/O specific */
#include <ensigdsp.h>
#include <math.h>

/*Functions*/
void c_int01(); /* Interrupt routine declaration */

#define no_coeffs 128 /* No of coeffs in Forward Path */

long l_Index =0;
long l_DacOn =1;
int i_Function =0;
float f_SampleRate =6; /* Sampling Frequency in kHz */

/*Arrays*/
float c_est[no_coeffs];

/*Workspace*/
float *c_est_workspace;

/*Variables*/
float c_est_mu = 1e-13; /* Step size mu for LMS */
float x = 0; /* Input signal */
float d = 0; /* Desired signal */
float d_est = 0; /* Desired estimated signal */

/*Counters*/
int cnt;
int sample_cnt = 0;
main()
{
    ReleaseSemaphore(); /* Make sure the semaphore is released */
    SetISR(_INT0,c_int01); /* Set the interrupt service routine */
    SetDmASpace(_SPACE4); /* Set Amelia A address space */
    SetupBurrBrownDM(f_SampleRate,_SITE_A); /* Setup AMELIA A */

    for(cnt = 0 ; cnt<no_coeffs; cnt++) /*Zero c_est array */
    {
        c_est[cnt] = 0;
    }

    /* Initializes LMS workspace */
    InitLms(&c_est_workspace, no_coeffs);

    EnableInterrupts(_INT0); /* Enable interrupts and Go.. */

    for(;;)
    {
        /*----------------------------------------------------------
         // DpManager returns TRUE if data has been copied to or from the PC.
         // Use this to clear the count index after data samples have been
         // copied to the PC.
         */
        if( DpManager() )
        {
            if (i_Function==1)
            {
                i_Function = 0;
            }
        }
    }
}

/*Functions*/
void c_int01()
{
    sample_cnt++; /* Decimation Counter */

    GetAmeliaAIntStatus(); /* Read and clear the AMELIA interrupt flag */

    if(sample_cnt == 6) /*Decimate down to 1kHz*/
    {
        x = GetAmeliaACh0(); /* Read A/D data Channel A from Signal Analyzer*/
        d = GetAmeliaACh1(); /* Read A/D data Channel B from Microphone */
SetAmeliaACh0( x ); /* Copy input A to output C */

Lms1(&d_est, x, d, c_est, no_coeffs, c_est_mu, &c_est_workspace);

sample_cnt = 0;
}

ClearInterruptFlag(_INT0); /* Clear the TMS interrupt flag (IF) register */
C.2 Active Noise Control, Normalized FXLMS

/* Active noise control with normalized FXLMS 
(c) Marcus Asteborg, Niklas Svanberg, 2006 */
#include <hkrc32.h>  /* Processor specific */
#include <lsiio.h>  /* LSI I/O specific */
#include <ensigdsp.h>
#include <math.h>
#include "ForwardPath.H"  /* Forward Path float c[128] */

/*Functions*/
void c_int01();  /* Interrupt routine declaration*/
void Zero_Arrays();  /* Zero arrays */
void Beta_Control();  /* Change of Beta value */
void Reset_Control();  /* Reset Filter coefficients */
void Filter_Update();  /* Update filter coefficients */

#define no_coeffs 256  /* No. of coefficients in filter*/
#define fp_coeffs 128  /* No. of coefficients in C-array*/

long l_Index =0;
long l_DacOn =1;
int i_Function =0;
float f_SampleRate =6;  /* Sampling Frequency in kHz */

/*Arrays*/
float w_array[no_coeffs];
float xc_array[no_coeffs];
float power_array[no_coeffs];

/*Workspace*/
float *w_workspace;
float *xc_workspace;

/*Variables*/
float Beta = 0.008;  /* Beta for NLMS */
float e = 0;  /* Error signal */
float d_est = 0;  /* Desired estimated signal */
float x = 0;  /* Input signal */
float xc = 0;  /* Input signal filtered by C */
float nrm_factor = 0;  /* Normalized factor for NLMS */
float uem = 0;

/*Control Variables*/
int Beta_inc = 0;
int Beta_dec = 0;
int Reset = 0;

/*Counters*/
int cnt = 0;
int sample_cnt = 0;
int sample = 0;

main()
{
    ReleaseSemaphore();    /* Make sure the semaphore is released */
    SetISR(_INT0, c_int01); /* Set the interrupt service routine */
    SetDmASpace(_SPACE4);  /* Set Amelia A addres space */
    SetupBurrBrownDM(f_SampleRate, _SITE_A); /* Setup AMELIA A */

    Zero_Arrays();       /* Set all postions in arrays to zero */

    /* Initializes Fir workspace */
    InitFir(&w_workspace, no_coeffs);
    InitFir(&xc_workspace, no_coeffs);

    EnableInterrupts(_INT0); /* Enable interrupts and Go..*/

    for(;;)
    {
        /*-----------------------------------------------
        // DpManager returns TRUE if data has been copied to or from the PC.
        // Use this to clear the count index after data samples have been
        // copied to the PC.
        */
        if( DpManager() )
        {
            if (i_Function==1)
            {
                i_Function = 0;
            }
        }

        Beta_Control();
        Reset_Control();

        if(sample == 1) /* If new sample update filter coefficients */
        {
            Filter_Update();
            sample = 0;  /* Clear the flag */
        }
    }
}
```c
/* Functions */
void c_int01()
{
    /* Decimation Counter */
    sample_cnt++;
    /* Read and clear the AMELIA interrupt flag */
    GetAmeliaAIntStatus();

    if(sample_cnt == 6)
    {
        /* Decimate down to Samplerate / int = DSP samplerate */
        x = GetAmeliaACh0(); /* Read A/D data Channel A from Signal Analyzer (Reference) */
        Fir1(&d_est, x, w_array, no_coeffs, &w_workspace); /* Filter input signal with filter w */
        SetAmeliaACh0(-d_est); /* Copy input B to output C */
        e = GetAmeliaACh1(); /* Read A/D data Channel B from microphone (Error) */

        sample_cnt = 0; /* Clear the flag */
        sample = 1;
    }

    ClearInterruptFlag(_INT0); /* Clear the TMS interrupt flag (IF) register */
}

void Filter_Update()
{
    nrm_factor = 0;
    for(cnt=(no_coeffs-1);cnt>0;cnt--)
    {
        xc_array[cnt] = xc_array[cnt-1];
        power_array[cnt] = power_array[cnt-1];
        nrm_factor = nrm_factor + power_array[cnt];
    }

    power_array[0] = x*x;
    nrm_factor = nrm_factor + power_array[0];

    Fir1(&xc, x, c, fp_coeffs, &xc_workspace); /* Filter input signal x with C */

    xc_array[0] = xc;
    uem = (Beta*e)/nrm_factor;

    for(cnt=0;cnt<no_coeffs;cnt++)
    w_array[cnt] = w_array[cnt] + uem*xc_array[cnt];
}
```
void Zero_Arrays()
{
    for(cnt = 0 ; cnt<no_coeffs; cnt++) /* Put zeros in the arrays */
    {
        w_array[cnt] = 0;
        xc_array[cnt] = 0;
        power_array[cnt] = 0;
    }
}

void Beta_Control()
{
    if(Beta_inc == 1 )
    {
        Beta = Beta + 1e-3;
        Beta_inc = 0; /* Clear the flag */
    }
    if(Beta_dec == 1)
    {
        Beta = Beta - 1e-3;
        Beta_dec = 0; /* Clear the flag */
    }
}

void Reset_Control()
{
    if(Reset == 1)
    {
        Zero_Arrays();
        Reset = 0; /* Clear the flag */
    }
}
C.3 Active Noise Control, Partial-update FXLMS

/* Active noise control with partial-update FXLMS
(c) Marcus Asteborg, Niklas Svanberg, 2006 */

#include <hkrc32.h>  /* Processor specific */
#include <lsiio.h>    /* LSI I/O specific */
#include <ensigdsp.h>
#include <math.h>
#include "ForwardPath.H"  /* Forward Path float c[128] */

/*Functions*/
void c_int01();    /* Interrupt routine declaration*/
void Zero_Arrays(); /* Zero arrays */
void Reset_Control(); /* Reset Filter coefficients */
void Filter_Update(); /* Update filter coefficients */

#define no_coeffs 256  /* No. of coefficients in filter*/
#define fp_coeffs 128  /* No. of coefficients in C-array*/

long l_Index =0;
long l_DacOn =1;
int i_Function =0;
float f_SampleRate =30;  /* Sampling Frequency in kHz */

/*Arrays*/
float w_array[no_coeffs];
float xc_array[no_coeffs];

/*Workspace*/
float *w_workspace;
float *xc_workspace;

/*Variables*/
float e = 0;            /* Error signal */
float d_est = 0;        /* Desired estimated signal */
float x = 0;            /* Input signal */
float xc = 0;           /* Input signal filtered by C */
float mu = 1e-13;       /* mu for LMS */
float uen = 0;

/*Control Variables*/
int Reset = 0;

/*Counters*/
int cnt = 0;
int cnt1 = 0;
int sample_cnt = 0;
int sample = 0;
C.3. ACTIVE NOISE CONTROL, PARTIAL-UPDATE FXLMS

int Block = 0;
int No_Blocks = 4; /* Even Integer except 1;*/
int Blocksize = 0;

main()
{
    ReleaseSemaphore(); /* Make sure the semaphore is released */
    SetISR(_INT0, c_int01); /* Set the interrupt service routine */
    SetDmASpace(_SPACE4); /* Set Amelia A address space */
    SetupBurrBrownDM(f_SampleRate, _SITE_A); /* Setup AMELIA A */

    Zero_Arrays(); /* Set all positions in arrays to zero */

    /* Initializes Fir workspace */
    InitFir(&w_workspace, no_coeffs);
    InitFir(&xc_workspace, no_coeffs);

    EnableInterrupts(_INT0); /* Enable interrupts and Go..*/

    for(;;)
    {
        /*-----------------------------------------------
         // DpManager returns TRUE if data has been copied to or from the PC.
         // Use this to clear the count index after data samples have been
         // copied to the PC.
         */
        if( DpManager() )
        {
            if (i_Function==1)
            {
                i_Function = 0;
            }
        }

        Reset_Control();

        if(sample == 1) /* If new sample update filter coefficients */
        {
            Filter_Update();
            sample = 0; /* Clear the flag */
        }
    }
}

/*Functions*/
void c_int01()
{
    sample_cnt++; /* Decimation Counter */
    GetAmeliaAIntStatus(); /* Read and clear the AMELIA interrupt flag */

    if(sample_cnt == 14) /*Decimate down to 2.1kHz*/
    {
        x = GetAmeliaACh0(); /* Read A/D data Channel A from Signal Analyzer (Reference) */
        Fir1(&d_est, x, w_array, no_coeffs, &w_workspace); /* Filter input signal with filter w */
        SetAmeliaACh0( -d_est); /* Copy input B to output C */
        /*SetAmeliaACh1( x );*/ /* Send zero out on Output D */
        e = GetAmeliaACh1(); /* Read A/D data Channel B from microphone (Error) */
        sample_cnt = 0; /* Clear the flag */

        sample = 1;
    }

    ClearInterruptFlag(_INT0); /* Clear the TMS interrupt flag (IF) register */
}

void Filter_Update()
{
    for(cnt=(no_coeffs-1);cnt>0;cnt--)
        xc_array[cnt] = xc_array[cnt-1];

    Fir1(&xc, x, c, fp_coeffs, &xc_workspace); /* Filter input signal x with C */
    xc_array[0] = xc;
    Block++;
    Blocksize = (Block*(no_coeffs/No_Blocks));
    uen = e*mu;

    for(cnt1=cnt1;cnt1<Blocksize;cnt1++)
        w_array[cnt1] = w_array[cnt1] + (uen*xc_array[cnt1]);

    if(Block == No_Blocks)
    {
        Block = 0;
    }
C.3. ACTIVE NOISE CONTROL, PARTIAL-UPDATE FXLMS

```c

cnt1 = 0;
}

void Zero_Arrays()
{
    for(cnt = 0 ; cnt<no_coeffs; cnt++) /* Put zeros in the arrays */
    {
        w_array[cnt] = 0;
        xc_array[cnt] = 0;
    }
}

void Reset_Control()
{
    if(Reset == 1)
    {
        Zero_Arrays();
        Reset = 0; /* Clear the flag */
    }
}
```