

**EMBEDDED SIGNAL PROCESSING SYSTEMS**  
**(course 1TE682)**  
**Uppsala University – Spring 2015**  
**Report for Project 2**

Tim Josefsson  
Dept. of Information Technology  
Uppsala University  
Uppsala Sweden  
Tim.Josefsson.9673@student.uu.se

Philip Åkerfeldt  
Dept. of Information Technology  
Uppsala University  
Uppsala Sweden  
Philip.Akerfeldt.4987@student.uu.se

25th February 2015

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Filter design and properties</b>	<b>3</b>
2.1	Frequency response . . . . .	3
<b>3</b>	<b>Implementation</b>	<b>3</b>
<b>4</b>	<b>Results</b>	<b>4</b>
<b>5</b>	<b>Discussion</b>	<b>4</b>
<b>6</b>	<b>Conclusions</b>	<b>4</b>
<b>7</b>	<b>Appendix</b>	<b>4</b>

# 1 Introduction

- A short introduction to digital filters and design methods

## DIGITAL FILTERS

In the field of signal processing there are different kinds of filters that can be implemented. One common filter design is the digital filter which is a system that operates mathematically on discrete-time signals in order to either enhance or reduce attributes of the systems input signal.

Digital filter systems are usually built in a certain way. The system uses an ADC to convert the input signal so that it can be operated upon. The filter is then followed by some components that will for example for storing data for the system. Lastly is the DAC that will convert the signal into a useful output.

There are two different kinds of digital filters; Infinite Impulse Response and Finite Impulse Response which are also referred to as *IIR*-filter and *FIR*-filter respectively.

### IIR

Systems which are known as IIR systems or IIR filters are recognized for having an impulse response which, past a certain point, does not become zero but continues indefinitely.

### FIR

The FIR system, or filter, can be recognized by its impulse response which does become exactly zero at time thus making it have a finite duration.

# 2 Filter design and properties

- A detailed description of the filters (e.g., FIR or IIR filters of the different types - low pass, high pass, band pass, or the combination of them, etc.) chosen for the tasks

The sound file that was given by the project handler consisted of two types of sound contamination which were describes as noise and interference. The simplest way of handling the two types of contamination to the file was to design two separate filters for each of the them. This method was followed and it produced good results. In order to remove the noise from the given sound file we designed and implemented a FIR filter. We choose the FIR filter since that filter allows us to avoid phase shifting which leads to a smoother output signal. The interference of the sound file was handled by implementing a IIR-Notch filter and that specific type of filter was chosen because we simply wanted to handle interference of a specific frequency. A Notch filter is a band-stop filter which means that is leaves most frequencies unaltered. The specific attribute of the Notch filter is that it has a very narrow stopband and this is what gives it the ability to removes specific interfering frequencies. The choice was easy to do since the IIR-Notch filter fits our desired template.

$$H(z) = \frac{(z-z_1)(z-z_2)}{(z-p_1)(z-p_2)} = \frac{(z-e^{-j\Omega_N})(z-e^{j\Omega_N})}{(z-re^{-j\Omega_N})(z-re^{j\Omega_N})} = \frac{z^2-2z\cos\Omega_N+1}{z^2-2rz\cos\Omega_N+r^2}$$

## 2.1 Frequency response

- The frequency responses of the designed filters and their plots made in MATLAB

# 3 Implementation

- Implementation of the filters on the embedded system

## **4 Results**

- Results

## **5 Discussion**

## **6 Conclusions**

- Conclusions

## **7 Appendix**

- C codes (only your own part!) in the appendix