Concept Development for the Next Generation of Metal Detectors

Long Master Thesis, AAU
Applied Signal Processing
and Implementation (ASPI)

Group 1040
Bo Bjerrum
Klaus Dahl Kristiansen

This page is intentionally left blank.

AALBORG UNIVERSITET

Institute for Electronic Systems
Fredrik Bajers Vej 7B
Telefon 96 35 98 36
Fax 98 15 36 62
http://www.esn.aau.dk

Title:
Concept Development for the
Next Generation of Metal Detectors

Theme:
Master Thesis

Projectperiod:
P9, Fall Semester 2007
P10, Spring Semester 2008

Project group:
ASPI 07gr940/08gr1040

Members:
Bo Bjerrum
Klaus Dahl Kristiansen

Supervisors:
Anders Brødløs Olsen
Kjeld Hermansen

Copies: 10

Pages in report: 136

Appendices: 8

Turned in 4th of Juni 2008

Abstract:

This Master Thesis, at the specialisation of "Applied Signal Processing and Implementation", concerns analysis of the magnetic metal detection application, and FPGA implementation of a digital metal detector solution.

Analyses of the magnetic metal detection application, are initiated by the developing a mathematical model, based on Maxwell's equations. The model is compared to measurements, which reveals correlation between theory and practice.
The model is then used for signal analysis, for extraction of features, separating the metal from the food. It is found that different materials, have distinct amplitude and phase values. The amplitude and phase values are also shown to depend on size and frequency, which leads to the conclusion, that the frequency can be chosen *product optimal*. The results of the signal analysis, are then used for developing methods for metal detection and classification.

The methods are then analysed, and algorithmic representations are developed. The implementation of algorithms, is then performed by use of Quartus and gate level implementation in VHDL. The goal for the digital metal detector solution, is set to detect below 1 % of the metal detector head opening. Evaluation of the implemented digital metal detector reveals, that a iron sphere smaller than 0.58 % of the detector head opening can be detected. This result concludes, that the concept of a digital metal detector is applicable.

This page is intentionally left blank.

AALBORG UNIVERSITET

Institut for Elektroniske Systemer
Fredrik Bajers Vej 7B
Telefon 96 35 98 36
Fax 98 15 36 62
http://www.esn.aau.dk

Titel:
Koncept Udvikling af den Næste Generations Metal Detektorer

Tema:
Civilingeniør Speciale

Projekt periode:
P9, Efterårs Semester 2007
P10, Forårs Semester 2008

Projekt gruppe:
ASPI 07gr940/08gr1040

Medlemmer:
Bo Bjerrum
Klaus Dahl Kristiansen

Vejleder:
Anders Brødløs Olsen
Kjeld Hermansen

Kopier: 10

Sider i rapport: 136

Antal bilag: 8

Afleveret d. 4 Juni 2008

Synopsis:

Dette Civilingeniør Speciale i specialiseringen "Anvendt Signalbehandling og Implemenentering", omhandler analyse af magnetisk metal detektion, samt FPGA implementering af en digital metal detektor løsning.

Analyse af applikationen indledes ved udvikling af en matematisk model, baseret på Maxwell's ligninger. Modellen sammenholdes med målinger, som afslører korrelation mellem teori og praksis.
Modellen benyttes bl.a. til signalanalyse for uddragelse af karakteristika, som adskiller metal fra fødevarer. Det viser sig, at forskellige materialer resulterer i forskellige amplituder og faser. Amplitude og fase værdier vises at afhænge af objekt størrelse, og frekvens, hvilket leder til konklusionen, at frekvensen kan vælges *produkt optimal*. Resultatet af analysen benyttes efterfølgende til udvikling af metoder for metal detektion og klassifikation.

Metoderne bliver derefter analyset, og en algoritmisk representation optilles. Implementering af algoritmerne foretages i Quartus, samt i VHDL på gate niveau. Målet for den digitale metal detektor løsning, er at kunne detektere under 1% af spole hoved åbningen. Evaluering af systemet afslører, at en jern kugle på 0.58 % af spole hoved åbningen kan detekteres. Det konkluderes, at konceptet om en digital metal detektor er brugbart.

This page is intentionally left blank.

Preface

This report is the documentation for the 9th and 10th semester ASPI Master Thesis, concerning "Concept Development of a New Generation of Metal Detector", at the Institute of Electronics at Aalborg University (AAU). This report is prepared by group 07gr940/08gr1040, and spans from the 3rd of September 2007 to 4th of Juni 2008. The original project proposal "New Generation of Metal Detectors used in Connection with Article of Food Processing" was presented by Kjeld Hermansen, Associate Professor at AAU, together with Anders Brødløs Olsen, Assistant Professor at AAU. Kjeld Hermansen and Anders B. Olsen have also functioned as the project supervisors.

The project group would like to thank Detectronic A/S, for supplying a metal detector for analysis and data acquisition, but also for the time used on questions and meetings. Also thanks are send to Hans Ebert (KOM), Ole Kiel Jensen (KOM) and Hans Nielsen (IET) for taking time to answer questions about electromagnetic fields and material properties. Lastly the fellow students of ASPI-10 group 08gr1041, are thanked for cooperation concerning the Power Amplifier, and for lending out small signal analog electronic circuitry, built in relation with measurements on the metal detector from Detectronic A/S.

The accompanying DVD contains a copy of this report, the three-coil model implemented in MATLAB, hardware implementation in Quartus format and VHDL, and software implementation in C. Also test signals are supplied.

Bo Bjerrum

Klaus Dahl Kristiansen

This page is intentionally left blank.

Contents

Report Introduction

The application for this master thesis is metal detection in the food article industri. Two specific examples in the food industri, were given by a the company Detectronic A/S.
Metal detection is widely used in slaghterhouses, to minimize the amount of foreign metal pieces in the meat. One may wonder, why metal pieces show up in the food articles, in the first place. But the explanation is that in slaughterhouses, the butchers wear a protective glove, which is pieced together by small rings. Occasionaly, the small rings are separated from the glove, because of worness, and then they end up in the meat.
Food articles shipped to Japan, need to be completely clean. Detectronic A/S told, that once a container full of products, were shipped to Japan. In Japan they examined the products for metal using a metal detector, and found a piece of metal. The consequence was, that they shipped the whole container back to Denmark, and stopped the cooperation, which was very expensive for the producer of the products.
From these examples the motivation for metal detection is initiated. It is seen that metal detectors can reduce the expences of contaminating food articles, which is one the motivating factors for developing metal detectors.

Detectronic A/S have been in the metal detection industri since 1971, and therefore they have solid experience in making metal detectors. From the beginning they have used a three-coil metal detector head for magnetic metal detection. Another approach used, when food articles are packaged in metal foil, is x-ray, and this technology is also supported by Detectronic A/S. This master thesis concerns magnetic metal detection, which is why the focus is put on the three-coil detector head.
Based on correspondance with Detectronic A/S, the authors of this thesis have selected two problematics, focusing on the three-coil metal detector head. Two problematics are selected, based on Detectronic's practical experiences, for further analysis.

Increasing the Transmitting Frequency of the metal detector is observed to have impact on the signals, that origin from the metal detector head. Detectronic A/S believe that performance can be gained by increasing the frequency, which is based on practical experience, and on their competitors. Analysing this aspect could contribute to improved performance of the metal detector if correct, which gives the motivation to perform in-depth analysis of this problem.

Unique Phase and Amplitude for Metals is observed in the signals from the metal detector. Detectronic A/S obtained information about this, when the company bought the three-coil detector design, but have been unable to perform supporting theoretical results. The motivation

of treating this problem, is to (eventually) obtain information about how to perform detection and classification of the material, put into the detector head.

The results of analysing the two problematics are important, because they may improve the performance of the three-coil metal detector. Detectronic A/S are also interested in development of the Next Generation of Metal Detectors, based on a digital solution.

A Digital Solution for the metal detector, gives the opportunity to perform advanced computations, and therefore more advanced solutions than for the case of analog electronics, which is the technology Detectronic A/S use primarily in their present three-coil metal detectors. This problematic is in fact the overall motivating factor for doing the project in relation to this master thesis on the specialisation Applied Signal Processing and Implementation.

Detectronic A/S do not have the specific competencies in the field of signal processing, which is needed when developing a digital solution for their metal detectors. This is the reason for Søren Øgård, Managing Director at Detectronic A/S, contacting the university, to obtain *specialist competencies* for the development of the concept of a digital solution.

The two problematics, concerning the transmitting frequency, and the phase/amplitude for the specific metals, are presented in the first part of the report. The first part concerns analysing the three-coil detector head, and the results from the analyses are used for developing methods for metal detection, that can be implemented in a digital solution.
The development and implementation of the digital solution is performed in the second part of the report. The implementation concerns both algorithm development and hardware/software codesign. The second part is ended by concluding on the report, but also by taking the report into perspective. Supporting analyses, measurement reports, and other documentation, are found in the appendix.

Part I

Metal Detection System - Analysis

Introduction

The first part of three in this master thesis, is devoted the analysis of the metal detection application. The part covers modelling of the metal detector, analysis of signals here of, feature extraction, and classification methods. The introduction to the report clarified some of the aspects of metal detection, which the authors have worked with. The authors have performed detailed theoretical and practical analyses, for verifying Detectronic's practical experiences, and some of these are documented in this part. In appendices App. B and App. C, measurements are documented. In appendix App. D and App. E supporting analyses for this part of the report are found, which are referenced to in the report.

The documentation of the analysis phase is organised according to the design phase of a pattern recognition system, which is one of the main topics in [Duda et al., 2001]. The book does however assume some *a priori* knowledge, which is reflected by not concerning the "Application Analysis". The flow of this part is thereby extended to include modelling and analysis of the three-coil metal detector head. The flow of the analysis part of this master thesis, is organised according to Fig. 1.

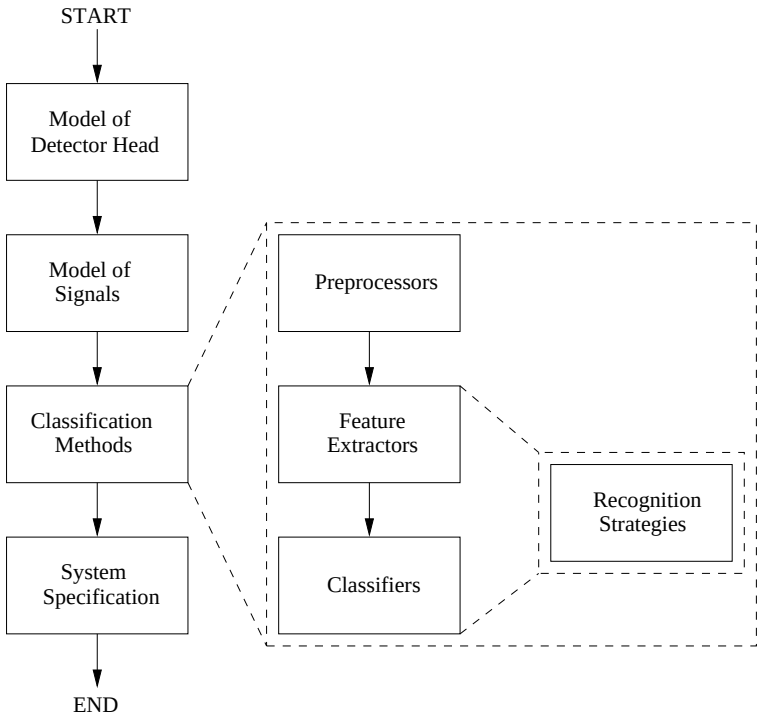

Figure 1: Organisation of flow of the analysis part of this report.

The design flow consists of four main topics covering Model of Detector Head, Model of Signals, Classification Methods, and System Specification. The four topics are related to the design of the Pattern Recognitions System each in its own unique way.
The first topic is modelling of the three-coil metal detector, which contribute *a priori* knowledge to understand the application. The topic covers a basic description of how the metal detector works, but also an in-depth mathematical derivation by use of Maxwell's equations.
The second topic is modelling of the signals, which origins from the metal detector. The topic covers analysis of signals by use of the derived mathematical model from topic one, but includes measurements obtained from a metal detector, borrowed from Detectronic A/S, for verification and discussion.
The third topic concerns the development of methods for metal detection and classification, where the methods include preprocessors, feature extractors and classifiers. When development of the feature extractors and classifiers is performed, the flow is modified, such that two Recognition Strategies are presented. This is also illustrated in Fig. 1. Both approaches are made, such that they interface the same preprocessor.
The fourth and last topic covers the system specification, which contains delimitations, and decription of how to structure the implementation phase. The topic can be viewed as covering the requirements for the prototype of a Digital Metal Detector System to be developed and implemented in Part II of the report.

The motivation, for carrying out the analysis phase, has now been discussed, and the organisation of the part is clarified. Throughout this part, Fig. 1 is used to keep track of, where the reader is in the phase. The following chapter concerns the development of a valid model for the three-coil metal detector head.

Chapter 1

Model of Detector Head Coils

This chapter presents a mathematical model for describing the physical response of the three-coil detector head, when a metal object is in the vicinity. The purpose of the model is to state the origin of the signal, from which detection of metal should be performed. From the knowledge of the origin of the signals, features which seperate metal from surrounding food articles can be extracted. This is treated in Cha. 2. In the overall design phase this is the first topic, as depicted in Fig. 1.1.

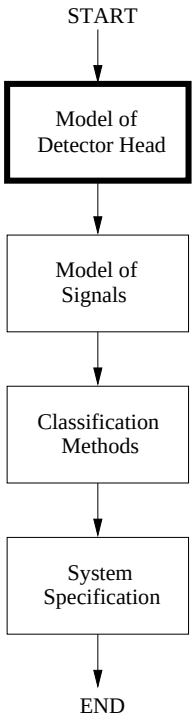

Figure 1.1: Present stage in the overall design flow is marked with a thick box. Present stage is Model of Detector Head

The following section describes Maxwell's Equations, as they are the basis for modelling the detector head. Then a basic analysis of the detector head operation is performed, followed by the development of the mathematical model for the physical response. A fully detailed mathematical derivation of the model is found in Cha. D.1. After the mathematical model is obtained, the

limitations are discussed followed a verification of the model. The verification is performed by comparing the mathematical model with a physical one, through different materials. The chapter concludes by discussing the results.

1.0.1 Maxwell's Equations

Maxwell's equations are the fundamental laws of electromagnetism, describing the connnection between various electromagnetic fields. In the differential form they are given as Ebert and Raskmark [1998],

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{1.1}$$

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \tag{1.2}$$

$$\mathbf{J} = \sigma \mathbf{E} \tag{1.3}$$

$$\mathbf{B} = \mu_0 \mu_r \mathbf{H} \tag{1.4}$$

Maxwell's four equations are briefly described seperately in the following:

$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$ describes the connection between an electric field $\mathbf{E}$ $[\frac{\mathrm{V}}{\mathrm{m}}]$, and the equivivalent magnetic field $\mathbf{B}$ $[\frac{\mathrm{T}}{\mathrm{m}}]$ in a point. It can be written in the integral form by Faraday's Law of Electromagnetic Induction, where Faraday states, that the induced electromotive force [V] in a closed loop C, equals the time rate of change of the magnetic field through the surface S, closed by the loop,

$$\oint_C \mathbf{E} \cdot d\mathrm{l} = -\frac{\partial}{\partial t} \int_S \mathbf{B} \cdot d\mathrm{s} \tag{1.5}$$

where dl is an infinitesimal length segment of the curve C and ds is an infinitesimal area element of the surface S, closed by the curve C.

$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}$ describes the connection between the magnetic field strength $\mathbf{H}$ [T], and the current density $\mathbf{J}$ $[\frac{\mathrm{A}}{\mathrm{m}^2}]$ added with the displacement current $\frac{\partial \mathbf{D}}{\partial t}$. The term $\mathbf{D}$ [V] is the electric field strength, related to $\mathbf{E}$ through $\mathbf{D} = \epsilon \mathbf{E}$, with ϵ $[\frac{\mathrm{F}}{\mathrm{m}}]$ being the permittivity of dielectric materials. More formal, Maxwell's second equation can be bescribed as the relation between the magnetic field and the associated current in a conducting wire. This is the description made by Biot-Savart, which is presented in Sec. 1.1.

$\mathbf{J} = \sigma \mathbf{E}$ describes the connection between $\mathbf{J}$, in a material of conductivity σ $[\Omega^{-1}]$, and the field $\mathbf{E}$.

$\mathbf{B} = \mu_0 \mu_r \mathbf{H}$ describes the relation between $\mathbf{B}$, in an area of permeability $\mu_0 \mu_r$ $[\frac{\mathrm{H}}{\mathrm{m}}]$, and the field $\mathbf{H}$.

These equations are essential in the description of the detector head operation in Sec. 1.1 and the derivation of the mathematical model in Sec. 1.2.

1.1 Basic Principle of Detector Head

This section describes the principle and operation of the detector head coils. In Fig. 1.2 the detector head is visualised, which consists of one transmitting coil, Tx, and two opposite wound receiving coils, Rx_1 and Rx_2.

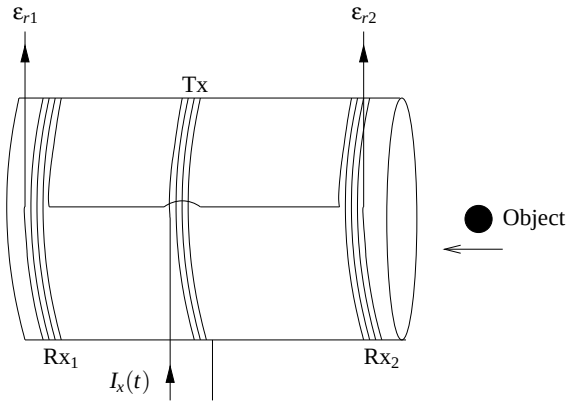

Figure 1.2: Illustration of detector head consisting of a transmitting coil Tx and two receiving coils Rx. From the right side a metallic object is moving towards the opening of the detector head.

In the Tx coil a sinusoidal current I_T is applied, which consequently generates a magnetic field $\mathbf{B}_T$, described by Biot Savart's law as,

$$\mathbf{B}_T = \frac{\mu_0 I_T}{4\pi} \int_0^{2\pi} \frac{d\mathbf{s}_i \times \mathbf{r}_i}{|\mathbf{r}_i|^2} = \frac{N_T \mu_0 I_T}{2R_T}$$

where

- μ_0 is the permeability of free space
- $d\mathbf{s}$ is an infinitesimal length vector in direction of I_T
- $\mathbf{r}$ is a vector from point on coil to point of measure, which is here the center of the coil
- N_T is the number of turns in the transmitting coil
- R_T is the radius of the transmitting coil

If, for simplicity, the propagation of the field $\mathbf{B}_T$ in the transmitting coil is assumed linear, it can be illustrated as in figure Fig. 1.3.
The field $\mathbf{B}_T$ induces an electromotive force, ε_R, in each of the opposite wound receiving coils, described by Faraday's law as,

$$\varepsilon = N_R \oint_C \mathbf{E}_R \cdot d\mathbf{l} = -N_R \frac{\partial}{\partial t} \int_S \mathbf{B}_T \cdot d\mathbf{s} = -N_R \frac{\partial \mathbf{B}_T s_R}{\partial t}$$

where

- $\mathbf{E}_R$ is the electric field in a wire of contour C
- $\mathbf{B}_T$ is the magnetic field through the surface S closed by C

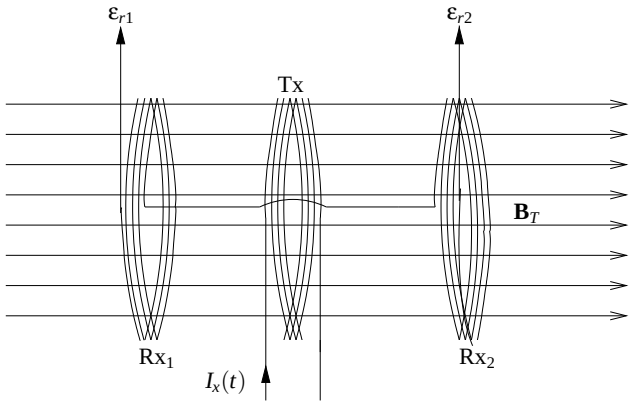

Figure 1.3: Magnetic field $\mathbf{B}_T$ because of current I_T in the transmitting coil. In the receiving coils Rx_1 and Rx_2 an electromotive force ε_R is induced.

- N_R is the number of turns in the Rx coils
- s_R is the area of the surface S, corresponding to the area of the Rx coils
- the negative sign is a consequence of Lenz' law

Because the two Rx coils are wound opposite of each other, the resultant force (or voltage), ε_R, measured differentially across the receiver terminals becomes,

$$\varepsilon_R = \varepsilon_{r1} + (-\varepsilon_{r2}) = \varepsilon_{r1} - \varepsilon_{r2}$$

When the physical dimensions of the Rx coils are equal, and they are placed in an equal distance from the Tx coil, ε_R becomes zero and the detector head is called *balanced*.

Now looking at Fig. 1.2 an electrical conducting metal object, s, is illustrated to the right of the detector head. The object is moving into the opening of the detector head, through it and out on the other side. When the object moves into the detector head it is exposed to the magnetic field $\mathbf{B}_T$, as depicted in part a) of Fig. 1.4.

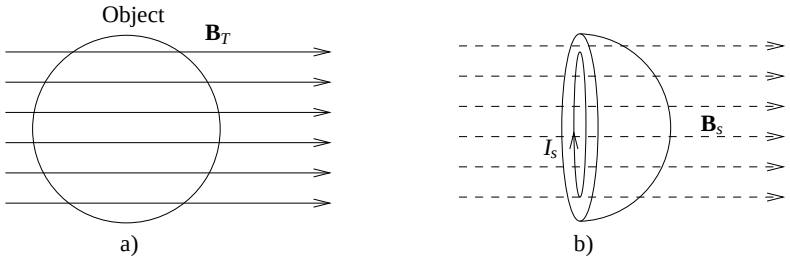

Figure 1.4: Object exposed to magnetic field $\mathbf{B}_T$. Figure a) illustrates the object being exposed to $\mathbf{B}_T$, and b) illustrates the induced Eddy currents I_s in the object, together with the new field, due to the Eddy currents.

The metal object is now steady in relation to the position of the coils. When the object is exposed to $\mathbf{B}_T$, currents are forced to flow - *these currents are called Eddy currents*. In part b) of Fig.

1.4 one half of the object is shown in which Eddy currents, I_s, flow. Due to the Eddy currents a secondary magnetic field occurs. This is shown in part b) of Fig. 1.4. This secondary magnetic field $\mathbf{B}_s$ is assumed to add with the primary magnetic field $\mathbf{B}_T$ (Superposition Principle), such that the resultant magnetic field, $\mathbf{B}_R$, at the Rx coils becomes,

$$\mathbf{B}_R = \mathbf{B}_T + \mathbf{B}_s$$

In practice the field strength decreases, when increasing the distance to the source of the field. Relating this to the scenario of moving the object into the detector head, gives rise to different field strengths of $\mathbf{B}_s$ at each Rx coil. Thereby the resultant induced electromotive force across the Rx coils in the balanced detector head becomes,

$$\varepsilon_R = (\mathbf{B}_T + \alpha\mathbf{B}_s) - (\mathbf{B}_T + \beta\mathbf{B}_s) = (\alpha - \beta)\mathbf{B}_s \tag{1.6}$$

where α and β are constants determining the magnetic field strength, because of the distance from object to receiving coil 1 and 2 respectively. Thereby, if an object is present in detector head, it is possible to detect it by means of $\mathbf{B}_s$.

From the above initial description of the detector head, the mathematical model can be established. But before this, two points should be commented before proceeding to the mathematical modelling.
The magnetic fields treated was assumed to propagate linearly for illustration purpose. Also the object moved into the detector head, was assumed electrically conducting to explain the Eddy currents. These assumptions are not needed, when modelling the detector head, which is caused by the model being derived from Maxwell's equations.

In the following the mathematical model is established from Maxwell's equations.

1.2 Three-Coil Detector Head Model

This section describes the mathematical modelling of the detector head. The model is based upon the articles [Yamazaki et al., 1996], [Yamazaki et al., 1998] and [Yamazaki et al., 2001]. The derivations are performed to understand the underlying assumptions, and to be able to analyse the origin of the signals, from which metal detection should be performed. Also by establishing the model, it is possible to indentify problems in the detector head, and to obtain parameters to optimize the signal-to-noise ratio. The section is parted into six subsections, each concerning a specific topic in the derivation of the model:

Sec. 1.2.1 Definition and derivation of the general vector potential, defined through the magnetic field due to a current carrying loop (the transmitting coil).

Sec. 1.2.2 Derivation of the general solution to Maxwell's equations, which is used to describe the vector potential in differential form inside and outside the object.

Sec. 1.2.3 Derivation of the vector potential inside the object, due to the applied field and the eventual Eddy currents in the object.

Sec. 1.2.4 Derivation of the vector potential outside the object, due to the current carrying loop and the existence of the object.

Sec. 1.2.5 A specific solution to the vector potential in space due to the existence of the object. The section also describes the boundary conditions, which are necessary to find a specific solution.

Sec. 1.2.6 Derivation of the induced electromotive force due to the presence of the object.

To model the response of an object in the detector head, the object is assumed to be solid and spherical. Assuming a spherical shape, will make the derivations less complex. Also the spherical shape is the standard reference shape of objects, in the metal detection industry, according to Detectronic A/S. The fully detailed mathematical derivation is found in Cha. D.1.

The model to be developed is based on Fig. 1.5, which shows the setup with the transmitting coil and an object in origo. The object is defined to be of a spherical shape, such that the derivations can be based on spherical coordinates. In the figure of the setup, the spherical orientation (r, θ, ϕ), together with the rectangular counterpart (x, y, z), is shown.

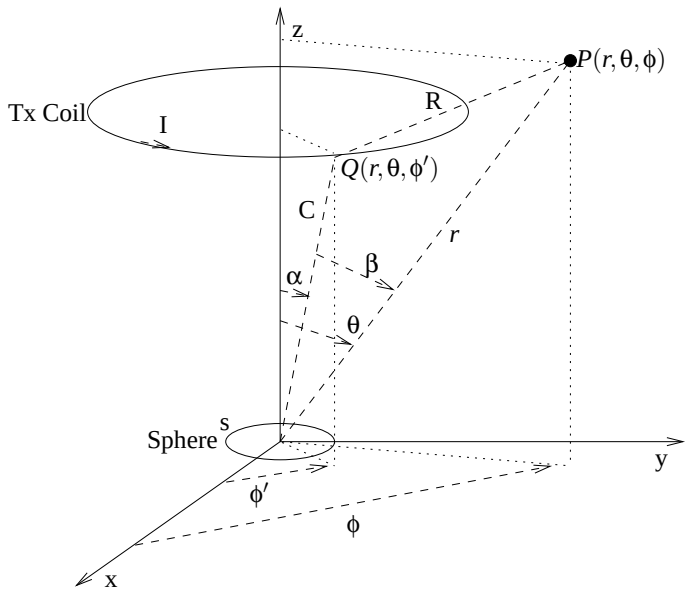

Figure 1.5: Model of the detector head system.

In the following subsections the angles and distances given in Fig. 1.5 are used to develop the mathematical model. First the general representation of the vector potential is defined.

1.2.1 Vector Potential in General

The vector potential, A, is way of representing the magnetic field, $\mathbf{B}$, in some point $P(r, \theta, \phi)$ in space. Through the definition of the vector potential, $\nabla \times A = \mathbf{B}$, the expression for the general vector potential can be obtained,

$$A = \sum_{n=1}^{\infty} A_n P_n^1(\cos\theta) r^n \tag{1.7}$$

$$A_n = \frac{\mu_0 I \sin\alpha}{2n(n+1)} C^{-n} P_n^1(\cos\alpha) \tag{1.8}$$

where

- $P_n^1()$ is the Associated Legendre function of first kind
- μ_0 is the permeability of free space
- I is the amplitude of the current in the transmitting coil
- α is the angle between the z-axis and the line of length C
- θ is the angle between the z-axis and the line of length r

The expression for A can be used to calculate the magnetic vector potential in the point $P(r, \theta, \phi)$. The vector potential is of interest in the derivations only, because the mathematical formulation is converted to an induced voltage in the receiving coils as the last step in the modelling.

The vector potential, in the form given in Eq. 1.7 can be written in the differential form due to the definition through **B**. This is used in the following, where Maxwell's equations are solved for the specific case.

1.2.2 Vector Potential in Differential Form

Due to the definition of the vector potential and the electric field, **E**, Maxwell's first equation can be expressed as,

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \frac{\partial A}{\partial t} \tag{1.9}$$

This representation of Maxwell's first equation is useful, because it states the relation between the magnetic vector potential in any point in space, and the electric field. Maxwell's second equation states the connection between a current density, **J**, and the magnetic field strength, **H**,

$$\nabla \times \mathbf{H} = \nabla \times \frac{1}{\mu}\mathbf{B} = \nabla \times \frac{1}{\mu} \nabla \times A \tag{1.10}$$

which by use of Maxwell's other equations can be written as,

$$\nabla \times \nabla \times A = -j\omega\mu(\sigma + \omega\epsilon)A \tag{1.11}$$

At this point it is assumed, that the term $\omega\epsilon$ can be neglected at sufficiently low frequencies. The validity of this assumption is discussed in Sec. 1.4. The expression in Eq. 1.11 describes the connection between the magnetic vector potential and the currents flowing in the spherical object - *this is the Eddy currents.*

The definition of the vector potential is now used to obtain a more general expression, where it is utilised that it only exists in the ϕ-direction. The solution is,

$$[\nabla \times \nabla \times A_n]_\phi \tag{1.12}$$

$$= \frac{1}{r^2}\frac{\partial}{\partial r}\left\{r^2 \frac{\partial A_n}{\partial r}\right\} + \frac{\sqrt{1-u^2}}{r^2}\frac{\partial^2}{\partial u^2}\left\{\sqrt{1-u^2}A_n\right\} = jPA_n \tag{1.13}$$

where $u = \cos\theta$ and $P = \omega\mu\sigma$. This general solution consists of two second order differential equations and is needed in the two following subsections, where the vector potential is evaluated inside and outside the spherical object.

1.2.3 Vector Potential Inside Sphere

The solution of Eq. 1.13 for the vector potential inside the sphere, $A_{s,n}$, is given as,

$$A_{s,n} = F_n \cdot \sqrt{r} \cdot I_{n+\frac{1}{2}}(\sqrt{jP} \cdot r) \cdot P_n^1(u) \tag{1.14}$$

where

- F_n is a constant due to the solution of the differential equation
- $I_{n+\frac{1}{2}}()$ is the Modified Bessel function of an order corresponding to half an integer

The expression is a rather complex form, but it should not be used as it is. In the following subsection an expression for the vector potential outside the sphere is obtained.

1.2.4 Vector Potential Outside Sphere

The solution of Eq. 1.13 for the vector potential outside the sphere, $A_{o,n}$, is given as,

$$A_{o,n} = S_n \cdot P_n^1(u) \cdot r^n + T_n \cdot P_n^1(u) \cdot r^{-n-1} \tag{1.15}$$

where S_n and T_n are constant due to the solution of the differential equation.

Expressions for the vector potential inside and outside the sphere are now obtained, and the next step is to find the change in vector potential, due the presence of the sphere. This change in potential is vital for expressing the induced voltage in the receiving coils due, to an object exposed to the magnetic field from the transmitting coil.

1.2.5 Vector Potential Due to Sphere

To obtain the change in magnetic vector potential due to the sphere, ΔA_o, the two solutions from Eq. 1.14 and Eq. 1.15 should be used. The problem is solved in the boundary of the sphere, where the inside and outside vector potentials are equal. The solution is given as,

$$\Delta A_o = \sum_{n=1}^{\infty} \frac{\mu_0 I \sin\alpha C^{-n}}{2n(n+1)} P_n^1(\cos\alpha) P_n^1(\cos\theta) \cdot V_n \tag{1.16}$$

$$V_n = a^{2n+1} \left(1 - \frac{(2n+1) \cdot \mu_r \cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}{\sqrt{jP} \cdot a \cdot I_{n-\frac{1}{2}}(\sqrt{jP}a) + n(\mu_r - 1) \cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)} \right) r^{-n-1} \tag{1.17}$$

where

- a is the radius of the sphere [m]
- $\mu_r = \frac{\mu}{\mu_0}$ is the permeability relative to some material
- $I_{n-\frac{1}{2}}()$ is the Modified Bessel function of half an integer order

The change in vector potential is the basis for deriving the induced voltage in the receiving coils due to the sphere. The induced voltage in found in the following subsection.

1.2.6 Induced Voltage In Receiving Coils

In the previous sections the change in vector potential due to a spherical object and a current carrying loop was derived. To obtain the model of the detector head, two receiving coils are combined. Faraday states, that a time varying magnetic field through a closed surface, induces an electromotive force expressed as,

$$\varepsilon = -\frac{\partial}{\partial t}\int_S \mathbf{B} d\mathbf{s} = -\frac{\partial}{\partial t}\int_S \nabla \times A d\mathbf{s} \tag{1.18}$$

By use of Stoke's theorem, Eq. 1.18 is converted from a surface integral to a line integral over a closed path,

$$\varepsilon = -\frac{\partial}{\partial t}\int_S \nabla \times A d\mathbf{s} = -\frac{\partial}{\partial t}\oint_C A dl = -j2\pi\omega R_r A \tag{1.19}$$

where R_r is the radius of the receiving coil. The receiving loops are placed in the ϕ direction, because the vector potential is only present in ϕ. The vector potential at the receiving coils are a sum of the vector potentials from the current carrying loop and the change in vector potential, due to the spherical object. If the receiving coils are placed in equal distance from the transmitting coil, and are otherwise equal, the coils will experience the same vector potential from the transmitting coil, which will cancel out because of the balanced setup. In Fig. 1.6 the model of the balanced

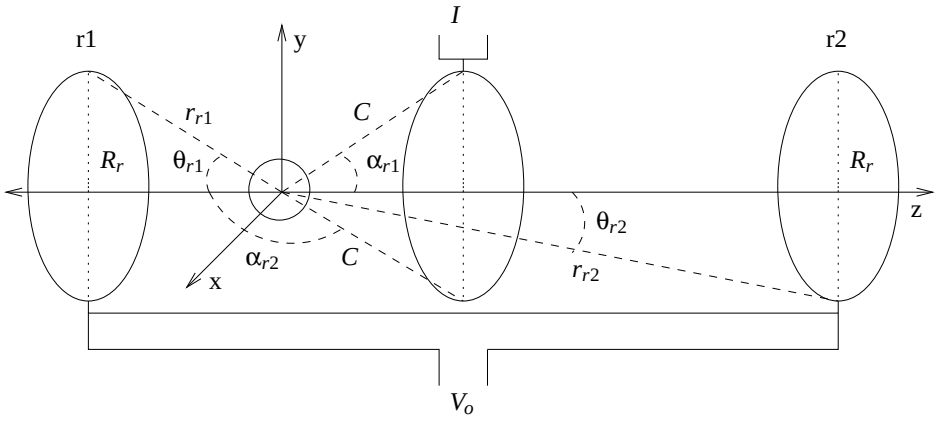

Figure 1.6: Model of the balanced system consisting of two receiving coils and one transmitting coil.

coupled N_r turn receiving coils and the N_x turn transmitting coil is shown. The transmitting coil, receiving coils and sphere are place on the z-axis. To find the resultant electromotive force in the receiving coils only one coil is observed. The vector potential at the contour of the receiving coil, is the same as a change in vector potential in a point P, described in the previous section. Because of this, the induced force in the coils, can be described as,

$$\varepsilon_{r1} = -j2\pi\omega N_x N_r R_r \sum_{n=1}^{\infty} \frac{\mu_0 I \sin\alpha_{r1}}{2n(n+1)} C^{-n} P_n^1(\cos\alpha_{r1}) P_n^1(\cos\theta_{r1}) \cdot r_{r1}^{-n-1} \cdot V_n \tag{1.20}$$

where r_{r1} is the distance from origo of the sphere, to the contour of the receiving coil, α_{r1} is the angle between the sphere and the contour of the transmitting coil, and θ_{r1} is the angle between

the sphere and the contour of the receiving loop. The term V_n is given in Eq. 1.17. To extend the model to two balanced receiving coils, the electromotive forces of the receiving coils are subtracted,

$$
\begin{aligned}
V_o &= \varepsilon_{r1} - \varepsilon_{r2} && (1.21)\\
&= -j2\pi\omega N_x N_r R_r \sum_{n=1}^{\infty} \frac{\mu_0 I \sin\alpha_{r1} C^{-n}}{2n(n+1)} P_n^1(\cos\alpha_{r1}) P_n^1(\cos\theta_{r1}) \cdot r_{r1}^{-n-1} \cdot V_n \\
&\quad +j2\pi\omega N_x N_r R_r \sum_{n=1}^{\infty} \frac{\mu_0 I \sin\alpha_{r2} C^{-n}}{2n(n+1)} P_n^1(\cos\alpha_{r2}) P_n^1(\cos\theta_{r2}) \cdot r_{r2}^{-n-1} \cdot V_n && (1.22)\\
&= j2\pi\omega N_x N_r R_r \sum_{n=1}^{\infty} \frac{\mu_0 I C^{-n}}{2n(n+1)} V_n \cdot (\sin\alpha_{r2} P_n^1(\cos\alpha_{r2}) P_n^1(\cos\theta_{r2}) \cdot r_{r2}^{-n-1} \\
&\quad - \sin\alpha_{r1} P_n^1(\cos\alpha_{r1}) P_n^1(\cos\theta_{r1}) \cdot r_{r1}^{-n-1}) \ \ [\mathrm{V}] && (1.23)
\end{aligned}
$$

V_o is the induced voltage in the receiving coils of the system, measured differentially across the receiving terminals.

The mathematical model is now capable of describing the system, which the sinusoidal current in the transmitting coil is affected by, before measured at the receiving coils. Throughout the derivation of the model, the permeability was assumed constant, which is not true. This is investigated in the following section.

1.3 Frequency Dependent Permeability

Until this point the permeability μ_r, relative to some material or object, has been seen as a constant - this is not true for ferromagnetic materials e.g. iron. The mathetical model derived in Sec. 1.2, was based on the paper [Yamazaki et al., 2001], being one out of three, in which the author verifies the model for aluminium and stainless steel, but obtain large deviations between measurements and simulation results, based on the model for iron. Based on this problem, this section is dedicated to indentify an approximation to the permeability function, which is assumed only to depend on the frequency. In practice, the permeability depends on many factors, e.g. the magnetic field strength [Ebert, 1998], but this fact is neglected in the model developed in this report.

From Maxwell's equations the permeability can be identified as,

$$\mathbf{B} = \mu_0 \mu_r \mathbf{H} \tag{1.24}$$

$$\mu = \frac{\mathbf{B}}{\mathbf{H}} \tag{1.25}$$

Where $\mu = \mu_0\mu_r$, and the permeability of free space μ_0 is constantly defined as $4\pi 10^{-7}$. If the relative permeability is *a constant* the magnetic field would follow an eventual change in the magnetic field strength momentarily. If this visualized in a plot, with $\mathbf{H}$ on the horisontal axis and $\mathbf{B}$ on the vertical axis, a zero-crossing straight line is obtained with slope μ. If μ varies with the strength of the applied magnetic field, the ratio $\mathbf{B}/\mathbf{H}$ is no longer a straight line with constant slope μ, i.e. the permeability is now a function of $\mathbf{H}$, giving $\mu(\mathbf{H})$. This fact makes the relation between $\mathbf{B}$ and $\mathbf{H}$ a nonlinear function of $\mathbf{H}$ - this is called magnetic hysteresis.
If instead the frequency of $\mathbf{H}$ is varied, and the magnetic field strength is held constant, the hysteresis is also expected to change. This is analysed in the following.

To make an approximation of the permeability function of frequency, $\tilde{\mu}_r(f)$, a linearisation is performed. The linearisation consists of fixing the magnetic field strength, as discussed above. This way $\tilde{\mu}_r(f)$ can be seen as the slope of a zero-crossing straight line, representing the $\mathbf{B}/\mathbf{H}$-relation. It is also assumed that the material of interest is completely magnetised, such that the zero-crossing assumption hold.
The low frequency value for the permeability is here defined as $\mu_r(0)$, corresponding to the permeability, when the frequency is close to DC. The relative permeability at infinite frequency $\mu_r(\infty)$ approaches one, which corresponds to the magnetic properties of air at sea level. The boundaries for the approximated permeability function is given as,

$$\tilde{\mu}_r(f) = \begin{Bmatrix} \mu_r & \text{for } f = 0 \\ 1 & \text{for } f = \infty \end{Bmatrix} \tag{1.26}$$

An expression, that is able to satisfy these conditions, can be given as,

$$\tilde{\mu}_r(f) = 1 + (\mu_r(0) - 1)\mathrm{e}^{\gamma(f)} \tag{1.27}$$

where $\gamma(f) = -f$ to satisfy the boundary conditions. If $f = 0$ the exponential function becomes one and $\tilde{\mu}_r(0) = \mu_r(0)$. If on the other hand $f \to \infty$ the exponential function becomes zero and $\tilde{\mu}_r(\infty) = 1$. The expression for $\tilde{\mu}_r(f)$ as it is now exhibits the disadvantage of never converging, because it depends on f alone, and then of the boundary in $f = \infty$. This can be compensated for by introducing a scaling constant, ψ, in the exponential, giving,

$$\tilde{\mu}_r(f) = 1 + (\mu_r(0) - 1)\mathrm{e}^{-\frac{f}{\psi}} \tag{1.28}$$

The scaling constant ψ should be determined either emperically or experimentally, however this is discussed in a later section, where the mathematical model is verified. In the following the limitations of the model are discussed.

1.4 Limitations and Validity

From the derivation of the model, different limitations are made. In this section, these limitations will be discussed to clarify the use of the model. If the limitations are not fulfilled, the outcome of the physical detector head cannot be interpreted. The derivation of the model, introduces five assumptions or constrains which limit the model. The limits are, that the receiving and transmitting coils have to be circular, the input has to be sinusoidal and low frequency, and the object has to be solid and spherical. In the next subsections, the assumptions are discussed, and the boundary for validation is identified.

Circular vs. Rectangular Coils

The model is assumed only to have circular coils. It is assumed that the model will be approximately valid for quadratic coils, where the length of the perimeter is equal to the circumference of the circular coils. The model is also assumed to be valid for rectangular coils, where the width and height of the coils are close.

Sinusoidal vs. Arbitrary Signals

The model is only valid for sinusoidal signals, because all derivatives in the derivations are expressed as,

$$\begin{aligned} s(t) &= \sin(\omega t) \\ \frac{\partial s(t)}{\partial t} = \omega\cos(\omega t) &= j\omega\sin(\omega t) = j\omega s(t) \end{aligned}$$

The only signals, which satisfies this condition, are sinusoidal signals. To extend the model to arbitrary signals all differentiations of signals has to be substituted with the derivative of the respective signal. The limitation to sinusoidal signals is performed to simplify the derivations, but can be validated by the fact, that the current in the transmitting coil is sinusoidal in the practical system.

Low Frequency

The model is valid for low frequencies, because it is assumed that $\omega\epsilon << \sigma$, and $P = \omega\mu(\sigma + j\omega\epsilon)$ is reduced to $P = \omega\mu\sigma$. This assumption holds for low frequencies and conducting materials. If the condition $\omega\epsilon << \sigma$ is not true, the model cannot be assumed valid.
Typical values for ϵ_r are 80.1 for water at 20 degrees celsius, and 3 to 15 for salt. This relative permittivity is related to ϵ through the relation,

$$\epsilon = \epsilon_r\epsilon_0 \tag{1.29}$$

where $\epsilon_0 = \frac{1}{36\pi}\cdot 10^{-9}[F/m]$. Thereby it is seen that ϵ is very small, and compared to the conductivity σ in Eq. 1.11, an estimate of the highest valid frequency can be given. Using 100 as a rather high value for ϵ_r, and 1 as a rather low value for σ, the limit where 1 % error is reached can be obtained,

$$\frac{\sigma}{\omega\epsilon} = 100 \Leftrightarrow \omega = \frac{\sigma}{100\epsilon} \tag{1.30}$$

$$\Leftrightarrow \omega = \frac{1}{100\cdot(100\cdot\frac{1}{36\pi}\cdot 10^{-9}}) = 11.294\cdot 10^6\ [Rad/s] \tag{1.31}$$

corresponding to 1.7975 MHz. The 5 % error limit is 8.9876 MHz. If instead a normal conductor is considered, which has a conductivity of about 10^7, the highest frequency becomes $10^7 \cdot 1.7975$ MHz. Thereby, if conducting material is analysed, the frequency need not be considered, but if non-conducting material is analysed, the frequency should be kept below about 5-10 MHz to minimize the error.

Spherical vs. Arbitrary Form Objects

The model is limited to spherical objects, because the boundary conditions are only valid for spherical objects. The model can be assumed to be valid for quadratic objects, because the geometry of an quadratic object does not differ much from a spherical object, and the boundary conditions are defined with use of the geometry of a sphere.

When the limitations of the model are not meet, the model cannot be assumed to be valid. In the previous sections some limitation were performed to ease the derivations of the model. When the limitations are violated, it will cause a difference between the physical system and model. The difference is not further described, but should be kept in mind, when the model is used.

1.5 Verification

The mathematical model of the detector head coils, was developed in the preceding sections. This section seeks to verify the model by comparing simulation results, with measurements on a physical detector head model.

The verification is grounded in the phase between the current in transmitting coil and induced voltage in the receiving coils, for frequencies from approximately 100 Hz to 500 KHz. The phase should be found for aluminium, brass, iron and ferrite, which are materials representing different conductive and magnetic properties.
Based on the discussion in [Yamazaki et al., 2001], where the author shows that the model is inaccurate for an iron sample, an extension to the model is made in Sec. 1.3. This extension is also a subject for verication in this section.

First a description of the physical detector head model is given, which is followed by presentation of the results.

1.5.1 Physical Model

To verify the validity of the mathematical model, a physical model is constructed. The physical model is made as a solenoid, which corresponds to the type of coil, on which the mathematical model is based. To obtain homogeneous geometry, the model is wound on a cardboard cylinder, which has no noticeable magnetic or conducting properties. In all three coils, there are $N_x = N_r = 25$ closely wound windings of 0.75 mm isolated copper wire. The radii of the coils are $R_x = R_r = 30.0$ mm, and the distance from the center of the transmitting coil to each of the receiving coils is 20.0 mm.

The measurement on the physical model is described in App. B, and the results are presented together with simulation results in the following.

1.5.2 Results and Discussion

Before presenting the results, the different materials to be used are discussed. The four types of material, represent different magnetic and electric properties. This is important, when verifying the mathematical model, because it is expected to be valid for any material, with dielectric properties that can be neglected. This is valid for aluminium, bras, iron and ferrite, together with many other materials. The specific properties of the materials are given in Tab. B.3 in App. B.

In Fig. 1.7 through Fig. 1.10, a comparison between the mathematical and physical model, is presented. The figures present two plots each,

- *the upper plot* gives the phase vs. frequency of model and measurements
- *the lower plot* gives the error between model and measurements (the difference)

Based on the results for the four materials, the error between model and measurements is analysed. As described in Sec. 1.3, the permeability is corrected, and the scaling constant is determined experimentally to $\psi = 80000$. In Tab. 1.1 the mean of the error, together with correlation coefficient

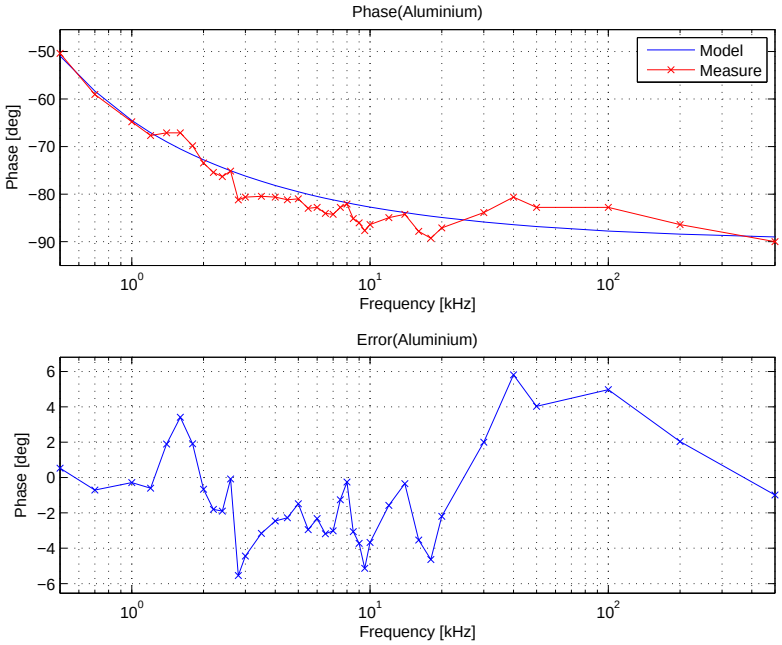

Figure 1.7: Comparison between mathematical model and measurements on physical model for aluminium. The upper plot shows phase vs. frequency and the lower plot is error vs. frequency.

Material	**Mean** [deg]	**CC**
Aluminium	-1.07	0.95
Brass	-3.32	0.97
Iron	0.08	0.99
Ferrite	-3.17	0.64

Table 1.1: Statistics concerning detector head model and measurements. For the four materials mean of the error, together with the correlation coefficient (CC) between model and measurement, are given. The values are based on the results given in Fig. 1.7 through Fig. 1.10.

between model and measurements, are given.
Looking at Fig. 1.7 through Fig. 1.10 the mathematical model and the measurements on the physical detector head model is seen to follow the same tendencies. From Tab. 1.1 the correlation coefficient is obtained, which describes the statistical connection between a set of observations,

$$CC_{i,j} = \frac{C_{i,j}}{\sqrt{C_{i,i} \cdot C_{j,j}}} \tag{1.32}$$

where $C_{i,j}$ is the covariance between observation i and j, $C_{i,i}$ is the variance of observation i, $C_{j,j}$ is the variance of observation j, and $CC_{i,j}$ is the correlation coefficient between observation i and j. From the correlation coefficient the model and measurements are seen to correlate (close to one), except for the case of ferrite. In the model ferrite is defined to have very high permeability and very low conductivity, which for the practical piece of material may be different. This is, however, the case for all the measurements.
The mean of the error is negatively biased for aluminium, bras and ferrite, which signifies a general offset in the model. It can however also be caused by consequent reading errors of the oscilloscope

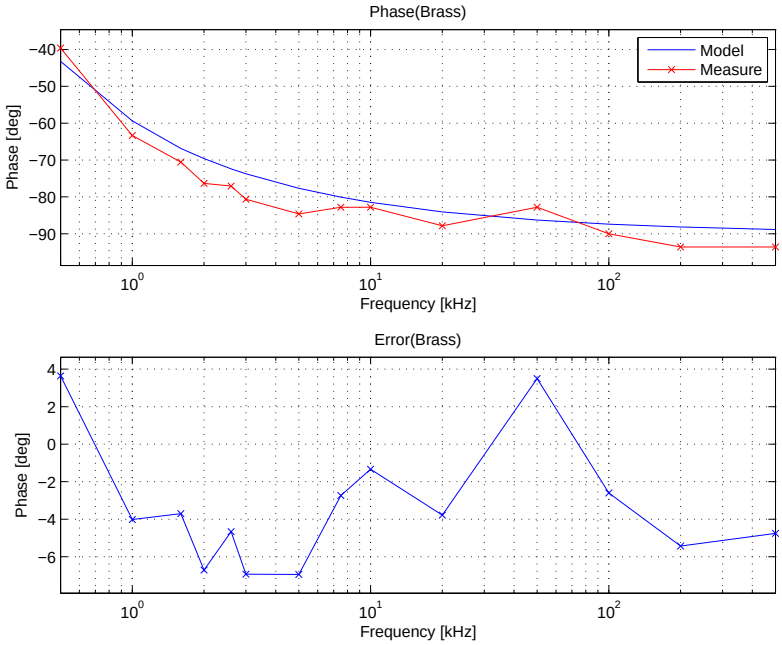

Figure 1.8: Comparison between mathematical model and measurements on physical model for bras. The upper plot shows phase vs. frequency and the lower plot is error vs. frequency.

in the measurement. For iron being, the mean value is close to zero, which can be caused by a good estimate of ψ.

1.6 Conclusion on Detector Head Model

A mathematical model has been developed, which is capable of describing the physical event of moving an object through the detector head. The output of the model, is the induced voltage across the receiving coils, which is a phase shifted and amplitude modulated version of the sinussoid in the transmitting coil. The model has been verified by comparison with a physical model for four different materials, possessing different magnetic and electric properties. The results show, that the output of the mathematical model correlates with the measurements on the physical counterpart. The mathematical model can be concluded as feasible for feature extraction, which was the goal of the modelling.

The following chapter takes the mathematical model into perspective, for the purpose of extracting features for metal detection.

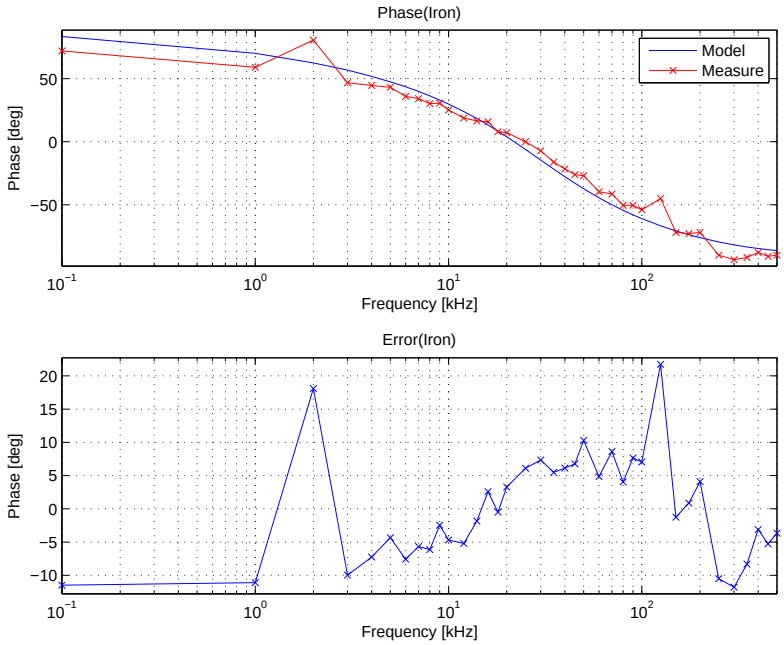

Figure 1.9: Comparison between mathematical model and measurements on physical model for iron. The upper plot shows phase vs. frequency and the lower plot is error vs. frequency.

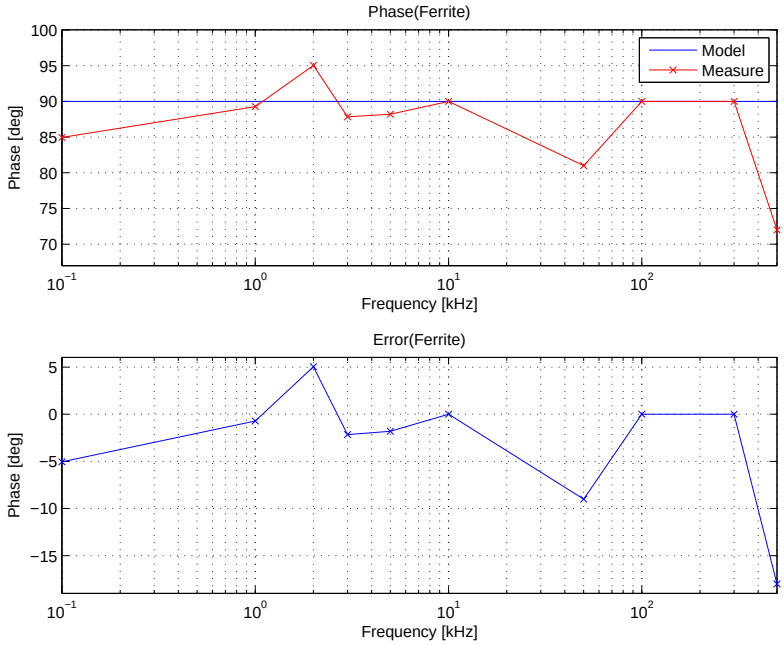

Figure 1.10: Comparison between mathematical model and measurements on physical model for ferrite. The upper plot shows phase vs. frequency and the lower plot is error vs. frequency.

Chapter 2

Model of Signals

In Cha. 1 the model of the metal detector head was derived for the purpose of describing and understanding the system behaviour. The model is used in this chapter to analyse the signals, that origins from the metal detector head, through simulations, but also through measurements. The purpose of simulation, is to find features, that can be used to differentiate a piece of metal from food articles in the detector head. In other words the chapter concerns modelling the signals and feature extraction. In the overall design phase this is the second topic, as depicted in Fig. 2.1.

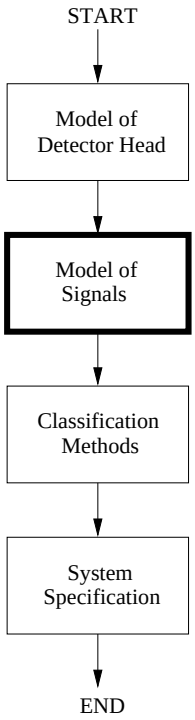

Figure 2.1: Present stage in the overall design flow is marked with a thick box. Present stage is Model of Signals

The chapter consists of four topics, which are all related to the signals, that origin from the metal detector head. There are two signals of concern, which are the current in the transmitting coil, $d(t)$, and the induced voltage in the receiving coils taken differentially, $u(t)$. The signal of interest

in this chapter is $u(t)$, because it is the one representing the response of the metal detector head. The four topics, which should investigate $u(t)$ are as follows:

- Out-of-Balance Component given in Sec. 2.1 is an analysis concerning the fact, that the receiving coils cannot be perfectly balanced. When the coils are not balanced, the induced voltages are unequal, which gives an Out-of-Balance (OoB) component in the differential signal $u(t)$. The section is based on measurements on a real metal detector, see App. C.
- Steady Objects in Detector Head given in Sec. 2.2 is a resume of an analysis concerning steady objects in the detector head. The purpose of the section is to extract features from $u(t)$, which can be used for separation of metal pieces from food articles. The complete analysis is found in App. E.2.
- Moving Objects in Detector Head given in Sec. 2.3 covers an analysis of moving objects in the detector head and further more a discussion of the superposition principle in relation to mixed objects.
- Product Signal-to-Noise Ratio given in Sec. 2.4 analyses the ratio between the induced voltages due to food articles and metal pieces, which is called Product Signal-to-Noise Ratio (SNR_p). This aspect is related to the transmitting frequency, which is shown to have great impact on the SNR_p.

The chapter concludes by summing up all the most important points and discusses these topics.

2.1 Out-of-Balance Component

The first topic of four in this chapter is an analysis of the raw signals due to the metal detector head. The raw signals are obtained according to App. C. From Cha. 1 it is known, that two signals are supplied from the metal detector head, and these are $d(t)$ and $u(t)$. In the mathematical model of the detector head the three coils are assumed to be in perfect balance. The fact that they are balanced, means that the induced voltages in both receiving coils, due to the magnetic field produced in the transmitting coil, are equal. If not balanced, unequal induction will result in different amplitudes in the signals from the receiving coils. Another aspect is that the coil dimensions are different, which implies that the inductances become unequal. A consequence of this is a phase difference in the two receiving coils, which will also give a contribution to the Out-of-Balance (OoB) component.
Assuming that the phase difference, between the signals from the receiving coils, can be neglected, the unbalanced observation signal *without* any objects in the metal detector can be expressed as,

$$s_{OoB}(t) = A_{R1}\sin(\omega t - \phi_{R1}) - A_{R2}\sin(\omega t - \phi_{R2}) \tag{2.1}$$

where A_{R1} and ϕ_{R1} are the amplitude and phase of the induced voltage in one receiving coil, and A_{R2} and ϕ_{R2} are those for the induced voltage in the other coil. Perfect balance in the system requires A_{R1} to equal A_{R2}, and ϕ_{R1} to equal ϕ_{R2}, which implies that $s_{OoB}(t) = 0$, when no objects are present in the detector head. The observation signal, when an object is present in the metal detector, can then be expressed as,

$$u(t) = s_{OoB}(t) + s(t) \tag{2.2}$$

where $s(t)$ refers to the signal resulting from the object. The OoB component cannot be neglected and therefore it is analysed in this section. To illustrate the effects of the out-of-balance component the sampled reference signal and the observation signal are presented. The frequency of the signals was defined to be 30 KHz, but to illustrate a duration of two seconds of the two signals in a single figure is impractical. Therefore, the signals are bandpass filtered, and then undersampled, such that the new signal frequency becomes 12 Hz . This undersampling process, corresponds to mixing the signals down by multiplication of a 29988 Hz sine, and then using a lowpass filter for removal of the *sum component*. The aliased signals correspond to the *difference component* in the mixing process, and they are shown in Fig. 2.2.

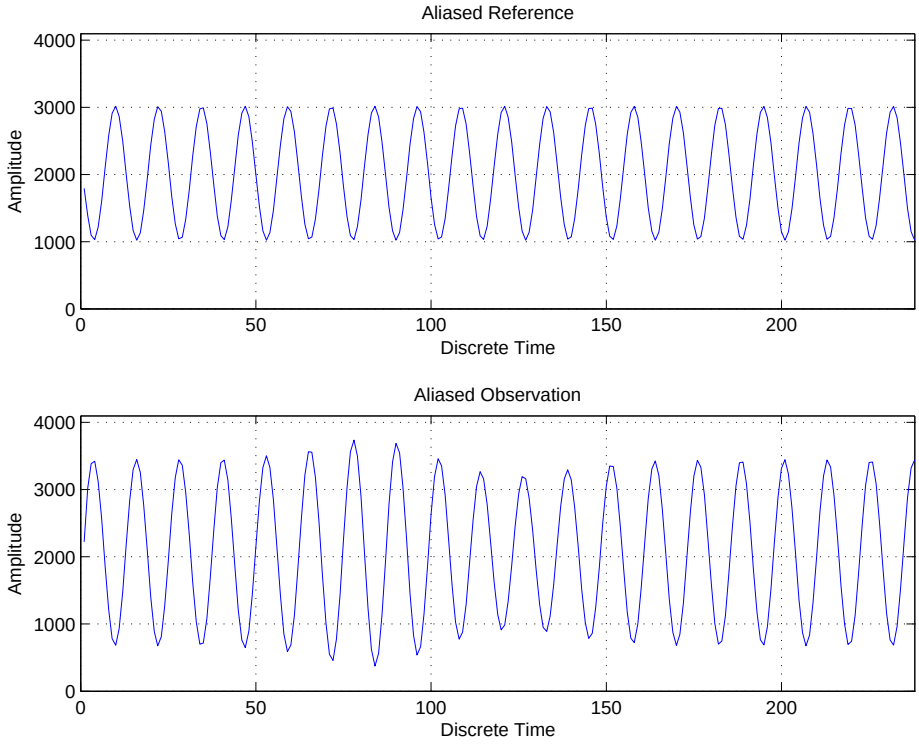

Figure 2.2: Aliased reference and observation signals, shown in 12 bit unsigned integer, corresponding to the data type of the data acquisition system. The original signal frequency is 30 KHz, but due to illustrational purposes, the signals are aliased into 12 Hz . The reference signal is due to the current in the transmitting coil, which is stationary. The observation signal shows the out-of-balance component and amplitude modulation due to a 10 mm iron sphere, moving through the metal detector.

In Fig. 2.2 the sampled and aliased measurements are shown. The horizontal axis presents time (2 seconds in total), and the vertical axis presents amplitude in 12 bit unsigned integer data type, corresponding to the precision of the data acquisition system. The reference signal from the transmitting coil is seen to be stationary in the window, which is also expected. The observation signal reveals amplitude modulation due a 10 mm iron sphere, moving through the detector head. The observation signal was expected to have zero amplitude in the beginning and end, because here no objects are present in the detector head. However, because the three coils are not balanced the OoB component is revealed. If the OoB component is removed with the technique described in

App. E.1, the signal is Fig. 2.3 is obtained.

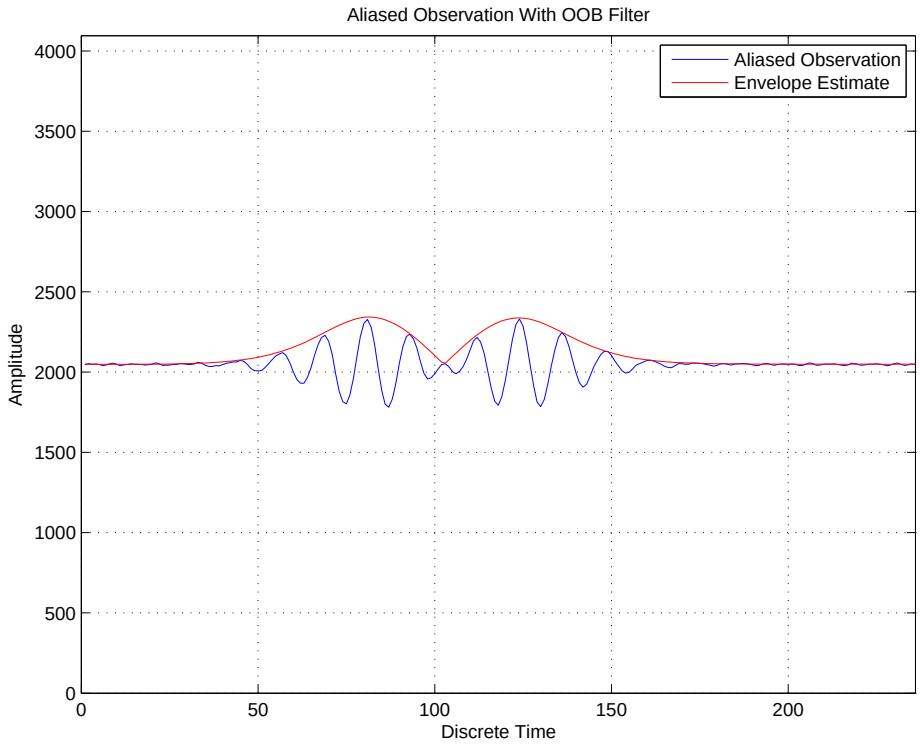

Figure 2.3: Aliased and balanced observation signal due to a 10 mm iron sphere. The original signal frequency is 30 KHz, but due to illustrational purposes the signals are aliased into 12 Hz . The discrete time interval shown is two seconds. The observations signal is now seen to be balanced, because $s_{OoB}(t)$ has been removed.

Fig. 2.3 shows the aliased observation signal with compensation for the OoB component. The vertical axis still represents the amplitude, and is illustrated in the same data type (12 bit unsigned integer) as in Fig. 2.2 to keep the scale. The horizontal axis shows discrete time, and in total 2 seconds are shown. The OoB compensation does not in itself reveal new information, however the amplitude and phase are biased by the OoB component, if it is not removed. To show the bias of the phase and amplitude, the signal containing the 10 mm iron sphere is Fourier transformed in smaller windows, corresponding to a Short Time Fourier Transform, without overlapping windows. The result is plotted with and without the OoB component in the complex plane.
Fig. 2.4 shows, that the OoB component displaces the spectrum from origo to a point, described by the phase and amplitude of the OoB component. The OoB component can be assumed to be a constant offset in the spectrum, which implies that the OoB component is stationary. When the OoB component is present in continuous (analog) time, it is the component that limits the amplification of the observed signal $u(t)$, in the sense that the amplitude of $u(t)$ is limited by the input voltage range of the A/D converter. Removing the OoB component in the digitised signal $u[n]$, does not contribute new information, however if the OoB component is removed in $u(t)$, more amplification can be performed and thereby smaller objects can be seen in $u[n]$. Observing Fig. 2.3 it is seen that the 10 mm iron sphere is represented in the amplitude interval $\{1800 : 2350\}$, corresponding to approximately 9 bits. If the OoB component is removed in $u(t)$, more amplification

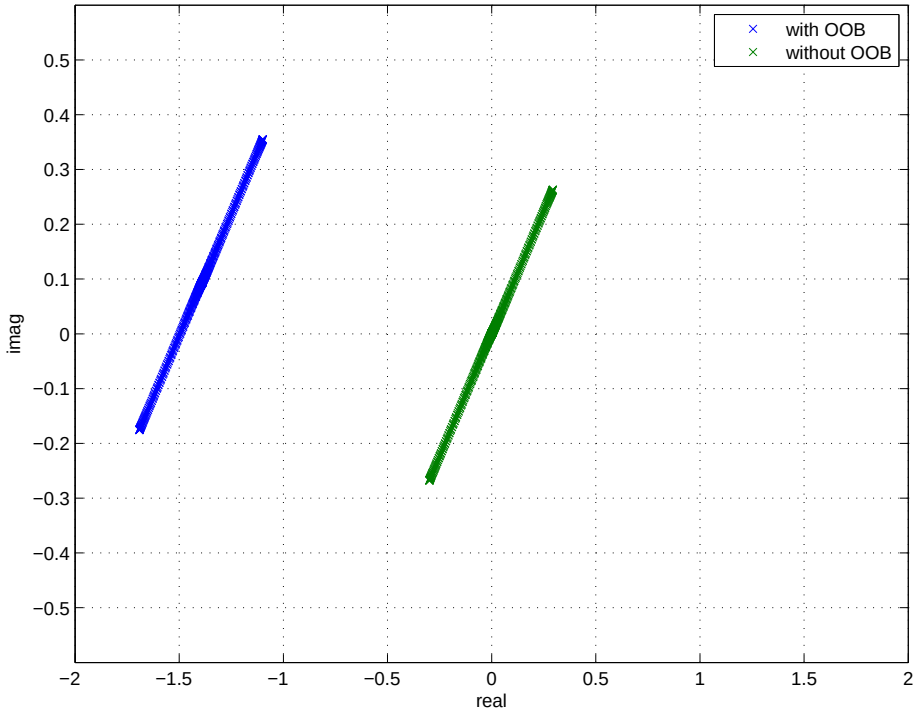

Figure 2.4: The figure shows the Short Time Fourier Transform, without overlap, of a signal that origins from a 10 mm iron sphere, with and without the OoB component.

can be performed and the 12 bits can be fully utilised.

2.1.1 Discussion

In this section the OoB component was described/discussed. It can be concluded, that the OoB component can be modelled as a constant and additive signal. In the complex plane, describing phase and amplitude, the OoB component can be interpreted as a constant offset from origo, due to the *quasi stationarity* of the component. The constant offset implies bias in the amplitude and phase estimates. It can be concluded, that the OoB component limits the ability to amplify the signal $u(t)$, before converting it to the digital domain $d[n]$, which limits the ability to represent signals from small objects.

The first of the four topics of this chapter has now been treated and an initial insight into the signals from the metal detector has been gained. In the next section the second topic of the chapter is treated, which concerns steady objects in the metal detector.

2.2 Steady Objects in the Detector Head

The importance of observing steady objects in the metal detector arises, because of two main aspects. The aspects consider the influence of phase and amplitude with respect to the size of object and material type. This section only covers the 30 KHz case, but analyses of steady objects with respect to frequency, size and type are placed in App. E.2. The reason for analysing objects in

the steady case, is that it is wanted to know which influence size and type of the object has on the phase and amplitude. The reason for not describing the product is based on the analysis in App. E.2, which showed that the phase and amplitude of products in general, was strongly dependent of the physical parameters. The types of material analysed in this section are aluminium, stainless steel (aisi 304) and iron. Measurements have been performed for all three metals, however simulations are only included for aluminium and iron, because the material parameters for aisi is unknown. The simulations are based on the model described in Cha. 1 and measurements on App. C.

2.2.1 Dependency of Size and Metal Type

This section has the purpose of clarifying the dependency of object size and material on the amplitude and phase. The materials taken into discussion is aluminium, aisi and iron. Fig. 2.5 shows the phase and amplitude of the three metals.

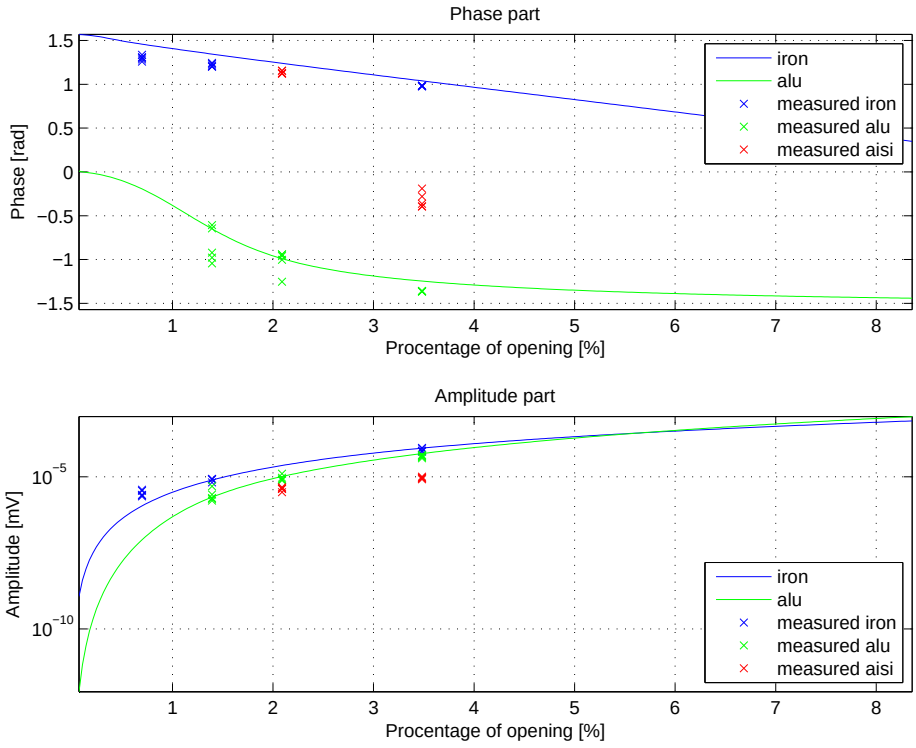

Figure 2.5: The figure shows the phase and amplitude of different sizes of iron and aluminium spheres based on simulations (solid lines). Also measurements (crosses) have been included, which cover aluminium, stainless steel (aisi) and iron.

The solid lines in the figure show the simulated phase and amplitude estimates as function of size, which is expressed as percentage of detector head opening. The phase and amplitude of aisi is not simulated, because the material parameters are unknown. The measurements are represented as crosses in the figure. At all measurements the amplification before the conversion to digital domain are equal. The exact amplification is not known, due to tuning of the amplification to fully utilize the input voltage range of the A/D converter. In the amplitude part of Fig. 2.5, all measurements are scaled with an equal constant, to be able to plot the measurements together with the simulations. The reason for only plotting the measurements of the small spheres is, that the gain, before

the A/D converter, had to be held constant, and due to the dynamic ranges of the signals, it is not possible to represent small and large spheres, within the same amplification range, without exceeding the input voltage range of the A/D converter. The phase part Fig. 2.5 shows, that the phase of iron starts at $\frac{\pi}{2}$ radians and drops towards zero. In fact, if a much larger iron sphere was used, it will converge to $-\frac{\pi}{2}$. The convergence at $-\frac{\pi}{2}$, when the size increases is valid for both iron and aluminium, but aluminium starts at zero instead of $-\frac{\pi}{2}$. Based on measurements of two sizes of aisi spheres, aisi seems to have the same tendency as iron, but converges faster. In general it can be concluded, that all spheres converges to $-\frac{\pi}{2}$ when the size is increased, but the transition and convergence speed depends on the type of material. In the complex plane, this corresponds to a rotation, when the size is changed.
Fig. 2.5 shows, that the measurements have the same tendency as in the simulation, but have a phase error in mean of 6.54 and 6.40 degrees for iron and aluminium respectively. The deviation in phase can be caused by not having the exact material parameters for the model, which is crucial for the simulation and measurement to correlate.

The amplitude part of Fig. 2.5 shows, that the amplitude of the signal from the sphere increases as the size increases. The measurements and simulations show that iron has the first, aluminium the second and aisi the third largest amplitude for small sizes of spheres. This means that it should be easier to detect and represent small iron spheres, than aisi spheres. The exponential increase, indicated by the straight lines in the logarithmic plot, shows that a little increase in size, makes it much easier to represent and detect the sphere due to the exponential increase. In the next section the dependency of the transmitting frequency will be shortly described, because there is an aspect in the transmitting frequency, that can be used in future work, but will not be further used in this master thesis.

2.2.2 Dependency of Transmitting Frequency

In App. E.2 the amplitude and phase are shown to depend on the transmitting frequency. The amplitude depends on the frequency, by increasing together with the frequency for all materials analysed. It is shown, that the phase resulting from the metal objects and food products, become independent of size, when the transmitting frequency is increased. The phase of convergence for the metal classes was found to be $-\frac{\pi}{2}$ Rad, and for the food articles the phase of convergence was 0 Rad. All materials analysed seemed to converge in phase at a transmitting frequency of approximately 1 MHz. There exist future aspects in moving the transmitting frequency to 1 MHz, and thereby get reduction in the amplitude for some products, without any reduction in the metal amplitude. This is further analysed in Sec. 2.4.

2.2.3 Discussion

The analysis of steady objects in the metal detector head revealed that the size of the object has a great impact on the amplitude and phase of the signal. The amplitude was seen to have an exponential increase, as the size increases. This means that it is harder to identify a sphere of a small size, than one of large size. The exponential increase should be remembered, because the amplitude is limited by the input voltage range of the A/D converter. It can from this be concluded, that it is better to amplify the signal before conversion, in a way that only the small spheres and product can be represented, without exceeding the input range of the A/D converter, because the larger

spheres has an larger amplitude, which makes them detectable in terms of amplitude alone. For small spheres it is easier to identify iron than aisi, because the amplitude of an iron sphere is larger than the one for aisi. The aluminium spheres are in between iron and aisi.

The phase was found to depend on the size and type, but all types of metal converges at $-\frac{\pi}{2}$, when the size was increased. The convergence speed was also found to depend on the type, because aluminium converges faster than iron. It can in general be concluded, that the phase of the signal from the detector is dependent on type and size. It can also be concluded, that when the size is increased, the phase moves towards $-\frac{\pi}{2}$.

To extend the signal model with the gained knowledge from this section, the signal term $s(t)$ in Eq. 2.2 is written as,

$$s(t) = A_m \sin(\omega t + \phi_m) \tag{2.3}$$

which gives the new observation signal,

$$u(t) = s_{OoB}(t) + A_m \sin(\omega t + \phi_m) \tag{2.4}$$

where A_m and ϕ_m is the amplitude and phase of the signal from the sphere.

In the following section, the third topic of the chapter is treated, which is strongly related to the second topic. Where the second topic was about steady objects, the third topic is about moving objects.

2.3 Moving Objects in the Detector Head

In the previous section the effects of a steady object in the detector head was analysed. In this section the signal model is extended to describe moving objects in the detector. The section starts by analysing the effects of one moving metal object in terms of phase and amplitude trajectory. Afterwards the model is extended to two or more objects moving. The section also analyses the effects of unpopped microwave popcorn moving through the detector head, to find differences that separate this product from metals. The analysis is based on simulations and measurements.

2.3.1 One Moving Object in the Detector Head

The section starts by analysing the object in terms of phase and amplitude trajectory, but moves it to cartesian coordinates, to see the relation between phase and amplitude in the complex plane.
In Fig. 2.6 the phase and amplitude of a 5 mm aluninium, and a 5 mm iron sphere, is plotted. Aisi is left out, due to the overflow of information in a single figure. The crosses represent measurements and the solid lines represent simulations. The distances at the horisontal axis, is the distance to the transmitting coil. The phase part of Fig. 2.6, shows that the phase shifts π radians, when the sphere crosses the centre of the transmitting coil. The phase away from the center of the transmitting coil is seen to be constant. The constant phases are the same as in Fig. 2.5. In Fig. 2.5 the phases were measured inside one of the two receiving coils. In Fig. 2.6 the complete phase trajectory is shown, which implies that the phase of the signal from the receiving coils is not

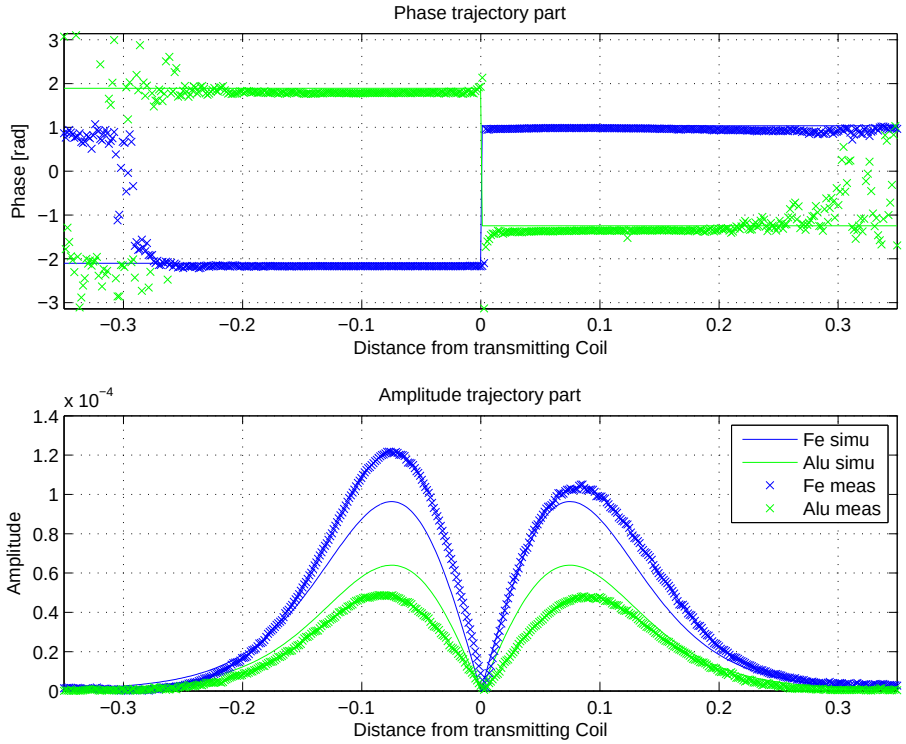

Figure 2.6: The figure shows the phase and amplitude trajectories for 5 mm aluminium and 5 mm iron spheres. The solid lines represent the simulation and the crosses the measurements.

constant due to the shift in the center of the transmitting coil. This shift is a consequence of the of the receiving coils being wound opposite of each other.

The amplitude trajectories are shown in the amplitude part of Fig. 2.6. The figure shows, that the amplitude of the signal increases until it reaches the first receiving coil, then decreases to zero at the center of the transmitting coil. The signal amplitude starts to increase again after the transmitting coil until the sphere reaches the second receiver coil, and then it decreases again. It shall be noted that the signals from the simulation are symmetric around the distance zero, which the measured signals are not. The reason for the asymmetry in the signals is due to the removal of the OoB component. If the OoB component is not removed perfectly, the amplitude trajectory is distorted by the asymmetrical error.

This discussion leads to, that the signal model in Eq. 2.4, is extended to,

$$u(t) = s_{OoB}(t) + A_m(t)\sin(\omega t + \phi_m(t)) \tag{2.5}$$

where $A_m(t)$ and $\phi_m(t)$ is the amplitude and phase of the signal, which are extended to be time dependent, due to Fig. 2.6.

To investigate the phase and amplitude trajectory of the unpopped popcorn, measurements have been performed.
In Fig. 2.7 the phase and amplitude of unpopped popcorn is presented. The horisontal axis represent discrete time, and at the discrete time index of approximately 340 the center of the transmitting

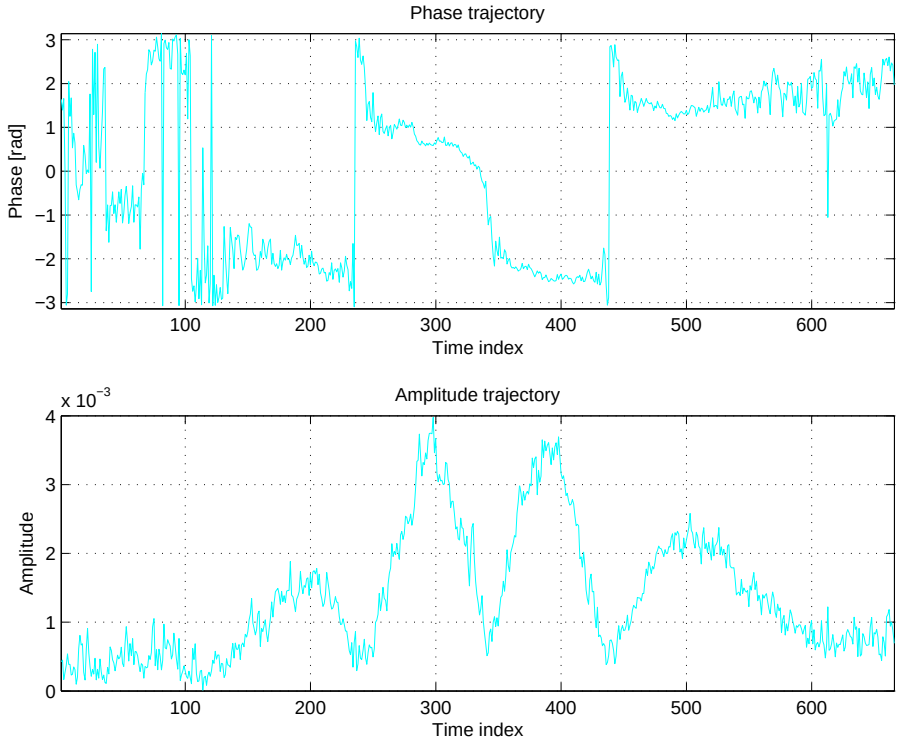

Figure 2.7: The figure shows the phase and amplitude trajectory of unpopped microware popcorn.

is located. The phase trajectory shows, that the phase shifts approximately π radians three times instead of one, as metal objects do. The three shifts are also reflected in the amplitude trajectory, because instead of two peaks, popcorn is represented with four peaks. The four peaks or three shifts cannot be verified by the model, because it is based on a single solid sphere. The popcorn package is flat and rectangular, which may result in more extrema. Also the popcorn consists of several materials of which fat and corn can be mentioned. The same is observed for other products, such as raisins and a package of winegums.
The phase and amplitude trajectory of unpopped popcorn shows, that the trajectory separates the product from the metal objects, due to the four peaks and three shifts. The different amplitude and phase trajectories makes it possible to separate popcorn, and other products, from metal objects in terms of trajectory.

The phase and amplitude trajectory of metal objects and popcorn, can be represented in cartesian coordinates, through the Short Time Fourier Transform. In cartesian coordinates, the phase and amplitude are given as a complex number. The reason for moving from polar coordinates, is to show the dependency between the real and imaginary part of the trajectory.

In Fig. 2.8 the trajectories for a 5 mm aluminium sphere and a 5 mm iron sphere are plotted, together with the trajectory of unpopped popcorn. All trajectories are normalised to have unit variance, to be able to plot them in the same figure. The figure shows, that the trajectory for aluminium and iron is represented by straight lines, with angles defined by the phases in Fig. 2.6. The phase shift of π in Fig. 2.6 corresponds to crossing origo. The length of the straight lines depends on the amplitude of the signals. The trajectory of the unpopped popcorn shows that it is

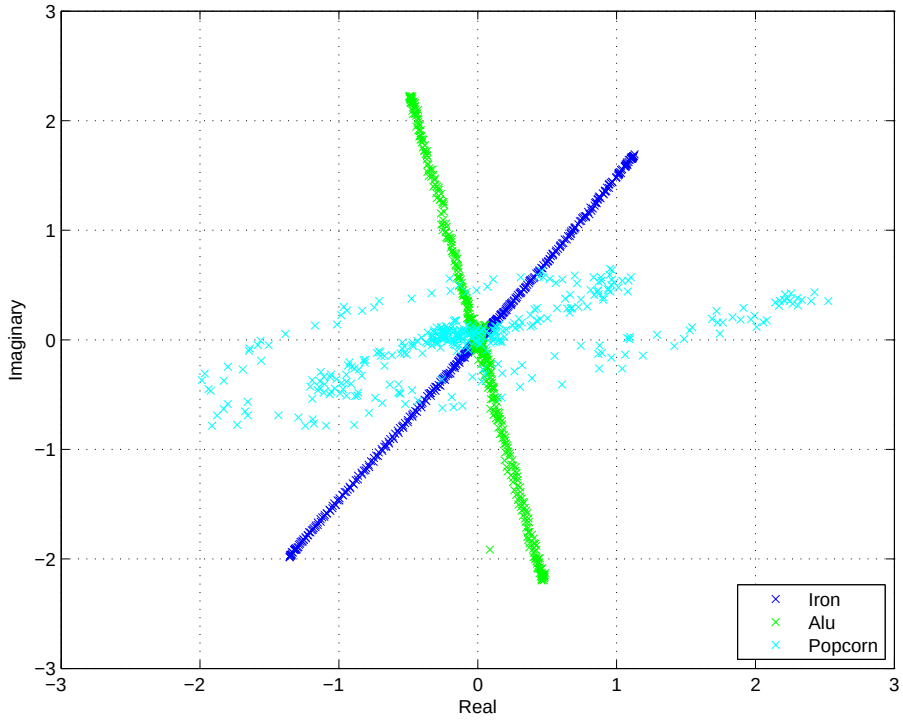

Figure 2.8: The figure shows the complex plane, where the phase and amplitude trajectory of a 5 mm aluminium sphere and a 5 mm iron sphere is plotted, together with the one for unpopped popcorn. All trajectories are normalised to have unit variance to be able to plot them together.

a rough approximation to a straight line, as it is not as well defined as the one for the aluminium and iron spheres. The spread around the straight line origins from the four peaks, but also a noise term is seen to contribute, if attention is paid to Fig. 2.7. The noise becomes significant for the popcorn product due to a weak amplitude. Fig. 2.8 also shows that the phase of unpopped popcorn is different, than that of an aluminium and an iron sphere of 5 mm. If however the size of the metal is changed, the phase between metal and product can coincide. This means that some products will coincide with the metal, which makes them unseperable unless the trajectory is examined.

In the next section, more than one metal object in the detector will be analysed.

2.3.2 More Moving Objects in the Detector Head

This section seeks to analyse the effect of more than one moving object in the detector head. The reason for this analysis, is to find out in which way the signals from two or more objects are mixed, in the resultant signal measured at the receiving coils. In normal operation, the metal detector has to find out if a product is contaminated with metal or not. This leads to the conclusion, that in normal operation of the metal detector, the signal measured at the receiving coils, will be a mixture of product and metal, if the metal is present. Because of this mixture, it is important to know how the signals are mixed, to be able to detect a contaminated product.

The analysis of the mixture is based on a 2 mm iron and a 3 mm aluminium sphere. To find

the way the signals mix in the detector head, the 2 and 3 mm spheres are moved through the detector head separately, and afterwards the spheres are fixed together and moved through the head. The reason for choosing a 2 mm iron sphere and a 3 mm aluminium sphere, is that the amplitude of the signals are nearly the same, and the phases of the straight line are nearly orthogonal. Under these conditions, it should be more clear, if the signals are an additive mixture, corresponding to the superposition principle to be valid.

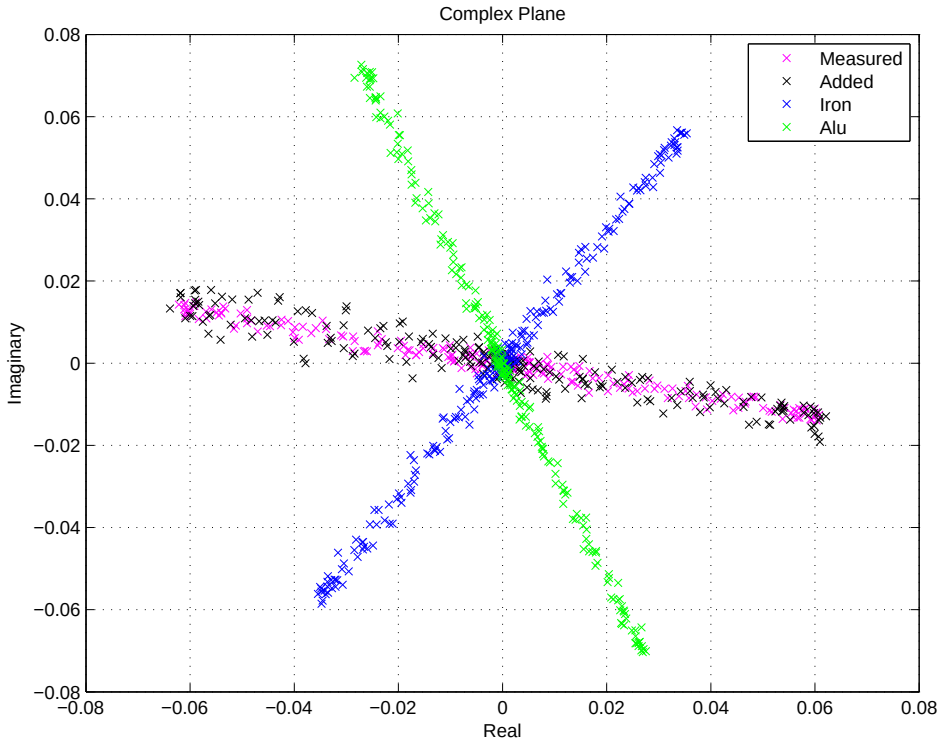

Figure 2.9: The figure shows the complex plane with a 2 mm iron sphere, a 3 mm aluminium sphere, a mixture of 2 mm iron and 3 mm alu and the addition between the 2 mm iron sphere and 3 mm aluminium sphere.

In Fig. 2.9 the iron and aluminium spheres are plotted in the complex plane, representing the phase and amplitude trajectory as straight lines. In the figure, the measurement with the iron and aluminium spheres fixed together is also plotted. The addition trajectory in the plot, is the manual addition, between the seperately measured aluminium and iron trajectories. The figure shows, that the addition trajectory is coinciding both in phase and amplitude with the mixed measurement. This means, that if two or more metal spheres are present in the detector head, the resultant signal will be a sum of the signals from each sphere. This implies that the signal model can be extended to,

$$u(t) = s_{OoB}(t) + \sum_{m=1}^{M} A_m(t)\sin(\omega t + \phi_m(t)) + n(t) \tag{2.6}$$

where $A_m(t)$ and $\phi_m(t)$ is the amplitude and phase of the m'th sphere of a total of M spheres. The term $n(t)$ is a noise term, which was observed in Fig. 2.7 for the case of a weak signal from the popcorn product. The intensity of the noise term depends on how well the OoB component is

compensated for, and thereby the amplification of the observation signal. The lower boundary for the noise term is evaluated to be the mechanical vibrations in the coils, which is a consequence of the moving conveyor belt. This means that if the OoB component is removed in the metal detector head with metals, then the lower boundary is reached.

2.3.3 Discussion

In this section the amplitude and phase trajectory, of a sphere passing through the detector head, was analysed. The analysis was based on measurements and simulations. From the analysis it can be concluded, that the amplitude of the signal increases until the sphere reaches the first receiving coil, then it starts to decrease again, until it reaches zero in the centre of the transmitting coil. Then it starts to increases again, until the second receiving coil is reached, after which it decreases to zero, as the sphere moves away from the detector head. The phase trajectory of the signal is constant outside the centre of the transmitting coil, where it shifts π radians. It was found that the amplitude and phase trajectory of a package of unpopped popcorn did not follow the trajectory described by the model, because instead of two peaks the unpopped popcorn had four peaks and instead of one phase shift of π radians, the popcorn had three. There cannot be stated any theoretical reason for this phenomenon. The difference in phase and amplitude trajectory between the popcorn package and the metal spheres, is seen as an advantage, because it separates the product and metal objects in terms of the trajectory.

It is found that if the trajectory is viewed in the complex plane, it is represented as straight lines, where the length of the line is determined by the maximum amplitude of the signal, and the angle of the line is determined by the constant phase. The constant phase corresponds to the real and imaginary parts being related through a first order linear regression coefficient.

The section ended up analysing the effects of more than one sphere present in the detector head. It can from this analysis be concluded, that the signal from the receiving coils, if more than one object is present, is a sum of the signals from each object. This lead to the extension of the signal model, which became,

$$u(t) = s_{OoB}(t) + \sum_{m=1}^{M} A_m(t) \sin(\omega t + \phi_m(t)) + n(t)$$

In the next section two food products are defined for simulation, and the amplitudes resulting from these are held against the ones from aluminium and iron. The comparison is performed as a ratio, which is defined as the Product Signal-to-Noise Ratio (SNR_p).

2.4 Product Signal-to-Noise Ratio

This section analyses the Product Signal-to-Noise Ratio (SNR_p), where the signal is the induced voltage in the receiving coils from the foreign metal object, and the noise is the induced voltage due to the food article. The SNR_p is relevant, because it can be shown that this ratio can be maximised by choosing the correct frequency, as discussed in Sec. 2.2. At this optimal frequency, metal detection should be most robust.
The section starts by mathematically analysing the SNR_p, based on the model developed in Cha.

1. This mathematical analysis is followed by an evaluation of the SNR_p, using the four classes of objects described Sec. E.2.1. The four classes cover:

- The Wet Food class covering products with no magnetic properties, but weak conductive proporties, which corresponds to e.g. animal fat and meat.
- The Dry Food class covering products with low magnetic proporties, but no conductive proporties, corresponding to e.g. cereals and white bread.
- The Conductive metal class, corresponding to aluminium, brass, etc.
- The Iron metal class.

The outcome of this evaluation, should lastly be used to obtain an estimate of the optimal frequency by means of the SNR_p, between the metal and product.

2.4.1 SNR_p Based on Model

The SNR_p depends on the position of the center of the metal object, and the center of the food article. To make a general analysis, it is assumed that the center of the foreign metal object, is equal to that of the food article. The superposition principle was shown earlier to be valied, such that the induced voltage in the receiving coil for the metal object and the food article, can be treated separately. The SNR_p is defined as,

$$\text{SNR}_p = 20 \cdot \log_{10} \frac{V_s}{V_f} \tag{2.7}$$

where V_s in the induced voltage due to the metal object and V_f is the induced voltage due to the food article. Using Eq. 1.23 for the induced voltage, an expression for the SNR_p is obtained,

$$\text{SNR}_p = 20 \cdot \log_{10} \frac{\sum_{n=1}^{\infty} \kappa(n) \cdot V_{n,(s)}}{\sum_{n=1}^{\infty} \kappa(n) \cdot V_{n,(f)}} \tag{2.8}$$

where the constant terms, that are independent of n are cancelled, $V_{n,(s)}$ is given in Eq. 1.17 using values for the metal object, $V_{n,(f)}$ is also given in Eq. 1.17 using values for the food article and $\kappa(n)$ is the terms left, that depends on the index n,

$$\begin{aligned} \kappa(n) \quad = \quad & \frac{C^{-n}}{2n(n+1)} \left(\sin \alpha_{r2} P_n^1(\cos \alpha_{r2}) P_n^1(\cos \theta_{r2}) \cdot r_{r2}^{-n-1} \right) \\ & - \frac{C^{-n}}{2n(n+1)} \left(\sin \alpha_{r1} P_n^1(\cos \alpha_{r1}) P_n^1(\cos \theta_{r1}) \cdot r_{r1}^{-n-1} \right) \end{aligned} \tag{2.9}$$

where $P_n^1()$ is the Legendre polynomium, and the terms left are defined geometrically in Fig. 1.6. The expression for the SNR_p, given in Eq. 2.8, is used in the following to evaluate the theoretical SNR_p.

2.4.2 Evaluation of SNR_p

Based on the expression in Eq. 2.8 the SNR_p for a scenario is given, where the size of the food article is 70 % of the opening of the detector head, and the size of the foreign metal object is 5 %. The SNR_p for the wet food class and the iron class is given in part a) of Fig. 2.10, and the

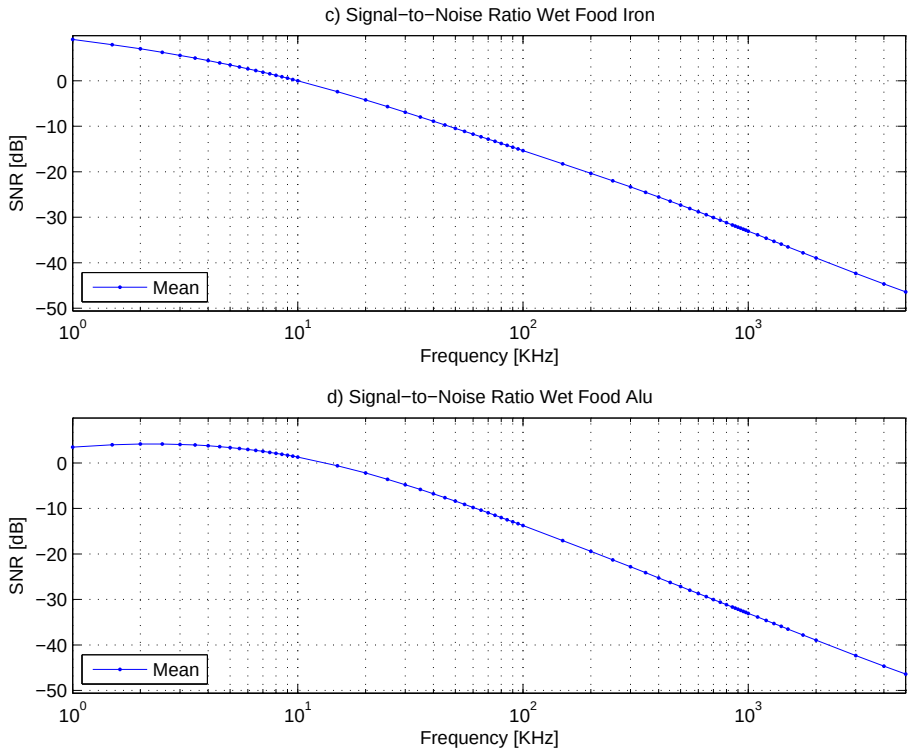

Figure 2.10: Product Signal-to-Noise Ratio based on the mathematical model. Figure a) shows the SNR_p at a given frequency for the wet food class and iron class. Figure b) shows the SNR_p for the class of conductive metal and the wet food class . The size of the food article is 70 % of the opening of the detector head and the size of the foreign metal object is 5 % of the opening.

SNR_p for the wet food class and conductive metal class is given in part b). The SNR_p for the dry food class is given in Fig. 2.11, and related to the metals in the same way. The SNR_p for both comparisons is represented as the mean value of the SNR_p as function of frequency.

In the following two paragraphs the SNR_p of the two different food/noise classes is described. Each noise signal is compared to the metal object, to find the optimal frequency by means of the SNR_p.

The wet food case In Fig. 2.10 part a) the SNR_p by means of the iron class is shown. The figure shows, that the SNR_p decreases, as the frequency is increased. The figure shows, that the optimum frequency is below 1 KHz. It shall be noted, that at 1 KHz, the amplitude of both the wet food class and iron class is small, due to Faradays law of induction. In part b), the SNR_p between the wet food class and the conductive metal class, is plotted. The figure shows, that the optimum frequency is around 2 KHz. The figure shows, that the SNR_p decreases, when the frequency is above and below the optimum frequency. From this it may seem, that the optimal frequency for the wet food class is located at about a few KHz, depending on the specific product.

The dry food case In figure Fig. 2.11 part a) the SNR_p, between the dry food class and the iron class, is shown. The figure shows, that the SNR_p is minimum at approximately 10 KHz. The SNR_p then increases to a (local) maximum at approximately 1 MHz, and then starts to decrease

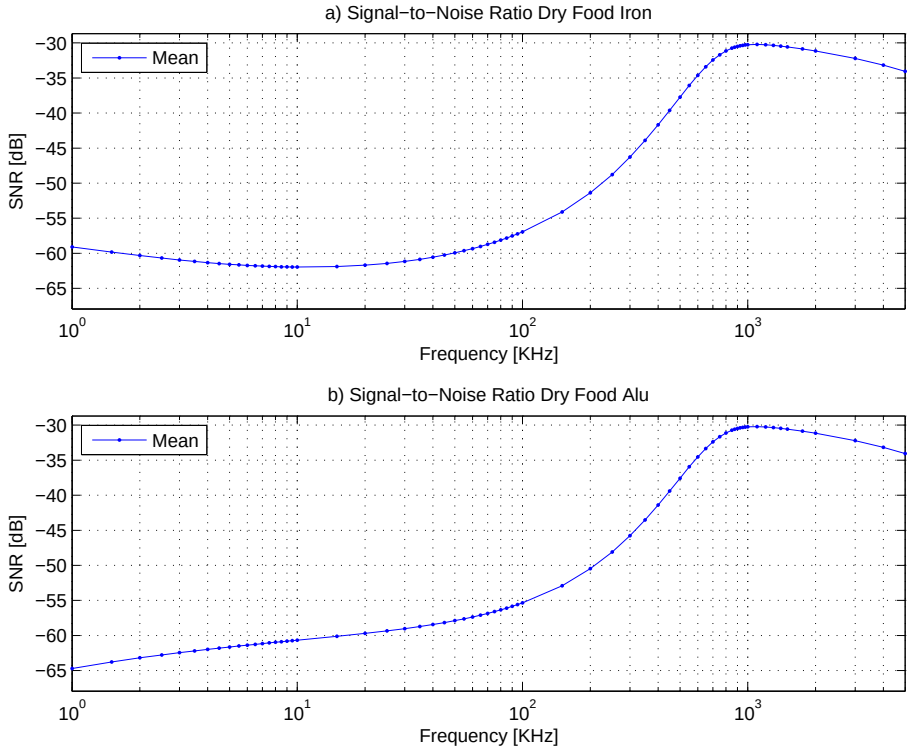

Figure 2.11: Product Signal-to-Noise Ratio based on the mathematical model. Figure a) shows the SNR_p at a given frequency for the dry food class and the iron class. Figure b) shows the SNR_p of the dry food class and the conductive metal class. The size of the food article is 70 % of the opening of the detector head and the size of the foreign metal object is 5 % of the opening.

again. At the maximum the optimum SNR_p is obtained. In part b) the SNR_p between the dry food class and the conductive metal class is shown. The figure shows, that the SNR_p is minimum below 1 KHz, and then increases to a maximum also at approximately 1 MHz, and then starts to decreases again. From this it seems that the optimal frequency, for the dry food class, is around 1 MHz, Which ofcourse depends on the specific product.

2.4.3 Discussion

Fig. 2.10 and Fig. 2.11 show that the optimum frequency depends on the specific product and class of food the foreign object has to be detected in. The optimum frequency for the wet food class is around 1-2 KHz. In Sec. E.2.2 it is shown, that the amplitude of the signals increases with frequency. This means, that the signal amplitude at 1-2 KHz is near the minimum amplitude of the signals. By means of noise from the outside environment, this minimum amplitude is not desirable. For a dry food articles, it is seen that 1 MHz is the optimum frequency according to the SNR_p. The 1 MHz frequency as optimum, is explained in Sec. 2.2 as the local minimum of the dry food class amplitude, which is caused by the food product loosing the magnetic properties.
In Sec. 2.2 it was concluded, that the amplitude of the signal depends on the size of the object. This means, that the SNR_p is increased, when the size of the metal object is increased. Fig. 2.12 and Fig. 2.13 show the change in SNR_p as function of object size.

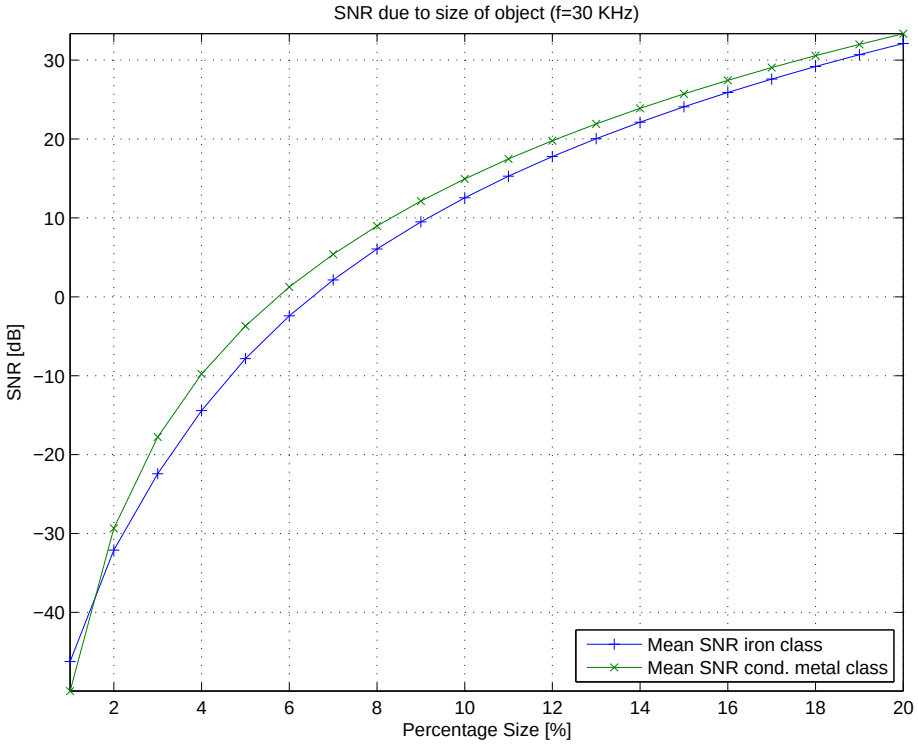

Figure 2.12: SNR_p due to change in foreign metal object size, at the optimum frequency $f = 30$ KHz for the wet food class. The size of magnetic food article is 70 % of the opening of the detector head, and the size of the foreign metal object is varied from 1 to 20 %.

The figures show, that the SNR_p is increased, as the size of the metal object is increased. It can be concluded from the SNR_p simulation, that the optimum frequency depends on the food class, in which metal should be detected. The optimum frequency for the wet food class is at low frequencies, which is not practical, because of the low amplitude values. If the SNR_p for the wet food class is compared to the dry food class, the wet food has a 35 dB better SNR_p under the same conditions in their optimum frequency respectively.

The fourth and last topic of this chapter has now been treated, and in the following conclusions for the chapter are drawn.

2.5 Conclusion on Model of Signals

This chapter has treated the practical part of modelling the detector head. The practical part covers simulation of the effects of objects in the detector head, which are held against measurements from a real metal detector. Based on the measurements it was discovered, that perfect balancing of the coils is impossible, which implied the out-of-balance (OoB) component. After the analysis of the raw signals from the detector head, steady objects were analysed. The steady objects revealed, that the phase and amplitude depends on the material properties of the object. Also the steady analysis revealed, that the amplitude and phase resulting from an object, depends on the transmitting frequency. The following analysis concerned moving objects, in which amplitude and

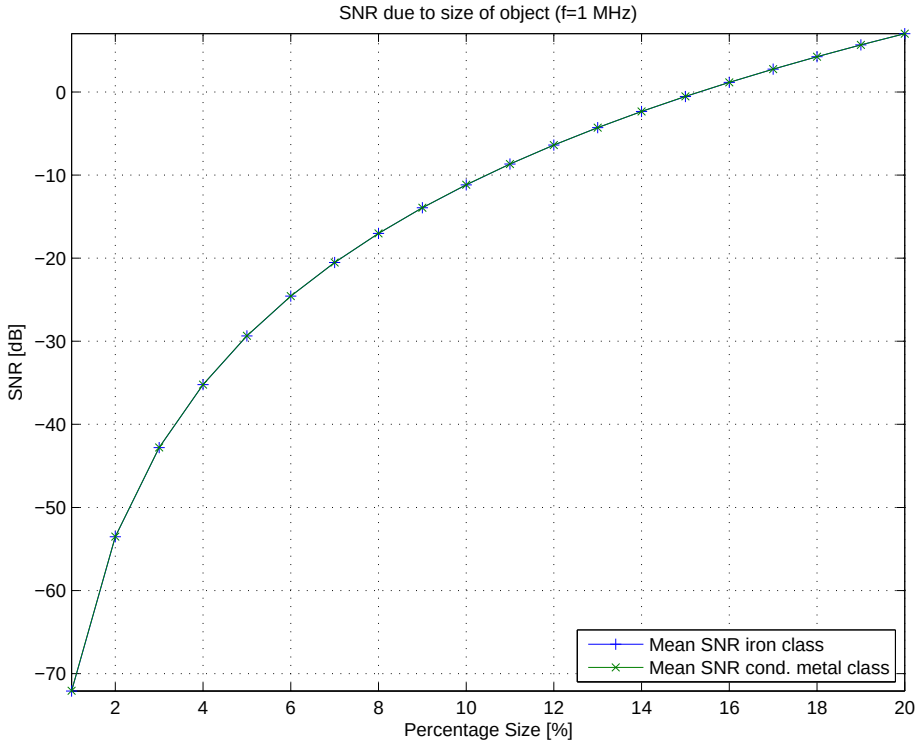

Figure 2.13: SNR_p due to change in foreign metal object size, at the optimum frequency $f = 1$ MHz for the dry food class. The size of magnetic food article is 70 % of the opening of the detector head, and the size of the foreign metal object is varied from 1 to 20 %.

phase trajectories were seen to seperate popcorn from the metal spheres. The moving analysis revealed, that the real and imaginary parts of the trajectory are related through a first order linear regression coefficient. Lastly the Product Signal-to-Noise Ratio was analysed, where the metal object was considered as the signal component, and the food object as the noise component. The analysis showed, that an optimal frequency can be found for a specific food article, resulting in optimal conditions for metal detection. In the end of all four analyses, discussions were supplied, which should be consulted for details.

Throughout the chapter a model of detector head signals has been developed, which in the general case became,

$$u(t) = A_{OoB}\sin(\omega t + \phi_{OoB}) + \sum_{m=1}^{M} A_m(t)\sin(\omega t + \phi_m(t)) + n(t) \tag{2.10}$$

where $A_m(t)$ and $\phi_m(t)$ are the amplitude and phase of the m'th object of a total of M objects present in the detector head. The phase and amplitude is made time dependent, because it was found that they where not constant, but depending on the position of the sphere. The term $n(t)$ is additive random noise from the analog electronics, but also vibrations in the detector head coils. The lower bound for the noise term was concluded as the mechanical vibrations in the coils, which is represented as phase and amplitude noise in the signal component.

Generally it can be concluded that the measurements correlate with the simulated values and

trajectories for spheres. For the case of the aluminium and iron spheres, the phase was seen to be offset approximately six degrees in mean, which was caused by the used material parameters. The estimated amplitude trajectory was seen to equal the simulated one, except from the difference in the maximum value, which may be caused by the removal of the OoB component.
In general it is concluded, that the model describes metal objects in phase and trajectory, but the exact amplitude of the model is not verified due to the amplification introduced before the A/D converter, to be able to represent the objects in the measurements.

In the following chapter, the conclusions from this chapter is used for development of methods for amplitude and phase estimation, together with methods for detection and classification.

Chapter 3

Classification Methods

Until now, analyses concerning the metal detector head and the signals here of, have been performed, which lead to the conclusion, that the amplitude and phase between the reference signal $d[n]$ and the observation signal $u[n]$ are suitable for detection and classification of metals. Also it has been concluded, that the amplitude and phase trajectories can be used. In this chapter these *features* are used for the development of methods for detecting and classifying metals. The chapter is organized according to two elements of pattern recognition, which covers preprocessors and recognition strategies.

The signals resulting from the detector, are both given as,

$$y[n] = A_y \sin(\frac{\omega}{F_s} n + \phi_y) \text{ in the discrete time case} \tag{3.1}$$

$$y(t) = A_y \sin(\omega t + \phi_y) \text{ in the continuous time case} \tag{3.2}$$

where F_s is the sampling frequency, A_y is the amplitude of the sinusoidal signals, and ϕ_y is the phase shift. The notation of using y is used, whenever u and d are not distinguished between, such as derivations of methods, where the signals are treated seperately. The preprocessors convert the sinusoidal signals ($u[n]$ and $d[n]$) to a suitable representation for feature extraction. A suitable representation, for amplitude and phase, is a complex number. The complex number has advantages over the amplitude and phase representation. An advantage is, that wrapping of the unit circle is avoided, when the phase exceeds $\pm\frac{\pi}{2}$, which is a consequence of using the arcus tangent function. The preprocessor should therefore output the amplitude and phase estimates, as a complex number, which is denoted $X[q,k]$. In the representation of $X[q,k]$, k is the digital frequency known *a priori*, and q is the present time instance. At the time instance q, the amplitude ratio and phase difference between the sinusoidal signals are obtained,

$$A[q] = \frac{A_u[q]}{A_d[q]} \tag{3.3}$$

$$\phi[q] = \phi_u[q] - \phi_d[q] \tag{3.4}$$

which is written in the complex form $X[q,k]$ as,

$$X[q,k] = \frac{X_u[q,k]}{X_d[q,k]} = \frac{A_u[q] \cdot (\cos(\phi_u[q]) + j \cdot \sin(\phi_u[q]))}{A_d[q] \cdot (\cos(\phi_d[q]) + j \cdot \sin(\phi_d[q]))} \tag{3.5}$$

The amplitude and phase trajectories are obtained, by observing $X[q,k]$ in a window of length M, corresponding to $X[m,k]$ for $m = [q - M + 1, .., q]$. This corresponds to a Short Time Fourier

Transform of M complex spectra without overlap, in the *a priori* frequency k.

The recognition strategies concern extracting features from the complex number, made by the preprocessor, and then making a decision on grounds of the features in it. The feature extractor converts the complex number to a normalised representation, that can be used in the classifier to decide whether metal is present or not. This is illustrated in Fig. 3.1.

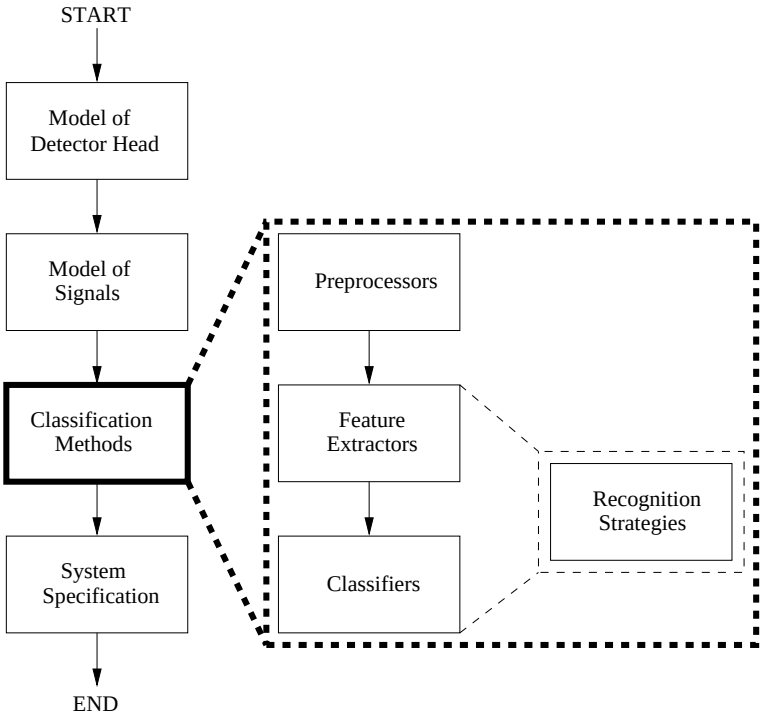

Figure 3.1: Illustration of present state in the design report of this master thesis. Present state is Classification Methods.

The Preprocessors utilize methods for estimation of phase and amplitude of a sinusoid of known frequency. Assuming *a priori* knowledge about the frequency is considered as valid, because it is decided once the metal detection system is being developed and implemented. In practice the consequence of assuming the frequency to be known, is that drift will occur in the seperate amplitude and phase estimates of $u[n]$ and $d[n]$, if the real frequency is not equal to the assumed one. The drift is reflected as amplitude and phase modulation in the seperate amplitude and phase estimates of the signals. However, when the amplitude ratio and phase difference between $u[n]$ and $d[n]$ is considered, the parameter drift disappears, as it is equal in both estimates. It is crucial to note that the frequency of $u[n]$ and $d[n]$ is the exact same, because the source of the signals is common. When considering the amplitude ratio and phase difference, the only problem of not having the exact frequency, is a decrease in the signal-to-noise ratio (SNR). For this to be a problem, a large deviation from the expected frequency is needed. But even in practice it should be possible to obtain a frequency close to the expected one, such that the SNR is maximised.

The Recognition Strategies concern two approaches each grounded in two different hypotheses.

The Product Hypothesis concerns detection of changes in the known food article or product, which indicates that the training phase should be based on the product. The food hypothesis is as follows,

$$\begin{aligned} H_0 &: \text{Food and/or noise is present} \\ H_a &: \text{Something else is present} \end{aligned}$$

The detection approach is based on learning the characteristics of the food product and noise, and then detect if something different is in the metal detector such as metals.

The Metal Hypothesis concerns classification of specific metal types. In Cha. 2 it was concluded that a mix of more objects can be considered as vector addition, corresponding to the superposition principle being valid. Therefore if food is contaminated by metal, the resultant vector is placed differently, than that of the metal itself. Assuming that the signal due to the metal object is the deciding part of the mix, the hypothesis is valid. Otherwise the food product component should be estimated and then removed. The remains from the product removal can then be classified as either noise or metal. These considerations are however not treated in this master thesis, and it is assumed that the metal component in the mix is the deciding factor.
The metal hypothesis approach is a pattern recognition approach, where the feature extractor seeks to find metal objects, instead of rejection of product. The hypothesis is,

$$\begin{aligned} H_0 &: \text{Metal object is present} \\ H_a &: \text{Metal object is not present} \end{aligned}$$

where H_0 states the hypothesis to seek the metal, and H_a is the alternative hypothesis, which states that metal is not present.

With these two hypotheses defined, and the purpose of the preprocessors clarified, the development of methods is initiated. The following section handles the development of methods for the preprocessor, i.e. estimation of amplitude ratio and phase difference between the reference and observation signals, in the complex plane.

3.1 Preprocessors

In the introduction to this chapter it was discussed, that the force of representing the amplitude and phase in the complex plane was to avoid both the tangent function, but also to avoid wrapping of the phase. Another strength in the complex plane, is that the out-of-balance component will be represented as the mean value of $X[q,k]$. Observing $X[q,k]$ in the time-frequency plane, in the known frequency k, at the present time index q, and $M-1$ previous samples of $X[q,k]$, the estimated out-of-balance component becomes,

$$S_{OoB}[q,k] = \frac{1}{M} \sum_{m=q-M+1}^{q} X[m,k] \tag{3.6}$$

where $S_{OoB}[q,k]$ is the Discrete Fourier transformed of s_{OoB}, at time instance q in the time-frequency plane. Eq. 3.6 corresponds to the mean value of the Short Time Fourier Transform in the

a priori frequency k. Digital balancing of the receiving coils can thereby be performed by spectral subtraction of the mean value, which can be done by the expence of only a few computations. Assuming that $S_{OoB}[q-1,k]$ is known, together with the spectrum $X[m,k]$ for $m = \{q, q-M+1\}$, the update step for $S_{OoB}[q,k]$ becomes,

$$S_{OoB}[q,k] = S_{OoB}[q-1,k] + \frac{X[q,k] - X[q-M+1,k]}{M} \tag{3.7}$$

Because the signal frequency k is known, it is left out in derivations in the rest of the chapter, unless it is crucial to include it. In the following, six narrowband methods for amplitude and phase estimation are presented and discussed.

3.1.1 The RGN-GCMA Method

The Recursive Gauss-Newton (RGN) method is a recursive way of tracking the amplitude and phase of a single sinusoidal signal. The method presented here contains an approximation, which minimizes the computational complexity. The approximation is called Gradient Correlation Matrix Approximation (GCMA) and is presented in [Zheng et al., 2007].

Theory

The core of the method is grounded in the definition of an error $e[n]$, and at time n the error is defined as,

$$e[n] = y[n] - \widehat{A}_y[n-1]\sin\Big(\omega n + \widehat{\phi_y}[n-1]\Big) \tag{3.8}$$

where $y[n]$ is the present signal sample in which the phase and amplitude is to be tracked, $\widehat{A}_y$ is the previously estimated amplitude, $\widehat{\phi}_y$ is the previously estimated phase, and ω is the *a priori* frequency. The recursiveness of the method is seen from the time indices, which signifies the use of an estimate calculated by means of $e[n-1]$. The error is recursively minimized by means of Gauss-Newton recursion, such that the desired frequency component in $y[n]$ is tracked by the synthetic sinusoid $\widehat{A}_y[n-1]\sin(\omega n + \widehat{\phi}_y[n-1])$.
The first step in the RGN method is to compute $e[n]$. Then the gradient ψ of $e[n]$ is found with respect to the parameter vector $\widehat{\theta} = [\widehat{A}_y[n]\ \widehat{\phi}_y[n]]^T$,

$$\psi[m] = \frac{\partial e[m]}{\partial \widehat{\theta}} \tag{3.9}$$

The gradient is now used for calculation of the Hessian matrix $\mathbf{H}[n]$,

$$\mathbf{H}[n] = \sum_{m=0}^{n} \lambda^{n-m}\psi[m]\psi^T[m] \tag{3.10}$$

where $0 < \lambda \leq 1$ is the forgetting factor. Instead of calculating $\mathbf{H}[n]$ each time, based on the history from time $n = 0$, it can be updated recursively as,

$$\mathbf{H}[n] = \lambda\mathbf{H}[n-1] + \psi[n]\psi^T[n] \tag{3.11}$$

The update step for the parameter vector $\widehat{\theta}[n]$ is then performed recursively as,

$$\widehat{\theta}[n] = \widehat{\theta}[n-1] - \mathbf{H}^{-1}[n]\psi[n]e[n] \tag{3.12}$$

Using the matrix inversion lemma, see for example [Haykin, 2002], the inverse Hessian matrix can be updated as,

$$\mathbf{H}^{-1}[n] = \frac{1}{\lambda}\left[\mathbf{H}^{-1}[n-1] - \frac{\mathbf{H}^{-1}[n-1]\psi[n]\psi^T[n]\mathbf{H}^{-1}[n-1]}{\lambda + \psi^T[n]\mathbf{H}^{-1}[n-1]\psi[n]}\right] \tag{3.13}$$

The Recursive Gauss-Newton method can be simplified, which is done by the approximation called Gradient Correlation Matrix Approximation (GCMA). The GCMA approximation is grounded in the assumption that the off-diagonal of the gradient correlation matrix $\psi[n]\psi^T[n]$ can be set to zero under two conditions,

- ω must not be either zero or π
- λ should be close to one

The GCMA gives the following update equations for the diagonal elements of the Hessian matrix,

$$H_{11}[n] = \lambda H_{11}[n-1] + \sin^2\left(\omega n + \widehat{\phi}_y[n-1]\right) \tag{3.14}$$

$$H_{22}[n] = \lambda H_{22}[n-1] + \widehat{A}_y^2[n-1]\cos^2\left(\omega n + \widehat{\phi}_y[n-1]\right) \tag{3.15}$$

from which the instantaneous amplitude $\widehat{A}_y$ and phase $\widehat{\phi}_y$ estimates can be found as,

$$\widehat{A}_y[n] = \widehat{A}_y[n-1] + \frac{\sin\left(\omega n + \widehat{\phi}_y[n-1]\right)e[n]}{H_{11}[n]} \tag{3.16}$$

$$\widehat{\phi}_y[n] = \widehat{\phi}_y[n-1] + \widehat{A}_y[n-1]\frac{\cos\left(\omega n + \widehat{\phi}_y[n-1]\right)e[n]}{H_{22}[n]} \tag{3.17}$$

Substituting $y[n]$ by $u[n]$ and $d[n]$ respectively, the complex number $X_n[q]$ representing amplitude ratio and phase difference becomes,

$$X[n] = \frac{\widehat{A}_u[n]}{\widehat{A}_d[n]} \cdot (\cos(\widehat{\phi}_u[n] - \widehat{\phi}_d[n]) + j \cdot \sin(\widehat{\phi}_u[n] - \widehat{\phi}_d[n])) \tag{3.18}$$

which shows, that $X[n]$ is obtained for every time instance n, which is not equal to q for methods operating on blocks of samples from $y[n]$. In the following the RGN-GCMA method is discussed.

Discussion

The RGN-GCMA method differs from the other five amplitude and phase estimation methods, as will be seen, by operating samplewise on the reference and observation signals. The force of operating samplewise, is that the changes in amplitude and phase is tracked "continuously", whereas methods operating on blocks of samples will require more samples per estimate.
The computational complexity is considered as low, because of the simple recursive update of the parameter vector. But taking into consideration the fact, that for each sample a sine and cosine function should be computed, the implementational aspect raises the complexity.

3.1.2 The FAPES Filter

The Forward Amplitude and Phase EStimation (APES) method is a filter design approach for narrowband applications. The source used for this method is [Stoica et al., 1999].

Theory

The design approach is grounded in estimating phase and amplitude in a given frequency from a set of data samples $y[n]$ of length N. This information is then used to obtain a M-tap narrowband FIR filter $h[n]$, which in the least squares sense, passes only the frequency of interest in the data set. The minimization problem becomes,

$$\min_{\mathbf{h},\,\alpha_y} \frac{1}{L}\sum_{l=0}^{L-1}\left|\mathbf{h}^H\mathbf{y}[l]-\alpha_y e^{j\omega l}\right|^2,\ \text{subject to}\ \mathbf{h}^H\mathbf{a}(\omega)=1 \tag{3.19}$$

where the vector $\mathbf{a}(\omega)=[1\ e^{j\omega}\ \ldots\ e^{j(M-1)\omega}]^T$, the complex number α_y is the FAPES approximation of the complex signal spectrum and $L=N-M+1$. The minimization problem in Eq. 3.19 consists of two optimization parameters $\mathbf{h}$ and α_y. The minimization is defined to pass only the frequency of interest such that the subject becomes $\mathbf{h}^H\mathbf{a}(\omega)=1$, meaning that the FAPES filter $\mathbf{h}$ should pass the frequency ω in the complex unity spectrum $\mathbf{a}(\omega)$. The solution to the minimization problem is found in two steps. First the solution relative to α_y is found, and finally the solution to $\mathbf{h}$. The FAPES filter solution is given as,

$$\mathbf{h}=\frac{Q^{-1}(\omega)\mathbf{a}(\omega)}{\mathbf{a}^H(\omega)Q^{-1}(\omega)\mathbf{a}(\omega)} \tag{3.20}$$

where the matrix $Q(\omega)=\frac{1}{L}\sum_{l=0}^{L-1}\mathbf{y}[l]\mathbf{y}^H[l]$. The amplitude and phase estimate can be found from the APES approximation to the complex signal spectrum α_y. The expression for α_y is found from the first solution to the minimization problem in Eq. 3.19 with respect to α_y. This solution is given as,

$$\alpha_y=\mathbf{h}^H\mathbf{g}(\omega) \tag{3.21}$$

where $\mathbf{g}(\omega)=\frac{1}{L}\sum_{l=0}^{L-1}\mathbf{y}[l]e^{-j\omega l}$. The complex spectrum α_y contains the amplitude and phase for $y[n]$. Substituting $y[n]$ by $u[n]$ and $d[n]$ respectively, the complex number representing amplitude ratio and phase difference is obtained as,

$$X[q]=\frac{\alpha_u}{\alpha_d} \tag{3.22}$$

meaning that α should be computed for both $u[n]$ and $d[n]$. In the following the method is discussed.

Discussion

The FAPES method gives the best possible estimate in the least squares sense, based on joint minimization in time ($\mathbf{h}$) and frequency (α). In theory it should be robust according to noise outside the frequency ω of interest, because the noise variance is spread in the complex spectrum, as the focus is put on a single frequency, resulting in a noise variance $\sigma_N^2(\omega)=\frac{\sigma_N^2(t)}{N/2}$, when a N-point DFT is considered.

The drawback of the method is however the computational complexity, which is based on that the FAPES filter needs to be calculated for both $u[n]$ and $d[n]$ to obtain amplitude ratio and mutual phase difference. The matrix inversion of Q in Eq. 3.20 can however be implemented more effient by performing recursive updating through the Matrix Inversion Lemma, see for example [Haykin, 2002].

3.1.3 The LMMSE Filter

The Least Minimum Mean Square Error (LMMSE) Filter is a filter design method for minimizing the error (squared difference) between two signals. The method presented here is based on the Wiener Hopf solution to the problem, see for example [Haykin, 2002].

Theory

The LMMSE Filter method is based on the solution to finding the filter $\mathbf{h}$, which minimizes the error $e[n] = d[n] - \sum_{l=0}^{\infty} h_l u[n-l]$ in the least squares sense. The minimization problem becomes,

$$\min_{\mathbf{h}} E\left[|e[n]|^2\right] = \min_{\mathbf{h}} E\left[\left|d[n] - \sum_{l=0}^{\infty} h_l u[n-l]\right|^2\right] \tag{3.23}$$

The minimization problem in Eq. 3.23 corresponds to minimizing the variance of the error between the reference $d[n]$ and the observation $u[n]$. A step towards the solution of the problem is to take the derivative of $E\left[|e[n]|^2\right]$ with respect to $\mathbf{h}$, and then solve the equation for the filter taps in $\mathbf{h}$, by setting the expression equal to zero. The solution to this least squares minimization problem is called the Wiener Hopf solution, and is given in the matrix formulation as,

$$\mathbf{h} = E\left[\mathbf{u}[n]\mathbf{u}^T[n]\right]^{-1} E\left[\mathbf{u}[n]d[n]\right] \tag{3.24}$$

$$= \mathbf{R}_{uu}^{-1} \cdot \mathbf{p}_{ud} \tag{3.25}$$

where $\mathbf{R}_{uu}^{-1}$ is the autocorrelation matrix of $u[n]$, which is assumed to be invertible, and $\mathbf{p}_{ud}$ is the crosscorrelation vector of $u[n]$ and $d[n]$.

The amplitude and phase estimates can be found by analysis of the optimum filter $\mathbf{h}$ in the *a priori* frequency. This can be done because $\mathbf{h}$ is the filter, which phase shifts and scales the amplitude of $u[n]$, such that $u[n]$ is as equal as possible to $d[n]$ in the least squares sense. A way to analyse $\mathbf{h}$ is to find the transfer function representation $\mathbf{H}[q]$ as the Discrete Fourier Transform of $\mathbf{h}$ in the *a priori* frequency. The amplitude ratio and phase difference estimate becomes $X[q] = \mathbf{H}[q]$. In the following the method is discussed.

Discussion

The LMMSE method estimates the amplitude ratio and phase difference in the least squares sense based on the two sequences $u[n]$ and $d[n]$ directly. Performing analysis on the optimum filter by means of DFT results in robustness for noise outside the desired frequency, as the noise variance is spread in the spectrum.
In Eq. 3.25 it is seen that the method requires inversion of the autocorrelation matrix $\mathbf{R}_{uu}$. As discussed about the FAPES method, the computational complexity of the matrix inversion can be reduced by recursively updating it.
Another weakness is that the minimization of the error is based on the time domain alone, meaning that the optimal filter $\mathbf{h}$ minimizes the error in mean for all frequencies in the signals. Compared to the FAPES and the RGN-GCMA methods, which utilize the *a priori* frequency in the core, this method is not expected to perform as well as the two others.

3.1.4 The Crosscorrelation Method

The Crosscorrelation method is based on the crosscorrelation between the two signals of interest. The reason for trying the approach is to obtain noise reduction through the crosscorrelation function under the assumption of uncorrelated noise.

Theory

The method is based on estimating the crosscorrelation function. The definition of the reference and observations signals is here given as,

$$d(t) = A_d \cdot \sin(\omega t) + N_d(t) \tag{3.26}$$
$$u(t) = A_u \cdot \sin(\omega t - \phi) + N_u(t) \tag{3.27}$$

Where ϕ is the phase difference, and the terms $N_d(t)$ and $N_u(t)$ are stochastic white noise sequences. The crosscorrelation between $d(t)$ and $u(t)$ is expressed as,

$$\begin{aligned}
R_{ud}(\tau) &= E\left[u(t)d(t+\tau)\right] && (3.28)\\
&= E\left[(A_d \cdot \sin(\omega t) + N_d(t))(A_u \cdot \sin(\omega(t+\tau) - \phi) + N_u(t+\tau))\right] && (3.29)\\
&= E\left[(A_d \cdot \sin(\omega t))(A_u \cdot \sin(\omega(t+\tau) - \phi))\right] + E\left[(A_d \cdot \sin(\omega t))N_u(t+\tau)\right] && (3.30)\\
&\quad + E\left[A_d \cdot \sin(\omega(t+\tau) - \phi)N_d(t)\right] + E\left[N_d(t)N_u(t+\tau)\right] && (3.31)\\
&= A_d A_u E\left[(\sin(\omega t))(\sin(\omega(t+\tau) - \phi))\right] && (3.32)\\
&= \frac{A_d A_u}{2} E\left[\cos(\omega t - (\omega(t+\tau) - \phi)) + (\cos(\omega t + \omega(t+\tau) - \phi))\right] && (3.33)\\
&= \frac{A_d A_u}{2} E\left[\cos(-\omega\tau - \phi))\right] && (3.34)\\
&= \frac{A_d A_u}{2} \cos(-\omega\tau - \phi)) && (3.35)
\end{aligned}$$

Eq. 3.35 shows that the crosscorrelation between $d(t)$ and $u(t)$ is a phase shifted cosine, which has amplitude $\frac{A_d A_u}{2}$ and is periodic in $\frac{1}{\omega}$. To find the phase shift between $d(t)$ and $u(t)$, the τ which maximises $R_{ud}(\tau)$ has to be found. The reason for finding the τ, that maximises Eq. 3.35 is,

$$\begin{aligned}
\frac{A_d A_u}{2} \cos(0) &= \frac{A_d A_u}{2} && (3.36)\\
&\Downarrow && (3.37)\\
0 &= -\omega\tau_{max} - \phi && (3.38)\\
\phi &= -\omega\tau_{max} && (3.39)
\end{aligned}$$

where ϕ is the phase shift between the $u(t)$ and $d(t)$, and τ_{max} is the delay in samples, which maximises the crosscorrelation function. The term τ_{max} has to be found in the interval $[0 : \frac{1}{\omega}]$ due to the periodicy of the crosscorrelation function.

The amplitude of the crosscorrelation function at τ_{max}, is in Eq. 3.36 expressed as $\frac{A_d A_u}{2}$. The amplitude of the autocorrelation of $d(t)$ is,

$$\begin{aligned}
R_{dd}(0) &= E\left[d(t)d(t)\right] && (3.40)\\
&= E\left[(A_d \cdot \sin(\omega t) + N_d(t))^2\right] && (3.41)
\end{aligned}$$

$$= A_d^2 E\left[\sin^2(\omega t)\right] + E\left[N_d(t)^2\right] \quad (3.42)$$

$$= \frac{A_d^2}{2} + \sigma_{N_d}^2 \quad (3.43)$$

where $\sigma_{N_d}^2$ is the noise variance. If $\sigma_{N_d}^2 << \frac{A_d^2}{2}$, $\sigma_{N_d}^2$ can be removed from Eq. 3.43. The gain between $d(t)$ and $u(t)$ can then be expressed as,

$$A = \frac{R_{ud}(\tau_{max})}{R_{dd}(0)} \quad (3.44)$$

$$= \frac{\frac{A_d A_u}{2}}{\frac{A_d^2}{2}} \quad (3.45)$$

$$= \frac{A_u}{A_d} \quad (3.46)$$

The complex number $X[q]$ is obtained from A and ϕ. In the next section the method is discussed.

Discussion

The crosscorrelation method was described in continuous time. If the method is moved to the discrete time, the phase estimation depends on the over sampling rate, because the sample space of τ_{max} is limited by the over sampling rate. The sample space for ϕ by means of the over sampling rate is expressed as $[0 : \frac{2\pi}{T_{over}} : 2\pi]$, which means that if the over sampling rate $\frac{1}{T_{over}}$ is large, the sample space is also large. If the over sample rate equals one (the Nyquist criterion), which implies that $\frac{1}{T_{over}}$ equals 2, the sample space contain 2 samples. This means that the method can only choose one of the samples in the sample space. The sample space can however be expanded by interpolation.
The gain estimation part of the estimation method is based on the assumption that $\sigma_{N_d}^2 << \frac{A_d^2}{2}$. If the assumption does not hold, the gain will be underestimated because of the noise term in the denominator,

$$A = \frac{\frac{A_d A_u}{2}}{\frac{A_d^2}{2} + \sigma_{N_d}^2} \quad (3.47)$$

It is concluded that the denominator of Eq. 3.47 will always be over estimated, but the degree of error depends on the signal-to-noise ratio.

3.1.5 The DFT Method

The Discrete Fourier Transform (DFT) method is based on spectrum analysis. The reason for trying the spectrum analysis is to obtain noise reduction, by knowing the *a priori* frequency. By noise reduction is meant, that white noise present in the signal is spread across the whole spectrum, when the signal is transformed and the signal power is focused in one frequency bin.

Theory

The Fourier transformation is in the discrete domain expressed as,

$$Y[q] = \sum_{n=0}^{N-1} y[n] e^{-\frac{2\cdot\pi}{N}\cdot k\cdot n} \quad (3.48)$$

where $Y[q]$ is the Discrete Fourier transformed at the discrete a priori frequency k, N is the sequence length, and $y[n]$ is the discrete time sequence representing $u[n]$ and $d[n]$ respectively. Due to the *a priori* knowledge about the transmitting frequency, k is known as $k = \frac{N}{F_s} \cdot f_t$, where F_s is the sample frequency, and f_t is the transmitting frequency known beforehand. Eq. 3.48 can be rewritten as,

$$Y[q] = \sum_{n=0}^{N-1} y[n] e^{-\frac{2 \cdot \pi}{FS} \cdot f_t \cdot n} \tag{3.49}$$

The spectrum $Y[q]$ is expressed in the complex form as,

$$Y[q] = \frac{N \cdot A_y}{2} e^{j\phi_y} \tag{3.50}$$

where A_y is the amplitude of the signal $y[n]$ at frequency f_t, and ϕ_y is the phase of the signal. To find the amplitude ratio and phase shift between $u[n]$ and $d[n]$, their spectra are divided. Substituting $y[n]$, by $u[n]$ and $d[n]$ respectively, gives,

$$\begin{aligned} X[q] &= \frac{U[q]}{D[q]} && (3.51) \\ &= \frac{\frac{N \cdot A_u}{2} e^{j\phi_u}}{\frac{N \cdot A_d}{2} e^{j\phi_d}} && (3.52) \\ &= \frac{A_u}{A_d} \cdot e^{j(\phi_u - \phi_d)} && (3.53) \end{aligned}$$

where the complex number $X[q]$ represents the gain and phase between $u[n]$ and $d[n]$. In the next section the method is discussed.

Discussion

The DFT method utilises the *a priori* frequency to suppress noise outside the frequency of interest. The computational complexity is low, as the transform consists of multiplications and sums, if it is assumed that the complex exponential is located in a lookup table.
If noise is present, the spectra are represented as,

$$U[q] = \frac{N \cdot A_u}{2} e^{j\phi_u} + \frac{A_{N_u}}{2} e^{j\phi_{N_u}} \tag{3.54}$$

$$D[q] = \frac{N \cdot A_d}{2} e^{j\phi_d} + \frac{A_{N_d}}{2} e^{j\phi_{N_d}} \tag{3.55}$$

where A_{N_u} and A_{N_d} are the amplitudes of the noise sequences, and ϕ_{N_u} and ϕ_{N_d} are the noise phases. The equations show that N is not multiplied to the noise term, due to the fact that the energy is spread in the whole spectrum. The noise terms are added to the signal terms, which means that the noise terms does not affect the phase and gain estimation, as long $A_{N_u} << N \cdot A_u$ and $A_{N_d} << N \cdot A_d$ is satisfied.

3.1.6 The LSDFT Method

The Least Squares DFT (LSDFT) method is an approach based on the DFT method described in Sec. 3.1.5, but tries to minimise the phase and amplitude estimation error in a least mean squares sense.

Theory

The method is based on the spectral transfer function,

$$U[q] = X[q] \cdot D[q] \tag{3.56}$$

where $U[q]$ is the frequency spectrum of $u[n]$, $D[q]$ is the frequency spectrum of $d[n]$, and $X[q]$ is the complex number, which transforms $D[q]$ into $U[q]$. The complex number $X[q]$ can be expressed as,

$$X[q] = Ae^{j\phi} \tag{3.57}$$

where A is the amplitude ratio between $u[n]$ and $d[n]$, and ϕ is the phase difference between them. From Eq. 3.56 an equation for the estimation error $e[q]$ can be derived,

$$e[q] = U[q] - X[q] \cdot D[q] \tag{3.58}$$

The estimation error in Eq. 3.58 equals zero if only one spectral pair of $D[q]$ and $U[q]$ are present. If $D[q]$ and $U[q]$ are extended to vectors of length M, containing spectral pairs of $d[n]$ and $u[n]$ respectively the minimisation problem can be expressed as,

$$e^2[1] = E\left[(U[m] - X[q] \cdot D[m])^2\right], \quad m = [q - M + 1, ..., q] \tag{3.59}$$

The problem seeks to find the complex number $X[q]$, which minimises the squared estimation error $e^2[q]$. The complex number $X[q]$ can be calculated by taking the derivative of Eq. 3.59 with respect to X,

$$\frac{de^2[q]}{dX} = E\left[-2(U[m] - X[q] \cdot D[m])D[m]\right], \quad m = [q - M + 1, ..., q] \tag{3.60}$$

To find the $X[q]$ which minimises Eq. 3.59 in least squares sense, Eq. 3.60 is set equal to zero, which is the same as finding the minimum of $e^2[q]$,

$$0 = E\left[-2(U[m] - X[q] \cdot D[m])D[m]\right] \tag{3.61}$$

$$0 = -2E\left[U[m] \cdot D[m]\right] + 2X[q]E\left[D[m] \cdot D[m]\right] \tag{3.62}$$

$$X[q] = \frac{E\left[U[m] \cdot D[m]\right]}{E\left[D[m] \cdot D[m]\right]} \tag{3.63}$$

for $m = [q - M + 1, ..., q]$. In the next section the method is discussed

Discussion

The LSDFT method is based on estimating the amplitude and phase in the least squares sense for a set of complex estimates, instead of focusing on a single. The strength of the method is that noise present in the *a priori* frequency is suppressed because of the minimization problem, but also noise outside the known frequency. The suppression of noise is based on the noise sequences in $u[n]$ and $d[n]$ being uncorrelated, and if this is not true the complex number X becomes biased.
The drawback of the method is that the ability to track the amplitude ratio and phase difference becomes limited, as minimizing the error for several complex estimates, corresponds to averaging. The method may therefore be more suited for estimating the out-of-balance component, which is stationary and can therefore be considered as the mean value in the spectrum. Estimating the out-of-balance component in the least squares sense may improve the performance of simply computing the mean value.

3.1.7 Evaluation of Methods

In this section the amplitude and phase estimation methods are evaluated. The evaluation is based on simulations, where the performance of the methods are tested under noisy conditions. Each method is exposed to the same test signals, and the amplitude and phase are estimated by each method.
The simulation is performed a 1000 times for each test signal, and the percentage error, together with mean and standard deviation, is obtained. The reason for obtaining the means and standard deviations of the simulations, is to be able to find the method, which is performing best, with least deviation in the mean estimate.
The signal-to-noise ratio (SNR) is defined in the time domain as,

$$\mathrm{SNR} = \frac{\sigma_Y^2}{\sigma_N^2} \tag{3.64}$$

where σ_Y^2 is the variance of the signal component, and σ_N^2 is the variance of the noise component. As most of the amplitude and phase estimation methods are based on the *a priori* frequency, the frequency domain SNR can be seen as significantly better as in the time domain, because the noise power is spread across the whole spectrum. Therefore the methods are evaluated for SNRs in the interval $\{-20 : +28\}$ dB.
The results of the simulations, concerning the amplitude estimation, are shown in Fig. 3.2 a) and b). The results concerning the phase estimation, are shown in Fig. 3.3 a) and b). Of both figures, part a) shows the mean estimation error, and part b) shows the standard deviation of the estimation error.
In the following section the performance of the methods will be discussed, and afterwards the computational complexity of the three best performing methods is evaluated.

Discussion of Estimation Performance

The mean amplitude estimation error is shown Fig. 3.2 a). The figure shows, that the methods perform differently in mean, under noisy conditions. In Tab. 3.1 the SNR, where the mean estimation error exceeds 1%, is shown.

Method	SNR [dB]
FAPES	-11
DFT	-11
RGN-GCMA	-9
LSDFT	2.5
LMMSE	6
XCORR	20

Table 3.1: Sorted list of methods, showing the SNR, where the *mean amplitude estimation error* exceeds 1%. The 5% limit for the standard deviation on the estimates, is for the methods approximately -1 dB, except from the XCORR method.

The methods are sorted in the table best mean estimation performance, according to the SNR. The FAPES and DFT methods, are those with the best performance in mean, and they exceed the 1%

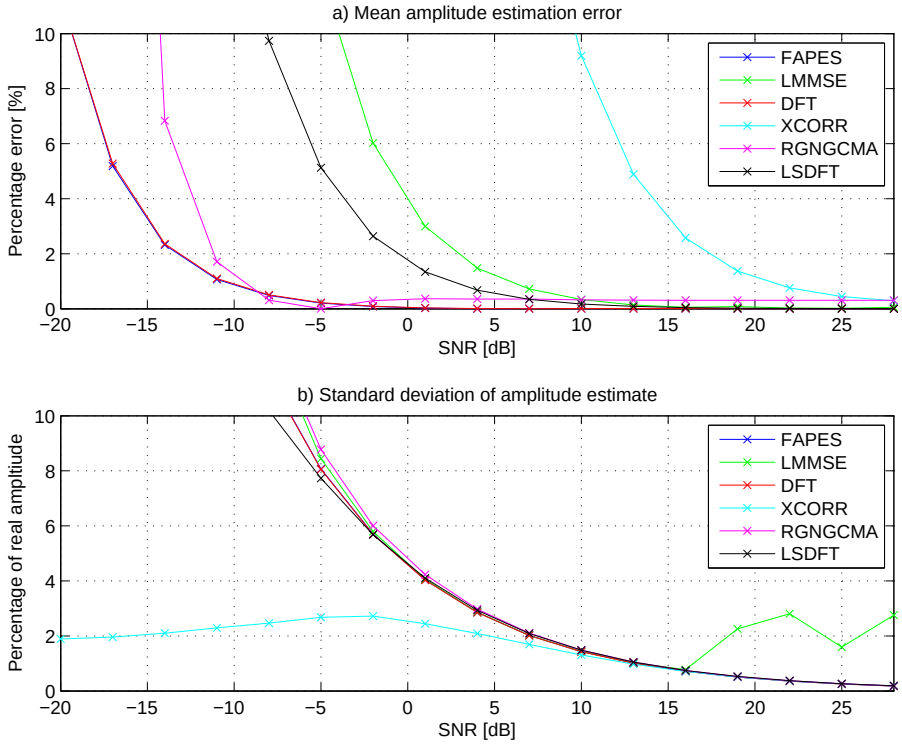

Figure 3.2: Simulation results concerning the amplitude estimation error. Part a) shows the mean estimation error, and b) shows the standard deviation of the error.

limit at -11 dB SNR. These two methods were also among those expected to perform best, because they are based on the *a priori* frequency is the core. The two best performing methods, are close followed by the RGN-GCMA method, which exceeds the limit at -9 dB. The RGN-GCMA method is based on parameter tracking in the time domain, but the method utilises the forgetness factor to limit the influence of noise. The three remaining methods, are all above 0 dB SNR, which is significantly different from the three best. The LSDFT method was expected to perform better, because it is also based on the *a priori* knowledge about the frequency. The LMMSE and XCORR methods, are based on the time domain, so they were also expected to perform worse, than the three frequency domain methods, due to the noise power being spread in the frequency spectrum. Especially the XCORR method, which is due to the bias in the amplitude estimate described Sec. 3.1.4.
Including the standard deviation in the discussion, attention is paid to Fig. 3.2 b). The figure shows, that the standard deviations for the methods increases, when the SNR is decreased. It is seen, that the methods are more or less identical, except from the XCORR method. Intuitively, the standard deviation for the method, is interpreted as outstanding, but the mean estimate is useless below approximately 10 dB. The other methods have a 1 % deviation from the correct amplitude at 15 dB.

In Fig. 3.3 a) the mean percentage error of the phase estimation is shown. The figure shows, that five of the six methods have an estimation error below 1 % at -14 dB SNR, but that the XCORR method fails below -2 dB. A threshold of 1 % is set, as with the amplitude results, and

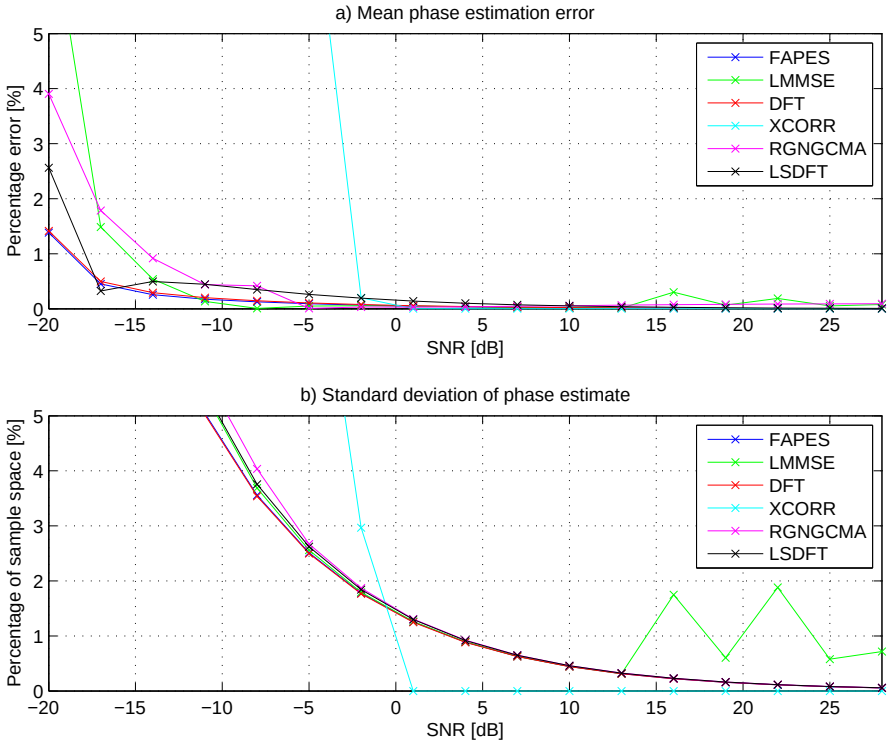

Figure 3.3: Simulation results concerning the phase estimation error. Part a) shows the mean estimation error, and b) shows the standard deviation of the error.

in Tab. 3.2 a sorted list of the methods is shown, together with the standard deviation at the 1% limit.

Method	SNR [dB]
FAPES	-18.7
DFT	-18.6
LSDFT	-17
LMMSE	-15.5
RGN-GCMA	-14
XCORR	-2

Table 3.2: Sorted list of methods, with the SNR, where the *mean phase estimation error* exceeds a 1% limit. The 5% limit for the standard deviation on the estimates, is for the methods approximately -10 dB, except from the XCORR method.

Again the methods are sorted according to the SNR, but this time, where they exceed the 1% error from the correct phase. The FAPES and DFT methods are again the best performing methods, with a SNR below -18 dB. This time the LSDFT is the third best, with a mean estimate close to the two best methods. The SNR for the LMMSE method, where the mean phase estimation error exceeds 1%, is this time lower, than that for the RGN-GCMA method, but the XCORR method is again the worst performing. The spread of the results for the mean phase estimation, is not as

significant as the one for the mean amplitude estimation.
The standard deviation of the phase estimation error, increases as the SNR decreases, which is seen from Fig. 3.3 b). The standard deviations for the methods are similar, except for the XCORR method. The reason for the performance of the XCORR method, is that the sample space is 10, meaning that the phase can only take on ten different values. This is also seen in Fig. 3.3 b), where results for XCORR method is "zig-zagging". The SNR, where 5% standard deviation is exceeded, is for the phase estimates approximately -10 dB, which is 9 dB lower, than the one for the amplitude estimates. The deviations on the estimates of the amplitude is larger than those of the phase estimation, which means that the methods in general produce better phase estimates, than amplitude estimates. For high SNRs the methods perform with low error in phase and amplitude estimation, but when the SNR is decreased, the standard deviation in the estimates starts to increase, which makes the estimates less precise.

Based on the amplitude and phase estimation performance, three methods are discarded and the remaining are analysed for computational complexity. The XCORR method is the first to be discarded, as this method is the worst in performance of both amplitude and phase estimation. Based on the phase estimation performance, the five remaining methods perform with only a difference of approximately 4.5 dB. Based on the amplitude estimation performance, the last two methods for elimination can be chosen. The LMMSE and LSDFT methods are the second and third least performing methods, why the choice of discarding these is clear. Thereby, the remaining three are the FAPES, DFT and RGN-GCMA methods, which are analysed for computational complexity in the following.

Computational Complexity

The method, which is most suited as the preprocessor for the classification system, cannot be chosen among the three remaining methods, when the amplitude and phase estimation performance is considered alone. Therefore the computational complexity is considered, which is based on the number of multiplications and summations for a block of a 1000 samples. Also mathematical functions (e.g. sin / cos) in the methods are taken into consideration, because multiplications and summations do not include the implementational complexity of these. In Tab. 3.3 the number of multiplications and summations are given for the three remaining methods.

Method	Additions	Multiplication
FAPES	198922	179082
RGN-GCMA	10000	10000
DFT	1998	2000

Table 3.3: Computational complexity of the amplitude and phase estimation methods, in terms of multiplications and summations.

In Tab. 3.3 the complexity of the three methods is shown. The complexity calculations are based on a block of a 1000 samples. The table shows that the FAPES method is between approximately 20 and a 100 times more demanding than the RGN-GCMA and DFT methods. The reason for this high degree of complexity, is that the FAPES method is based on matrix multiplications and summations. The difference between the RGN-GCMA and DFT method is approximately a factor

of five. The RGN-GCMA method is a samplewise method, where the DFT method is a block based method, which gives the DFT method an advantage in comparison. If the number of estimates is taken into consideration, instead of the number of samples, the RGN-GCMA does in fact give a 1000 estimates, where the DFT methods gives only one. If this is considered, the computational complexity of the RGN-GCMA method is lower, than that of the DFT method. Based on the number of the multiplications and summations the FAPES method is discarded.
The discussion of the computational complexity is now extended to include the implementational cost of the two remaining methods. The implementational cost covers mathematical functions. The DFT method is based on multiplying the signal component by the complex exponential, and these products are accumulated. The complex exponential consists of a sine and a cosine function. These functions can be computed beforehand, and put into a lookup table, and be read sequentially, which reduces the control path of the lookup table. The RGN-GCMA method also requires a sine and a cosine, but also the squared functions. The function can be located in a lookup table, but cannot be read sequentially, as the function values should be looked up according to a specific argument. This increases the implementational cost of the control path, when compared to the DFT method. Based on the discussion of the computational complexity, the DFT method is chosen as the preprocessor for the classification system. The DFT method was shown to to be among the three best performing amplitude and phase estimation methods, and also one of the least demanding in computational complexity, including the implementational aspects. If another candidate should be chosen for further analysis, according to a specific hardware architecture, the RGN-GCMA mathod should be chosen. This method performs nearly as well as the DFT method, and is also cheap in implementation. This is however not done because of time constraints put on the project.

The preprocessor for the classification system has been determined, based on amplitude and phase estimation performance, together with the computational complexity. In the next section two recognition approaches are developed, each based on different hypotheses.

3.2 Recognition Strategies

In this section two recognition strategies (including feature extractor and classifier) are developed. The two strategies are based on the Product and Metal Hypotheses respectively, defined in the beginning of this chapter. The feature extractor has the purpose of extracting a (statically) normalised feature, which can be feed to the classifier, and be used for evaluating the given hypothesis. The section is parted in two, where the first concerns the Food hypothesis and the second the Metal hypothesis.

3.2.1 Product Hypothesis Approach

The product hypothesis should be based on detection in which the food product is known, which signifies that training is based on learning the characteristics of the product. The characteristics are reflected through the amplitude, phase, and their respective trajectories, caused by a moving object in the detector head.
By making a statistically robust detection method, which is capable of determining whether the product is contaminated or not, the product hypothesis should be realizable. In Cha. 2 it was determined that the trajectory for the phase is a constant value, except from a change of $\frac{\pi}{2}$ Rad in

the middle of the detector head. In the complex plane, this corresponds to the real and imaginary parts, being related through a first order linear regression coefficient, which was also shown. The feature extractor for the product hypothesis is therefore implemented through a first order linear regression model. The classification is then based on the residual error, meaning the deviation from the model. The residual error is assumed to be normally distributed, which is utilised in the classifier. When the distribution is known, each residual error resulting from a new complex estimate $X[q]$ can be held against a defined confidence interval. The classifier for the food hypothesis is thereby made by testing each residual error against the hypothesis, whether the confidence interval is exceeded or not. The significance level of testing is determined from the number of complex estimates, that is needed to represent the complete trajectory, when an object is moving through the metal detector head. If the confidence interval is based on fewer estimates, it is exceeded each time a product is moved through the head, because at least one estimate will exceed the limit.
In the following the linear regression feature extractor is realised.

Linear Regression Feature Extractor

The idea for the Linear Regression (LR) feature extractor is based on the mathematical detector head model, and the analysis here of, see Cha. 1 and Cha. 2 respectively. The first order linear model relating the two random variables $X_R[q]$ and $X_I[q]$ is given as,

$$X_I[q] = \beta + \alpha X_R[q] + \epsilon[q] \tag{3.65}$$

in which α and β are the model parameters, and $\epsilon[q]$ signifies the residual error introduced by modelling the dependence between the two variables. More generally, Eq. 3.65 can be expressed for a set of M values of $X[q]$ as,

$$\sum_{m=1}^{M} X_I[m] = M\beta + \alpha \sum_{m=1}^{M} X_R[m] + \sum_{m=1}^{M} \epsilon[m] \tag{3.66}$$

The residual error term is found by rearranging Eq. 3.66,

$$\sum_{m=1}^{M} \epsilon[m] = \sum_{m=1}^{M} X_I[m] - (M\beta + \alpha \sum_{m=1}^{M} X_R[m]) \tag{3.67}$$

$$= \sum_{m=1}^{M} (X_I[m] - (\beta + \alpha X_R[m])) \tag{3.68}$$

The residual error ϵ is the interesting term to analyse, because it can be used to test, whether a *sample* of $X_R[q]$ and $X_I[q]$ belongs to a predefined model or not. This predefined model is based on *a priori* knowledge. In Cha. 2, it was shown that moving food articles and metal pieces through the metal detector head had a very deterministic nature. The deterministic behaviour revealed itself in the phase and amplitude of the induced voltage, resulting from the object, and this can be exploited for training the model. Using the trained model, the residual error can be found and analysed for the purpose of classifying the complex spectrum.
To analyse the residual error statistically a new term SS_R called the sum of squared residual errors is defined,

$$SS_R = \sum_{m=1}^{M} \epsilon[m]^2 \tag{3.69}$$

$$= \sum_{m=1}^{M} \left(X_I[m] - (\beta + \alpha X_R[m])\right)^2 \tag{3.70}$$

The term SS_R is according to [Ross, 2004] an unbiased estimator of the variance σ^2, through the following relation,

$$\sigma^2 = E\left[\frac{SS_R}{M-2}\right] \tag{3.71}$$

which is a consequence of $\frac{SS_R}{\sigma^2}$ being χ^2_{M-2}-distributed with $M-2$ degrees of fredom. The distribution of SS_R is χ^2_{M-2}-distributed due to the sum of squared normal random variables (the residual errors), with SSR assumed to be independent of X. This follows from the fact that the error term ϵ is assumed independent of X, which can be substantiated from the mathematical model. The degrees of freedom is $M-2$, because two dependent variables are present. The dependent variables are the real and imaginary parts. To perform classification by use of the linear model, the residual error between the complex spectrum and the model has to be normalized. Thereby hypothesis testing can be performed at a given significance level. The *standardized residual error* $\widehat{\epsilon}[q]$ is given as,

$$\widehat{\epsilon}[q] = \frac{X_I[q] - (\beta + \alpha X_R[q])}{\sqrt{\frac{SS_R}{M-2}}} \tag{3.72}$$

The standardized residual error $\widehat{\epsilon}[q]$ is assumed to be an independent standard normal random variable,

$$\widehat{\epsilon}[q] \sim N(0,1) \tag{3.73}$$

which can be used directly for hypothesis testing.

The standardization term $\sqrt{\frac{SS_R}{M-2}}$, together with the model parameters α and β, are found in the training phase for the food article, but it can also be found for metal, if specific classification is needed. The least squares model parameters are found from a data set as follows,

$$S_{X_R X_I} = \sum_{m=1}^{M} (X_R[m] - \overline{X}_R)(X_I[m] - \overline{X}_I) \tag{3.74}$$

$$S_{X_R X_R} = \sum_{m=1}^{M} (X_R[m] - \overline{X}_R)^2 \tag{3.75}$$

$$\alpha = \frac{S_{X_R X_I}}{S_{X_R X_R}} \tag{3.76}$$

$$\beta = \overline{X}_I - \alpha \overline{X}_R \tag{3.77}$$

where $\overline{X}_R$ and $\overline{X}_I$ signifies their respective mean value. For each sample $X[q]$ the standardized residual error is found and compared to the test statistic z for a standard normal distribution.

The LR feature extractor so far is solely based on information about the phase through the α parameter, and product variance $\frac{SS_R}{M-2}$. The feature extractor can be extended to include amplitude detection as well, which reduces the length of the decision region from $[-\infty : +\infty]$ to

$[-\max(|X|) : +\max(|X|)]$, the last being the maximum absolute value of the amplitude of X. In other words, regression is performed to a vector of finite length, instead of a straight line of infinite length. The modified LR feature extractor is called LR with Amplitude Detection (LRAD) feature extractor.

Confidence Interval Classifier

The simplest classifier approach is hypothesis testing based on comparing statistically standardized values, with a limit for a predefined distribution, at a given significance level. The significance level determines the propability, of which a given sample is guaranteed to be within the population. Testing the standardized residual errors $\widehat{\epsilon}[q]$ (assumed normal distributed) at a significance level of κ gives the following decision rule,

$$\mathrm{H}_0 \quad : \quad |\widehat{\epsilon}[q]| \leq z_{\kappa/2} \tag{3.78}$$

$$\mathrm{H}_a \quad : \quad |\widehat{\epsilon}[q]| > z_{\kappa/2} \tag{3.79}$$

where H_0 is the null-hypothesis and H_a is the alternative hypothesis. If $|\widehat{\epsilon}[q]|$ is smaller than $z_{\kappa/2}$ the Null-hypothesis is accepted, meaning that sample $X[q]$ can be classified as belonging to the class under test.

LRAD Training

Training of the LRAD feature extractor is done through the parameters α, β, $\sqrt{\frac{SS_R}{M-2}}$ and $\max(|X|)$. Training can be performed by estimating the absolute value of the noise floor, and then use the absolute value of $X[q]$, that exceed this limit. The noise floor limit, call it λ, is used as follows,

$$\text{Use } X[q] \text{ for training if } |X[q]| > \lambda, \text{ else } X[q] \text{ belongs to the noise class}$$

It is ofcourse essential, that the product sent through the metal detector is a pure product, that is not contaminated with e.g. metal. In Fig. 3.4 the decision region is seen for unpopped microwave popcorn, using a confidence interval of 99 %. Training is performed by use of 2048 values exceeding the noise floor threshold λ of the training data. The trajectory for the product is represented by approximately 250 complex estimates $X[q]$, resulting in approximately ten products used for training of the parameters. The 99 % confidence interval is intuitively expected to be exceeded, because it corresponds to one out of a 100 estimates exceeding the limit. However, because of the use of the noise floor threshold, fewer values are included for training, and the values used are the largest ones, which gives an increase in the standardization term $\sqrt{\frac{SS_R}{M-2}}$ (estimate of the standard deviation). Also a first order recursive lowpass filter is used on the residual errors, to minimize the number of false alarms. A false alarm means, that a clean product is detected as a contaminated one. However, because the number of false alarms is minimised by the lowpass filter, the propability for a miss is maximised. The time constant for the first order lowpass filter is set to 0.85.
In Fig. 3.4 a) clean product is sent through the LRAD classifier, and tested against the confidence interval. The boundary of the confidence interval is marked with green and the noise class with cyan. The product marked with blue, is seen to be placed inside the confidence interval, which signifies that the H_0 hypothesis is accepted. In Fig. 3.4 b) contaminated product is illustrated. The contaminated product is seen to exceed the confidence interval, which is marked with red. Whenever an object results in exceeding the confidence interval, the H_a hypothesis is accepted. When the

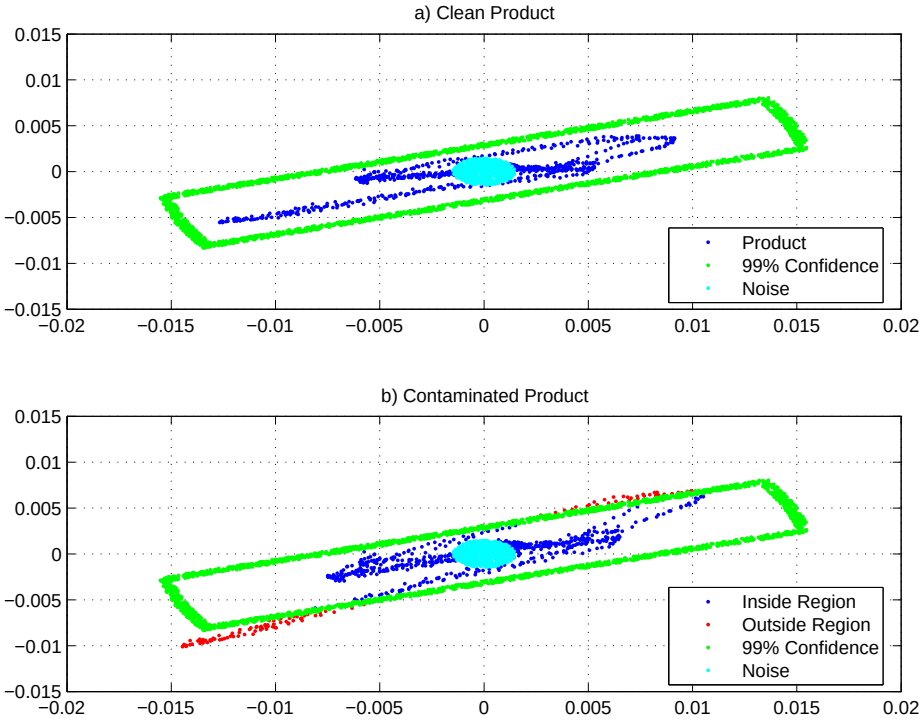

Figure 3.4: Illustration of decision region for LRAD classifier. Part a) shows clean product, and part b) shows contaminated product. The green boundary indicates a 99 % confidence interval.

alternative hypothesis is accepted, the object inside the metal detector head is considered as metal or contaminated product.

3.2.2 Metal Hypothesis Approach

The metal hypothesis is based on classification, where the metals are known beforehand. As discussed in the beginning to this chapter, the approach is expected to perform well, when the metal object is the dominating component in a mix with a food product.
The approach is based on the amplitude and phase trajectory in the complex plane, corresponding to the real and imaginary part of $X[q]$, representing the amplitude ratio and phase difference between $u[n]$ and $d[n]$. The trajectories were in Cha. 2 concluded as suited for detection and classification of metals. Observing $X[q]$ in a window of length M, corresponds to the Short Time Fourier Transform (STFT). The idea for the method is to utilise the trajectory, which was concluded as feasible for metal detection in Cha. 2. The trajectory can be seen as a deterministic signal, which is identical for all the metal types, except for amplitudes of the real and imaginary parts, respectively. The trajectory for the product was seen to be different, which can be utilized for seeking the metal. The idea is to obtain a *matched filter* for the deterministic signal, resulting from the trejectory of the metal. The covariance can then be used to obtain the degree of correlation, between the STFT of $X[m]$, $m = [q - M + 1, .., q]$, and the deterministic matched filter R. The Bayes Decision Rules (Maximum Likelihood), can then be used for deciding, whether the covariance fits the metal class or not. The following section describes the covariance based feature extractor.

Covariance Based Feature Extractor

The covariance based feature extractor is based on Schwartz inequality, which is defined as,

$$(E\left[RX[m]\right])^2 \leq E\left[R^2\right] \cdot E\left[X^2[m]\right], \quad m = [q - M + 1, .., q] \tag{3.80}$$

where $X[m]$ and R are random variables. The inequality states, that the squared cross-covariance of two random processes will be less than or equal to the product of the variances of the random processes. This means, that the cross-covariance of the signals is less than or equal to the product of the standard deviations, which can be defined as,

$$Cov_{RX} \leq \sigma_R \cdot \sigma_X \tag{3.81}$$

where Cov_{RX} is the cross-covariance between R and $X[m]$, and σ_R and σ_X are the standard deviation of R and X respectively. Eq. 3.81 can be used to compare two signals, because the cross covariance describes the correlation between two signals. The signal, which is received when metal objects are present, is known *a priori*, due to the deterministic behaviour of the signal. The covariance feature extractor is described as,

$$Y[q] = Cov_{RX}[q] \tag{3.82}$$

where $Y[q]$ is the output from the covariance function, $X[q]$ is stochastic input, and R is a known deterministic signal of length M, representing a metal object. Eq. 3.82 can be rewritten to,

$$Y[q] = E[R(X[m] - \mu_X)], \quad m = [q - M + 1, .., q] \tag{3.83}$$

$$Y[q] = \sum_{m=0}^{M-1} (R[M - m - 1] \cdot (X[q - m] - \mu_X)) \tag{3.84}$$

where it is assumed that R is zero mean. The sequence R has length M, and holds the deterministic signal. To be able to perform the calculation of covariance described in Eq. 3.84, the formulation has to be rewritten to complex numbers, because the input from the preprocessor is complex. Eq. 3.84 can be rewritten as,

$$Y[q] = E[R^*(X[m] - \mu_X)], \quad m = [q - M + 1, .., q] \tag{3.85}$$

$$Y[q] = \sum_{m=0}^{M-1} (R[M - m - 1]^* \cdot (X[q - m] - \mu_X)) \tag{3.86}$$

where * denotes complex conjugation. Eq. 3.84 and Eq. 3.86 are nearly similar, the only difference is that Eq. 3.84 is the case for real numbers, and Eq. 3.86 is the case for complex numbers, but the inequality Eq. 3.81 still holds. To be able to interpret the outcome of the covariance function, a general example is setup, where two signals are expressed as complex exponentials. The example is expressed in polar coordinates as,

$$Y[q] = \sum_{m=0}^{M-1} (|R[M - m - 1]|\, e^{-j\phi_R} \cdot |X[q - m]|\, e^{j\phi_X}) \tag{3.87}$$

$$Y[q] = e^{j(-\phi_R + \phi_X)} \sum_{m=0}^{M-1} (|R[M - m - 1]| \cdot |X[q - m]|) \tag{3.88}$$

where it is assumed that $X[m]$ is zero mean, and the phase of the signals does not change in a window of M samples. Eq. 3.88 shows that the phase of the covariance function is determined by the phase difference between $X[m]$ and R. The amplitude of the signal is determined by the cross covariance between the amplitude of R and $X[m]$. If a signal is a perfect match to R, the outcome of the covariance function will be places on the real axis, and be maximum, because the two signals have the same phase, and the cross-correlation between the amplitudes is maximum (+1) due to Eq. 3.81. If $X[m]$ is $-R$, the output will also be placed on the real axis, but the cross correlation between the amplitudes will be minimum (-1), due to the sign on R. If $X[m]$ is a white noise sequence, the correlation between the amplitudes will equal zero, which means that the outcome is independent of phase. If two signals correlate in amplitude, but not in phase, the output will be phase shifted $-\phi_r + \phi_x$. From these examples, it can be concluded, that if a signal pattern described by R, has to be extracted, the probability for R to be present in $X[m]$, increases as more positive real the output of the covariance function gets.

The amplitude of the output depends on the variance of $X[m]$ and R, which makes the output hard to compare in terms of classification. To overcome this problem Eq. 3.81 is rewritten to,

$$\left|\frac{Cov_{XY}}{\sigma_X \cdot \sigma_Y}\right| \quad \leq \quad 1 \tag{3.89}$$

which give rise to,

$$Y[q] \quad = \quad \frac{1}{\sigma_R \cdot \sigma_X \cdot M} \cdot \sum_{m=0}^{M-1} (R[M-m-1]^* \cdot (X[q-m] - \mu_X)) \tag{3.90}$$

The division with the standard deviations of $X[m]$ and R, normalises the output of the covariance function. The sample space for the normalised cross covariance is $|[-1;1]|$, which leads to the conclusion, that if the cross covariance is 1, $X[m]$ is a perfect match to R. If the cross covariance is $|1|$, but not only real, the amplitude pattern is a perfect match, but the phase does not match. This conclusion should be kept in mind in the classifications steps. In the next section the classifier is described.

Classification By Use of Bayes Decision Rules

The classifier has the purpose of implementing the hypothesis described in Sec. 3, based on the features from the covariance feature extractor. The classification is based on Bayes decision rules, or Maximum Likelihood estimation, which is a method that minimises the detection error, which means that the error of making a false alarm or miss is minimised. The method is based on Bayes formula,

$$P(C_i|Y) = \frac{P(Y|C_i) \cdot P(C_i)}{P(Y)} \tag{3.91}$$

where $P(C_i|Y)$ is the probability that $Y = \{y_1, y_2\}$ belongs to the class C_i, and $P(Y|C_i)$ is the probability that C_i belongs to Y, $P(C_i)$ is the probability of the class C_i occurring, and $P(Y)$ is the probability of Y. Eq. 3.91 describes the probability that Y belongs to the class C_i. To make the decision between the classes, the probability of Y, belonging to the different I classes, has to

be observed. Y has to be assigned to the class with the highest probability, which is the same as maximising the likelihood. The decision rule can be expressed as,

$$\begin{aligned} \text{if } P(C_1|Y) &> \{P(C_2|Y), ..., P(C_I|Y)\} \\ \text{then} \quad Y &\quad \text{belongs to the class } C_1 \end{aligned}$$

The likelihood can be expressed as a discriminant function $d_i(Y)$,

$$d_i(Y) = P(C_i|Y) = \frac{P(Y|C_i) \cdot P(C_i)}{P(Y)} \tag{3.92}$$

$$\tag{3.93}$$

where the discriminant function describes the probability that Y belongs to the class i. If the *a priori* probability $P(C_i)$ is equal for all I classes, the discriminant functions can be rewritten to,

$$d_i^*(Y) = P(C_i|Y) = P(Y|C_i)$$

because $\frac{P(C_i)}{P(Y)}$ can be interpreted as equal scale constants, which is applied to all I discriminant functions. It shall be noted, that the function $d_i^*(Y)$ does not describe the probability, because of the removal of the scale constant $\frac{P(C_i)}{P(Y)}$, but it does not influence the decision rule. To find the distribution of Y, and thereby the probability density $P(Y|C_i)$, Y is split into a noise term RN and a signal term RS. $Y[q]$ is then defined as:

$$\begin{aligned} Y[q] &= E\left[R(S[m] + N[m])\right] \\ Y[q] &= E\left[RS[m]\right] + E\left[RN[m]\right] \end{aligned} \tag{3.94}$$

for $m = [q - M + 1, .., q]$. The distribution of the noise term is assumed to be normal, due to the central limit theorem, which states that a sum of independent identical distributed random variables, will converges to a normal distribution, as the sum moves towards infinity. The independent noise sub terms, which sum up to the normal distribution, are e.g. uniformly distributed due to fact that some are assumed to origin from roundoff/truncation noise. The distribution of $Y[q]$ is assumed to be normal distributed, where the mean is defined by the deterministic term $E\left[RS[m]\right]$, and the variance is defined by $E\left[RN[m]\right]$. The probability $P(Y|C_i)$, due to the normal distribution, is described as [Shanmugan and Breipohl, 1988],

$$P(Y|C_i) = \frac{1}{(2 \cdot \pi)^{I/2} \sqrt{|\sum_{C_i}|}} e^{(-\frac{1}{2}(Y-\mu_{C_i})^T \Sigma_{C_i}^{-1}(Y-\mu_{C_i}))} \tag{3.95}$$

where Σ_{C_i} is the covariance matrix, belonging to the class C_i, and μ_{C_i} is the mean value of the feature Y, belonging to the class C_i. The probability that Y belongs to C_i, is uniquely described by Σ_{C_i} and μ_{C_i}, because the covariance matrix and mean are the only differences in the discriminant functions for the classes. The covariance matrix and mean for the classes, have to be known *a priori* to perform the classification. The mean estimation is defined as,

$$\mu_{C_i} = \begin{bmatrix} E\left[x_{i1}\right] \\ E\left[x_{i2}\right] \end{bmatrix} = \sum_{j=1}^{J} \begin{bmatrix} \frac{X_{i1j}}{J} \\ \frac{X_{i2j}}{J} \end{bmatrix} \tag{3.96}$$

where x_{inj} is a sample belonging to the class i, feature n and sample j of the set of samples belonging to i. The covariance matrix Σ estimator is described as,

$$\begin{aligned}\Sigma_i &= \begin{bmatrix} E[x_{i1}x_{i1}] & E[x_{i1}x_{i2}] \\ E[x_{i2}x_{i1}] & E[x_{i2}x_{i2}] \end{bmatrix} \\ &= \sum_{j=1}^{J} \begin{bmatrix} \frac{(x_{i1j}-\mu_{i1})^2}{J} & \frac{(x_{i1j}-\mu_{i1})(x_{i2j}-\mu_{i2})}{J} \\ \frac{(x_{i2j}-\mu_{i2})(x_{i1j}-\mu_{i1})}{J} & \frac{(x_{i1j}-\mu_{i1})^2}{J} \end{bmatrix}\end{aligned} \tag{3.97}$$

To implement the hypothesis by use of the Bayes decision rule, two discriminant functions representing the noise and metal objects respectively has to be used. The decision rule implementing the hypothesis is expressed as,

$$\begin{aligned} H_0 &: \text{Metal object present } \Rightarrow d^*_{metal}(Y[q]) > d^*_{noise}(Y[q]) \\ H_a &: \text{Metal object not present } \Rightarrow d^*_{metal}(Y[q]) < d^*_{noise}(Y[q]) \end{aligned}$$

In the next section the estimation and grouping of classes are discussed.

Training

Training of the classifier is basically estimation of the mean vector in Eq. 3.96, and the covariance matrix Eq. 3.97. The estimation is based on a set of data for each class, which has to be known *a priori* to estimate the classifier parameters. The performance of the classifier is dependent of the training sets, because the classifier adapt to the data sets, and assume that the data sets, describe the total population of the class. In Fig. 3.5 different types of metal and sizes after the feature extractor are plotted.
The clouds in Fig. 3.5 represent data sets used for training. The selection of the metal data sets, are based on the *a priori* knowledge of the correlation between the matched signal and the signals from the different metal types, which means that the supervised learning seeks to find the features, which absolute values are approximately equal to 1. The absolute values chosen in the plot are the values above 0.9. In Sec. 3.2.2 the theory is based on two discriminant functions, one describing the noise class, and one describing the metal class. If the metal datasets are grouped into one data set, and the noise/product in another set, the type error has to be observed, because those errors describe the probability of choosing metal, when noise/product is present(false alarm), and choosing noise/product, when metal is present (miss). The type errors are defined as:

- False alarm
  The probability mass of the noise/product class, when the metal class is chosen.
- Miss
  The Probability mass of the metal class, when the noise/product class is chosen.

The type errors has to be minimised, because it makes the classifier more robust in choosing the correct class. In Tab. 3.4 the type errors of different groupings of the metal class are shown.
The errors show, that one metal class (d_{metal}) containing all types of metal has a miss error of 13 %, and a false alarm error of 4.5 %. The errors are relative larger, compared to the errors, when the metal class is subdivided into a FE class, and an ALU/AISI class, because the errors are approximately zero. If the ALU/AISI class is further subdivided into a separate ALU and

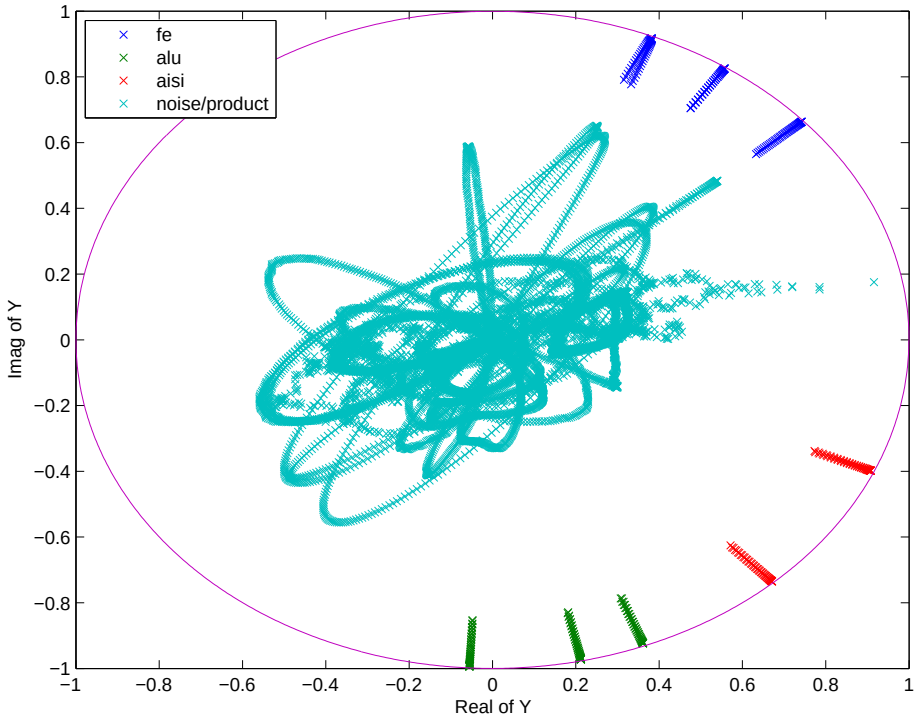

Figure 3.5: Plot of feature space with observations from the metal and noise/product classes.

Grouping strategy	False Alarm	Miss
d_{noise}, d_{metal}	0.0490	0.1350
$d_{noise}, d_{FE}, d_{ALU\ AISI}$	$9.4495 \cdot 10^{-5}$	$1.3662 \cdot 10^{-5}$, $1.3591 \cdot 10^{-5}$
$d_{noise}, d_{FE}, d_{ALU}, d_{AISI}$	$7.7334 \cdot 10^{-5}$	$1.3662 \cdot 10^{-5}$, $2.2346 \cdot 10^{-8}$, $2.7673 \cdot 10^{-8}$

Table 3.4: The table shows the dependency between the grouping of class and type errors.

AISI classes, the errors are further lowered, but is assumed to be insignificant due to the errors close to zero. The errors show, that the classifier has to be trained for two metal classes, and one noise/product class, which imply that the hypothesis is rewritten to:

$$H_0 \quad : \quad \text{Metal object present} \ \Rightarrow \{d^*_{FE}(Y[q]), d^*_{ALU\ AISI}(Y[q])\} > d^*_{NOISE/PRODUCT}(Y[q])$$
$$H_a \quad : \quad \text{Metal object not present} \ \Rightarrow \{d^*_{FE}(Y[q]), d^*_{ALU\ AISI}(Y[q])\} < d^*_{NOISE/PRODUCT}(Y[q])$$

In Fig. 3.6 the decision regions based on the training, where the AISI data set is grouped with the ALU data set. The lines represent the limit, where the classifier shifts from choosing noise/product to metal. The figure shows, that the FE class and ALU/AISI class do not have any intersection, which means that the two classes are mutual exclusive. The mutual exclusion means that the classes are independent, and represent the metal class as two separate subclasses, which are independent of each other.
The decision regions also separate the observations used for training perfectly, which also should be possible, because the noise and metal classes do not have any intersection.

The Product and Metal hypotheses have now been developed and discussed. In the following section, the DFT preprocessor and both recognition strategies are evaluated.

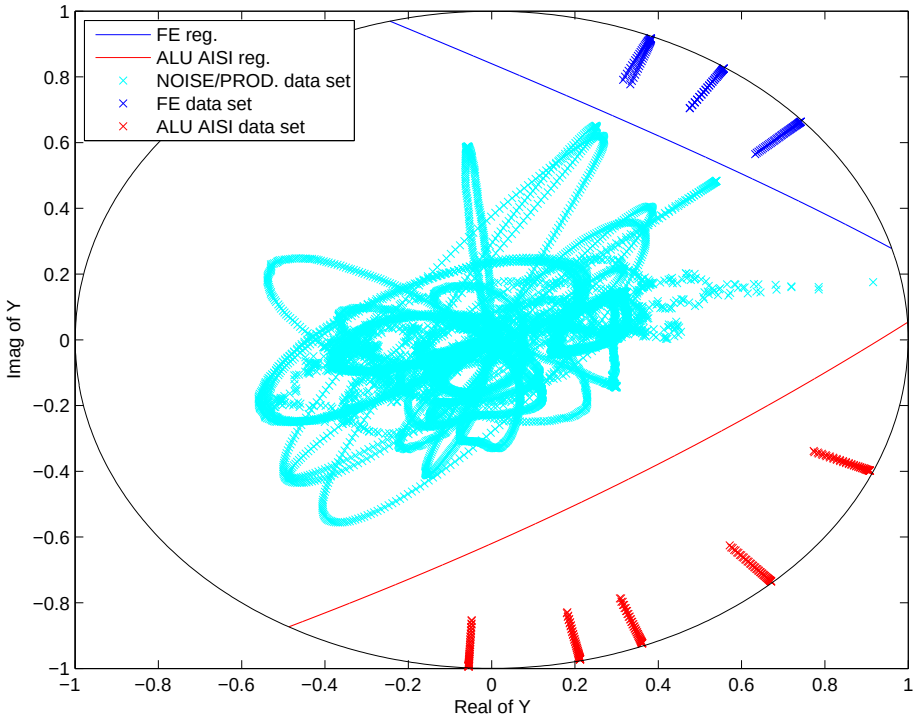

Figure 3.6: Plot of feature space with observations from the metal and noise/product classes. The lines represent the boundary between the decision regions.

3.3 Evaluation of Recognition Strategies

This section evaluates the classification methods. The evaluation is based on showing the functionality of LRAD classifier, and the classification performance of the Covariance/Bayes classifier. The reason for not showing the performance of the LRAD classifier, is that the method depends on the spread of the residual error of the popcorn, which cannot be expected to be the same for all measurements measured *at different days*. The differences lie in the amplification of the signal before the A/D conversion, which was not possible to keep fixed, because two student groups have worked with the same detector. In a signal analysis point of view, the signal shall be amplified as much as possible, to get the best representation of the signals. This amplification "distorts" the amplitude of the signal in a classification point of view, because the LRAD depends on the amplitude. The Covariance/Bayes classifier does not depend on the amplitude, which makes it possible to test this classifier with the different signals. In the first section the evaluation of the functionality of the LRAD classifier will be presented.

3.3.1 LRAD Classifier

The LRAD classifier is based on classification through "level detection" of residual errors. In this section, two scenarios are setup, to show the functionality of the classifier. The first scenario, is where the classifier is trained with five packaged of popcorn measurements, measured at the same day, with the same amplification. After the training, 34 other popcorn measurements from the same day, with the same amplification, is used for test. Both the training and test signals are lowpass filtered to reduce the phase noise. The wanted result, is that the classifier should not detect any

of the other packages. To visualise the result, the residual errors as function of time is plotted in Fig. 3.7.

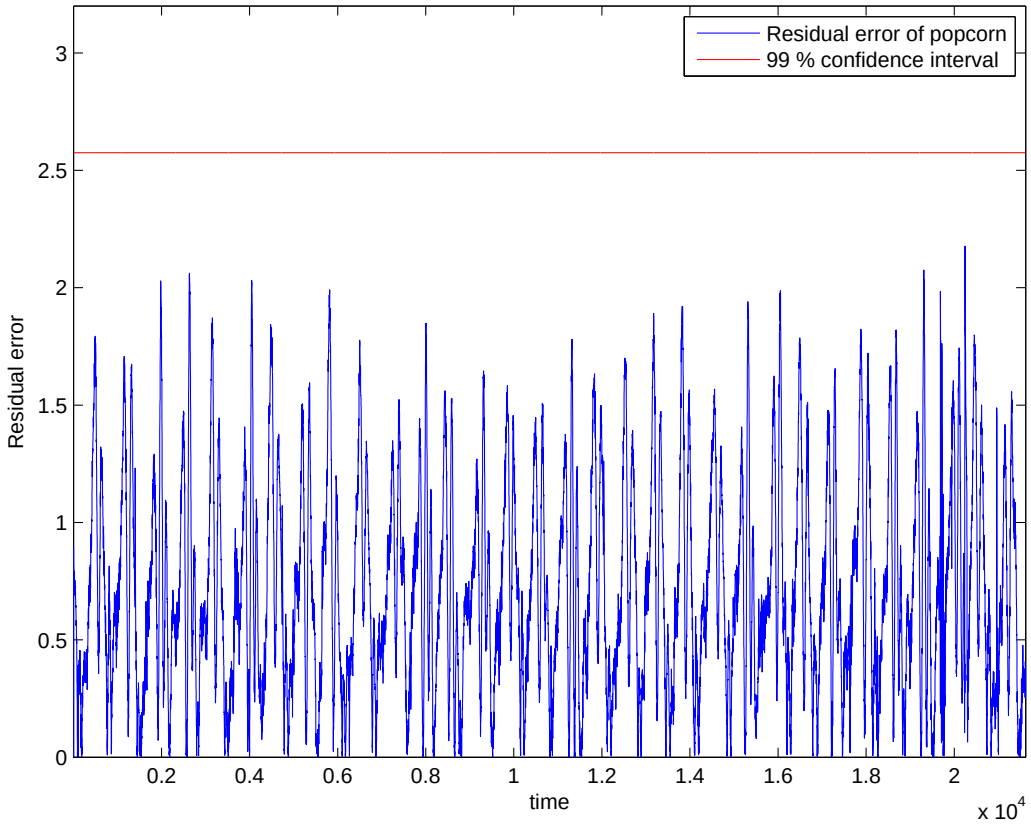

Figure 3.7: Residual errors for 34 popcorn measurements.

The figure shows that the 34 popcorn measurements do not exceed the decision threshold, which was expected. It can however be seen, that effect of the lowpass filter is a reduction in the residual errors. If the 99% confidence interval is lowered to be just above the largest residual error, the detection of metal will also be increased. It can be concluded, that the method does not classify popcorn as metal.
The next case is where a metal object is present, together with the popcorn. The residual error is observed again, and eight measurements from the same measurement series is used. The results are shown in Fig. 3.8.
The figure shows eight metal spheres, where the spheres are aligned as.

1. 1 mm iron sphere
2. 2 mm iron sphere
3. 5 mm iron sphere
4. 2 mm aluminium sphere
5. 3 mm aluminium sphere
6. 5 mm aluminium sphere

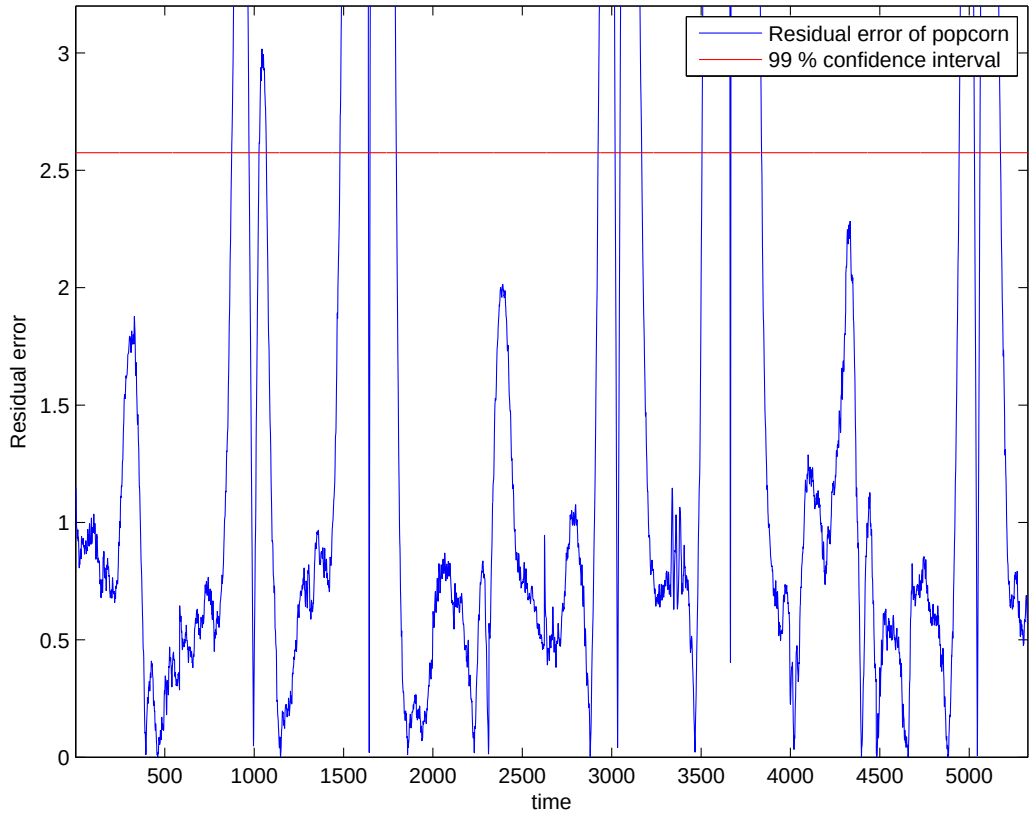

Figure 3.8: Residual error of metal objects

7. 3 mm aisi sphere

8. 5 mm aisi sphere

The the figure shows, that the method cannot detect the small metal spheres, because they do not exceed the decision threshold. It is respectively the 1 mm iron sphere, 2 mm aluminium, and 3 and 5 mm AISI spheres. The reason for this is due to the lowpass filter, which reduces the sensitivity. To detect the smaller spheres, the confidence interval should be lowered. The reason for this, is that the amplitudes of the small spheres, are relative weak compared to the popcorn, so the *change* in the product is small.

3.3.2 The Covariance/Bayes Classifier

In this section the Covariance/Bayes classifier is evaluated. The evaluation is performed with all measurements available, because it does not depend on the amplitude of the signal. The evaluation is performed by shipping the test measurements to the classifier, and then observe the result at the output. The training data is not used for evaluation.
The results are shown in Tab. 3.3.2. The results show that method detects all pure iron and aluminium spheres, and the mixtures between aluminium and popcorn. It is seen that the classifier wrongly detects 9 % of the popcorn, which does not correlated with the computations of miss and false, in the description of the classifier. The performance of the classifier degrades, when the metal

TYPE	Detection ratio	Number of measurements
FE	100 %	19
ALU	100 %	20
AISI	84 %	19
POPCORN	91 %	34
FE POPCORN	91 %	12
ALU POPCORN	100 %	10
AISI POPCORN	50 %	10

objects are mixed with the popcorn. This was also expected, due the additive mixture between the trajectory of the metal sphere and the popcorn.

3.4 Conclusion on Classification Methods

This chapter started by deriving and discussing amplitude and phase estimation methods for the preprocessor. Afterwards the methods where simulated under noisy conditions. The simulations showed, that the methods performed nearly equal in the phase estimation, but the methods differed in amplitude estimation. Due to the simulation of the amplitude estimation the FAPES, DFT and RGNGCMA methods were compared by means of their computational complexity. The comparison showed that the FAPES method where more computational demanding than the other two methods, without any increase in the precision of the amplitude and phase estimates. Among the two remaining, the DFT method was chosen for the preprocessor, due to the lower implementational cost.
After the choice of preprocessor, two recognition strategies were developed, based on the Food and the Metal Hypotheses respectively. The method developed for the food hypothesis was based on the residual error due to linear regression of the complex number X. The method for the metal hypothesis was based on normalised covariance through the idea of matched filters, and classification based on Bayes Decision Rule. In the evaluation of the methods it was found that the bayes and covariance classifier detect all iron and aluminium spheres, which not was the case in the functionality test of the LRAD classifier. It was discussed that the LRAD classifier maybe could improve performance, when the sphere is mix with popcorn, this was not tested. Both methods are chosen for implementation, such that the two hypotheses can be compared in performance.

The following and last chapter in the design report covers specification of the metal detection system, based on the analyses and developed methods in the design report.

Chapter 4

System Specification

Through the first part of this report, specifications for the metal detection system have been found through analyses of the detector head and the signals here from. The analyses have covered both simulations and measurements. Also methods for metal detection and classification have been developed and requirements for these have been estimated through simulations in MATLAB. The system specification is the last topic of the analysis report as depicted in Fig. 4.1.

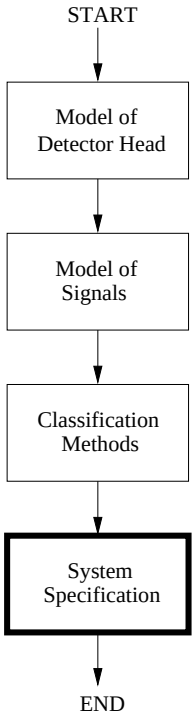

Figure 4.1: Present stage in the overall flow of the first part, is marked with a thick box. Present and last stage is System Specification

This chapter consists of three sections concerning the metal detection system including interfaces and requirements. The first section presents a block diagram for the metal detection system and specifies interfaces in it. The second section presents the requirements for the system. The chapter is concluded by discussing practical aspects of the choices made so far in the report, and gives an indication of how the practical system shall be implemented.

4.1 Interfaces

This section presents an overview of the system, based on the first part the report, together with the interface between the metal detection system and the "analog world". In Cha. 1 it was found that two signals origin from the metal detector head, and methods for extracting features of those were developed in Cha. 3.
In Cha. 1 it was seen that a differential amplifier is needed to obtain the observation signal $u(t)$ from the receiving coils. Moreover amplification is needed for the signal to fit the input of an A/D converter, which is necessary to perform digital signal processing. The amplification should also be performed for the reference signal from the transmitting coil. Thereby two A/D converters are needed to process the two signals.
When the two signals have been sampled in the A/D converter, they are feed to the classification methods described in Cha. 3. The signals are considered as input, and the decision, whether metal is detected, is considered as the output. The output signal can be used for e.g. triggering a mechanism, that removes the contaminated product from the conveyor belt or sets an alarm. It is also crucial to keep track of where the contaminated product is on the conveyor belt, such that the contaminated product can be removed. But this is not considered in this master thesis.
An overview of the overall metal detection system, and its interfaces, is presented in Fig. 4.2.

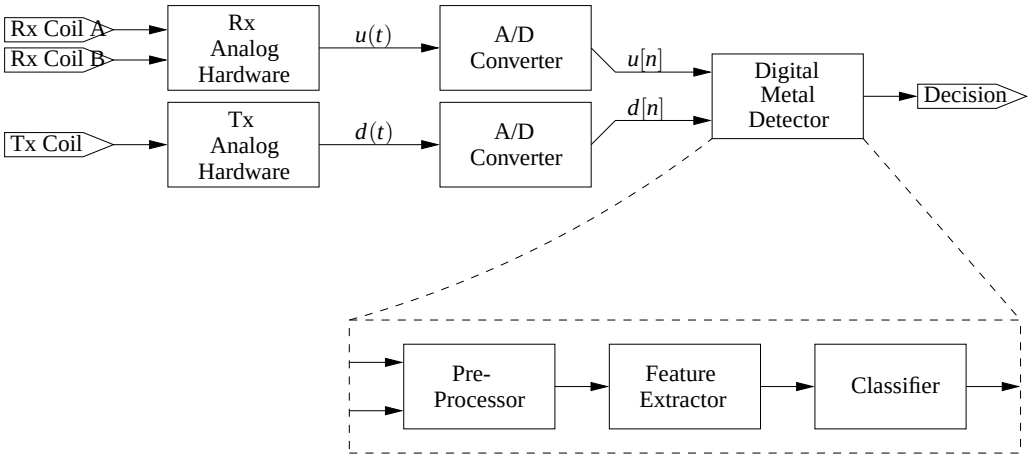

Figure 4.2: Overview of overall metal detection system for specifying interfaces.

In the following four paragraphs the system blocks are defined, and interfaces commented.

Rx Analog Hardware specifies the interface between the two receiving coils (called coil A and B), and the A/D converter. The observation signal $u(t)$ is defined as the difference between the induced voltages in the receiving coils, and therefore a differential amplifier is needed. Also amplification is needed to fit the input voltage range of the A/D converter.

Tx Analog Hardware specifies the interface between the transmitting coil and the A/D converter. It is the current in the transmitting coil that is of interest, so to sample the reference signal $d(t)$, the current has to be converted to a voltage, which can be done by putting a resistance in series with the coil. Also amplification is needed here, such that the amplitude of $d(t)$ fits the input

voltage range of the A/D converter.
To force current through the transmitting coil a power amplifier is needed, and to minimize the impedance of the load (the coil), a capacitor can be placed in series with the transmitting coil and the current-to-voltage converter (resistor). The resistor should be chosen such that the effects of it can neglected, see App. B.

A/D Converter specifies the interface between the continuous signals and the discretised signals. One converter is needed for both $u(t)$ and $d(t)$.

Digital Metal Detector specifies the *digital part* of metal detection system performing the decision, whether metal is present in the metal detector or not. The metal detector includes Preprocessor, Feature Extractor and Classifier, as depicted in Fig. 4.3.

Figure 4.3: Overview of digital part of metal detection system for specifying interfaces.

In the block diagram of the metal detector, two interfaces are identified. The output of the Preprocessor is the complex spectrum based on the DFT method, see Cha. 3, which is feed to the Feature Extractor block. The second interface is between the Feature Extractor and the Classifier, which differs between the two recognition strategies described in Cha. 3. For the approach concerning the Food Hypothesis, see Sec. 3.2.1, the interface is given by the statistically normalized residual error, and for the Metal Hypothesis, see Sec. 3.2.2, the interface is given as the normalised covariance. Based on the classification methods, a decision whether metal is present or not is given.

A block diagram of the overall metal detection system have been given, and the interfaces defined. In the following section specifications for the blocks and interfaces are given based on the first part of the report.

4.2 Requirements and Delimitations

This section concerns a summary of the requirements. Requirements for the analog hardware is primarily based on Cha. 1 and Cha. 2, concerning the metal detector head and the signals here from. Requirements for the digital hardware is primarily based on Cha. 2, but also Cha. 3 indeed concerning methods for classification.
The system to be implemented is usually situated in an industrial production facility, where food articles are produced. In such production facilities, the machinery consume much space and power, which is why no constraints are put on the power usage, nor area of the metal detection system. On the other hand, the performance factor is placed, which is the focus of optimization in this master thesis. The performance factor should here be interpreted as the capability of detecting metal pieces in food articles. The performance factor is bounded by the methods chosen in Cha. 3, so the implementation should be focused on implementing them under optimal conditions. The optimal conditions for the methods are found in Cha. 3.

- Performance Measure

  Detection of a single iron (FE) sphere of diameter less than 1 % of detector head opening. This definition is used in the metal detection industri and Detectronic A/S design their metal detectors according to this constrain. The definition does not tell how sensitive the metal detector should be to contaminated food articles, but this is also difficult to define, as different food articles have different magnetic and conductive properties. In this master thesis the goal is to go below 1 % of the detector head opening.

The signals from the detector head were analysed in Cha. 2, where it was seen from simulations, that the amplitude of the signals were in the range of approximately 0.1 to 200 mV. Therefore amplification is needed for the signal to fit the input range of the A/D converter. The large dynamic range of the signal amplitudes makes it a necessity to have level detection, because the risk of exceeding the A/D converter's input range is inevitable, if for example a large piece of metal is sent through the metal detector. This can however be incorporated after the converter. Another possibility is to make e.g. a logaritmic amplifier to reduce the dynamics of the signal, or to choose an A/D converter with sufficient precision. Another aspect is the out-of-balance component, which minimizes the dynamics because of the offset in the complex plane, see Cha. 2. When the offset is present, less amplification can be performed.

- Amplification Before A/D Conversion

  Based on the range of signal amplitudes the amplification is defined to be variable in the range $\{\frac{V_{ADC,in}}{V_{SIG,max}} : \frac{V_{ADC,in}}{V_{SIG,min}}\}$, corresponding to $\{\frac{V_{ADC,in}}{200 \cdot 10^{-3}} : \frac{V_{ADC,in}}{0.1 \cdot 10^{-3}}\}$.

- Level Detection

  Based on the large range of input amplitudes, level detection is needed if the input range of the A/D converter is exceeded.

Based on the dynamic range of the amplitude of the observation signal, an estimate of the precision of the A/D converter can be given. Assuming the above given amplitude range of 0.1 to 200 mV, the dynamic range becomes $20 \cdot \log_{10}(200/0.1) = 66$ dB. This corresponds to the SNR of a 11 bit A/D converter. The amplitude range is for a large range of metal sizes, so it can be expected that the amplitude range is significantly smaller. Also the out-of-balance component should be taken into consideration. The above dynamic range is given for the balanced case, but introducing the out-of-balance component in the observation signal reduces the dynamic range of the amplitude, because the out-of-balance component is added vectorially to the signal due to an object in the detector head.

- A/D Converter Precision

  Based on the dynamic range of the amplitude of the observation signal and the presence of the out-of-balance component, the precision of the A/D converter is defined to 12 bits.

As discussed in Cha. 2 the signal frequency can be chosen optimally for a specific product, to increase the signal-to-noise ratio between metal pieces and product. Unpopped microwave popcorn has been used in examples in the first part of the report, but also raisins among other products have been used. The transmitting frequency for the metal detection system, is however chosen to approximately 30 KHz, which can be supported by the frequency used in some of Detectronic A/S's

metal detectors. This frequency was concluded in Cha. 2 to be in the optimal frequency range for wet food articles. Another possibility is to choose approximately 1 MHz, and then turn the focus on dry food articles. This is not done in this report, but should be included in the next iteration for the project.

- Transmitting Frequency

  The focus is put on wet food articles, and therefore 30 KHz is chosen for metal detection system.

Based on the choice of the DFT preprocessor, and the transmitting frequency, the over sampling rate is defined according to App. A.2. In App. A.2 a simulation is performed concerning the amplitude and phase estimation performance versus over sampling. In the simulation the signal frequency is held constant, and in each DFT five periods of the siusoid is used. The simulation shows, that by increasing the over sampling rate, the window contains more samples, which decreases the variance of the phase and amplitude estimates in the specific signal frequency. Based on the results in the appendix, the over sampling rate is defined to be 50 times, which implies a sample frequency F_s of,

$$F_s = oversampling \cdot (2 \cdot Nyquist\ frequency) = 50 \cdot (2 \cdot 30000) = 3 \text{ MHz} \tag{4.1}$$

which is performed in two channels - reference and observation signals.
In App. E.3 the frequency content of the amplitude and phase trajectory of moving objects is analysed. Based on the conclusion in the appendix, 250 samples are needed to secure that Shannon's sampling theorem is meet. In App. A.3 it was found that the covariance based preprocessor needed a window length of at least 100 sample, to perform with a mean estimation error close to zero and a standard deviation of 5 % at a SNR of 6 dB. The 250 samples sets the requirement to the sampling of the phase and amplitude trajectory.

- Sampling of Phase and Amplitude Trajectory.

  The phase and amplitude trajectory of an iron sphere shall be represented with at least 250 samples. When the speed of the conveyor belt is approximately 40 meters per minute, the sample frequency of the trajectory becomes $F_c > 200$ Hz .

4.3 Discussion of System Specification

The last section of this chapter concerns a discussion of the practical aspects of the metal detection system, and the requirements for it. The section discusses the analog hardware, the choice of sample frequency, and the target architecture for implementation.

4.3.1 Analog Hardware

The analog hardware constructed in connection with the measurements, is used in the implementation phase. The analog hardware is described in App. C, and will not be futher treated in this report, as focus is put on the development of a prototype for the digital metal detector.

4.3.2 Sample Frequency

The high sample frequency can be compensated for, by mixing the signals down to a lower frequency. The force of doing this, is that the data rate is lowered, which results in either lower requirements for the performance of the digital hardware, or increase in time for more complex metal detection methods. The drawback is that the amplitude and phase estimates from the preprocessor, become less robust, because fewer periods and samples of the sinusoid can be used. For the 30 KHz metal detection system it is practical though, but if a 1 MHz system should be made, it would be an advantage to mix it down.
A possibility is to base the metal detection systems, with different transmitting frequencies, on the same digital hardware. This can be done by defining a fixed input frequency to the A/D converter, and then mix the signals to the fixed frequency. In this way, one can expect the digital part of the system to perform equally well, for all transmitting frequencies, and the optimal frequency for a specific food product can be chosen, with a minimum of reconfiguration of the system. The reconfiguration will in this case concern changing the multiplying frequency in the mixer, such that the difference or sum signal[1] ends up in the fixed frequency. This solution is however not chosen, because of focus of this master thesis.

4.3.3 Target Architecture

It is in the interest of the authors of this report to perform hardware development, which is why the DE2 Development Board based on the Alterra Cyclone-II FPGA[2] is chosen as the target architecture for the prototype of the digital metal detector. Using an FPGA gives the opportunity to implement dedicated hardware, and in the same time microprocessors with DSP functionalities. Alterra has delvoped a softcore called NIOS II, which is a microprocessor, that can be fit to a specific application, in the sense that DSP functionalities among others, can be added as the need arises.

This chapter concluded the first part of the report. The results of the analyses in Cha. 1 and Cha. 2, together with the classification methods, were summed in this chapter, to specify the requirements metal detecttion system. In the next part of the report, the design and implementation of the digital metal detector is presented.

[1] Because $\sin(\omega_1) \cdot \sin(\omega_2) = \frac{1}{2}(\sin(\omega_1 - \omega_2) + \sin(\omega_1 + \omega_2))$
[2] Short for Field Programmable Gate Array

Part II

Metal Detection System - Design & Implementation

Introduction

In the first part of the master thesis, a model of the detector head was derived. The model was used for signal analysis, which was extended to include measured signals. The signal model developed, was used to develop two classification strategies for metal detection in food articles. This part concerns the implementation of the algorithms. The implementation part consists of four chapters, based on the Rugby meta model. In the following the contents of the chapters will be described.

Algorithms

The first chapter concerns the algorithm development, where the DFT method and the two classification strategies, are mapped to algorithms. The section makes a in-depth analysis of the algorithms, to find opportunities or limitations in the algorithms. The training part of the classifier algorithms, is not described, but the method used for training by the authors is described. The algorithm chapter leads to the system design chapter, because the exact algorithms for implementation is described, and issues concerning the algorithms are known.

System Design

In Cha. 4 the specifications for the digital metal detection system was setup. The specifications were made, based on knowledge gained in the analysis of the signals, and the functionality of the classification methods. In this chapter the algorithms are split into a hardware part and a software part. The chapter describes the HW/SW codesign, and the considerations made in the split. The chapter uses the Rugby meta model, to describe the hardware and software design. The specifications set in Cha. 4, are reviewed due to the new gained knowledge in the algorithm chapter. The hardware design is described on a functional level, to see the implementation aspects from a birds eye view, which leads to the actual system implementation.

System Implementation

The system implementation chapter, focuses on the actual implementation of the system, described in the previous chapter. The chapter focuses on the implementation of the interfaces, described in the system design. The hardware part of the implementation focuses specificly on the implementation of the multiplier, and the considerations in form of timing, is discussed. The software part of the implementation, focuses on the implementation of a software platform, where the classification algorithms are executed.

System Evaluation

The part ends up evaluating the system. Specific topics have been chosen for the documentation, but all blocks in the hardware implementation have been tested, but are not shown in the report. The chapter ends up testing the classification methods, described in 3.

Chapter 5

Algorithms

In this chapter the algorithms selected for implementation are analysed. The analysis is based on aspects, concerning the implementation of the algorithms in either hardware or software. The chapter consists of three sections, where the first section focuses on the DFT preprocessor, the second on the LRAD classifier and the third the Covariance/Bayes classifier. The DFT section mainly focuses on implementational aspects, concerning fixed point implementation in hardware. The LRAD and Covariance/Bayes classifier sections focus on a software implementation, where aspects as memory usage and computational complexity is in focus. Each section ends up with an exact recipe for the real-time execution of the algorithm.

5.1 The DFT Preprocessor

In this section the DFT will be analysed in preparation for the implementation of the method. The section is divided into two parts, the first describing the actual spectrum computation, and the second describing the extraction of amplitude ratio and phase difference from the spectra.

5.1.1 The Spectrum Computation

The spectrum computation is inherit from the DFT algorithm, where the DFT computes the spectrum of a window of N samples. The DFT amplitude and phase estimation method does not need to compute the complete spectrum of both $u[n]$ and $d[n]$, because the only frequency of interest is the one representing the transmitting frequency. The computation of the spectra of $u[n]$ and $d[n]$ is expressed as,

$$U[q,k] = \sum_{n=0}^{N-1} u[n] \cdot e^{-\frac{2\pi}{N} \cdot k \cdot n} \tag{5.1}$$

$$D[q,k] = \sum_{n=0}^{N-1} d[n] \cdot e^{-\frac{2\pi}{N} \cdot k \cdot n} \tag{5.2}$$

where k is a natural positive number in the range $[0; N-1]$, describing the digital frequency. The index q represents the present time instance after the DFT. The digital frequency is known a priori to be $\frac{N}{F_s} f_t$, where F_s is the sampling frequency and f_t is the transmitting frequency. In the rest of this section only $U[q,k]$ will analysed, because the calculations for $u[n]$ and $d[n]$ are identical, and

the a priori frequency k is omitted, because it is constant. When substituting k into Eq. 5.1, the spectrum $U[q]$ in the digital frequency k becomes,

$$U[q] = \sum_{n=0}^{N-1} u[n] \cdot e^{-\frac{2\pi}{F_s} \cdot f_t \cdot n} \tag{5.3}$$

If the exponent is observed together with the digital frequency k, it is seen that the ratio $\frac{f_t}{F_s}$ has to be less than one and a natural number when multiplied with the window length N. This sets constraints to the sample frequency and window length, because the transmitting frequency is fixed as described in Cha. 4. If F_s is chosen such that,

$$F_s \quad = \quad f_t \cdot p, \ p = \{2, 3, 4, ..., \infty\} \tag{5.4}$$

it implies, that the exponential function in Eq. 5.3 will be *periodic*. The number of samples in each period of the sinusoids equals p. The ratio $\frac{f_t}{F_s}$ can ofcourse be chosen random (still with the Nyquist criterion in mind), but this does not automaticly imply periodicity. The periodicity is seen as an advantage in terms of data reduction in an implementational point of view. The number p sets constraints to the window length N, because $\frac{f_t}{F_s} \cdot N$ should equal a natural positive number. The window length can be expressed as,

$$N = p \cdot m, \ m = \{1, 2, 3, ..., \infty\} \tag{5.5}$$

where m is a natural positive number. Basically Eq. 5.5 states that a integer number of periods is present in the window used for the spectrum computations. The number m equals the number of periods of the transmitting frequency f_t present in the signal. If Eq. 5.4 and Eq. 5.5 are satisfied, it implies that the exponentials (twiddle factors) will be periodic in p samples.

Eq. 5.3 expresses the spectrum computation as a product sum of complex exponentials. This notation is rewritten by the use of Euler's formula, and separated into a real and imaginary computation,

$$U_R[q] \quad = \quad \sum_{n=0}^{N-1} u[n] \cdot \cos(\frac{2\pi}{F_s} \cdot f_t \cdot n) \tag{5.6}$$

$$U_I[q] \quad = \quad -\sum_{n=0}^{N-1} u[n] \cdot \sin(\frac{2\pi}{F_s} \cdot f_t \cdot n) \tag{5.7}$$

The cosine and sine do not need to be calculated for each sample in the window, because they are constant for all computations of the spectra. The cosine and sine are expressed as constants, giving,

$$w_R[n] \quad = \quad cos(\frac{2\pi}{p} \cdot n) \quad n = \{0, 1, 2, ..., p-1\} \tag{5.8}$$

$$w_I[n] \quad = \quad -sin(\frac{2\pi}{p} \cdot n) \quad n = \{0, 1, 2, ..., p-1\} \tag{5.9}$$

where the sine and cosine are periodic in p samples. When the constants in Eq. 5.8 and Eq. 5.9 are substituted into Eq. 5.6 and Eq. 5.7, and the periodicity property is used, the spectrum

computations are expressed as,

$$U_R[q] = \sum_{n=0}^{p \cdot m-1} u[n] \cdot w_R[mod(n,p)] \tag{5.10}$$

$$U_I[q] = \sum_{n=0}^{p \cdot m-1} u[n] \cdot w_I[mod(n,p)] \tag{5.11}$$

where $mod(n,p)$ is the modulo operator, which ensures that $w_R[n]$ and $w_I[n]$ are used periodically for the whole window of N samples. Eq. 5.10 and Eq. 5.11 express the algorithm for calculation of the spectrum of a single A/D converter channel of the system, which consists of two in total. The determination of p and m is not performed in this section, but later in Cha. 6. In Eq. 5.10 and Eq. 5.11 the operations identified are multiplications and additions. Considering both A/D converter channels, a signal flow graph of the DFT calculations of $U[q]$ and $D[q]$ can be obtained. In words the DFT consists of a multiplication, which is accumulated (summed). This multiply-and-accumulate (MAC) operation is performed N times, after which the DFT is calculated. After the result is obtained, the sum should be reset, such that a new sum can be started, corresponding to a new DFT. This is shown in Fig. 5.1.
The figure shows data dependency between the multiplication and addition, which implies that the

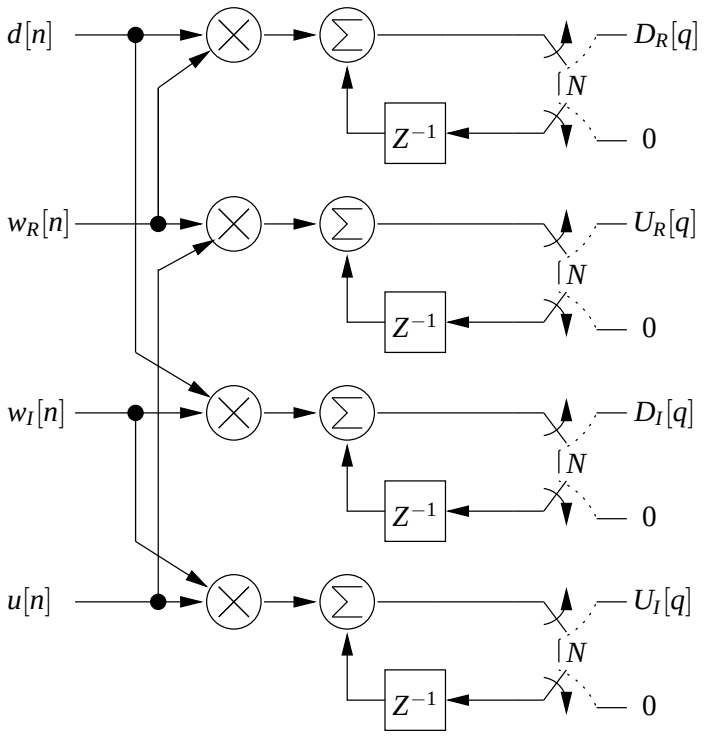

Figure 5.1: Signal flow graph of spectrum computations.

multiplication has to be performed before the addition. To perform the exact number of additions, a actuator is seen, which connects the output of the adder through a delay element to the input of it self. When $N = m \cdot p$ sums have been performed, the result is ready and the accumulator is reset. It is seen that a consequence of the actuator is a sample rate down conversion. The conversion from F_s to the new sample frequency F_c, which represents the rate of complex spectra per second,

is expressed as,

$$F_c = \frac{F_s}{N} = \frac{F_s}{m \cdot p} \tag{5.12}$$

which shows that the sample rate conversion is determined by the window length N.

An important aspect from a fixed point implementation point of view, is the gain in the spectrum after the accumulator, because the accumulator size depends on the gain in the algorithm. The maximum gain can be obtained by supplying a sinosodal signal to the system, which is in phase with $w_R[n]$ and $w_I[n]$ respectively. The signal applied cannot be in phase with both $w_R[n]$ and $w_I[n]$ at the same time, because $w_R[n]$ and $w_I[n]$ are orthogonal. The gain is calculated as,

$$\begin{aligned}
U_I[q] &= \sum_{n=0}^{m \cdot p-1} \sin(\frac{2\pi}{p} \cdot n) \cdot \sin(\frac{2\pi}{p} \cdot n) && (5.13)\\
U_I[q] &= \frac{1}{2} \sum_{n=0}^{m \cdot p-1} (1 + \cos(\frac{4\pi}{p} \cdot n)) && (5.14)\\
U_I[q] &= \frac{1}{2}(\sum_{n=0}^{m \cdot p-1} 1 + \sum_{n=0}^{m \cdot p-1} \cos(\frac{4\pi}{p} \cdot n)) && (5.15)\\
U_I[q] &= \frac{1}{2} \sum_{n=0}^{m \cdot p-1} 1 && (5.16)\\
U_I[q] &= \frac{m \cdot p}{2} && (5.17)
\end{aligned}$$

where it is seen that a signal with amplitude 1 is gained $\frac{m \cdot p}{2}$ after the spectrum calculation. It must be noted, that $\frac{m \cdot p}{2}$ is the maximum gain through the calculation, which appears when the signal applied is in phase with either $w_R[n]$ or $w_I[n]$.

In App. F a noise model is developed to described the effects of truncation before and after the accumulator. The model shows that truncation before the accumulator proportionally increases the noise in both variance and mean with the number of accumulations. When the truncation is performed after the accumulator, the noise variance is mapped directly to the output. From the analysis it can be concluded that it is preferable to truncate after the accumulator. The analysis also analyses the effect of coefficients quantisation. It is found, that if p is an even number, the coefficients can be chosen such that the fundamental frequency (the peak in the frequency response of the coefficients) does not move in frequency, when the coefficients are quantised to two or more bits. The cost of quantisation is a displacement of power, from the fundamental frequency to the harmonic frequencies. For the full explanation it is recommended that the reader consults App. F. In the next section the algorithm for amplitude and phase extraction is derived and described.

5.1.2 Phase and Amplitude Extraction

The phase and amplitude extraction is the second part of the DFT method. The extraction part has the purpose of finding the amplitude ratio and phase difference, between the complex spectra derived in Sec. 5.1.1. The extraction of amplitude ratio and phase difference is grounded in a

division between the complex spectra $U[q]$ and $D[q]$,

$$X[q] = \frac{U[q]}{D[q]} = \frac{A_u[q]e^{\phi_u[q]}}{A_d[q]e^{\phi_d[q]}} = \frac{A_u[q]}{A_d[q]}e^{(\phi_u[q]-\phi_d[q])} = A[q]e^{(\phi[q])} \tag{5.18}$$

where $A_u[q]$ and $A_d[q]$ are the amplitudes of the spectra $U[q]$ and $D_[q]$, and $\phi_u[q]$ and $\phi_d[q]$ are their respective phases. The derivation in Eq. 5.18 shows, that the division between the complex spectra, results in a complex exponential describing amplitude and phase. The spectra $U[q]$ and $D[q]$ consist of complex numbers. The complex division is rewritten to,

$$\begin{aligned} X[q] &= \frac{U[q] \cdot D[q]^*}{D[q] \cdot D[q]^*} && (5.19)\\ &= \frac{1}{D_R[q]^2 + D_I[q]^2}(U_R[q]D_R[q] + U_I[q]D_I[q] + j \cdot (U_I[q]D_R[q] - U_R[q]D_I[q])) && (5.20)\\ X_R[q] &= \frac{1}{D_R[q]^2 + D_I[q]^2}(U_R[q] \cdot D_R[q] + U_I[q] \cdot D_I[q]) && (5.21)\\ X_I[q] &= \frac{1}{D_R[q]^2 + D_I[q]^2}(U_I[q] \cdot D_R[q] - U_R[q] \cdot D_I[q]) && (5.22) \end{aligned}$$

where * denotes the complex conjugate. $X_R[q]$ and $X_I[q]$ are the real and imaginary parts of the amplitude ratio and phase difference between the signals. In Fig. 5.2 the signal flow graph of the phase and amplitude extraction part is shown.

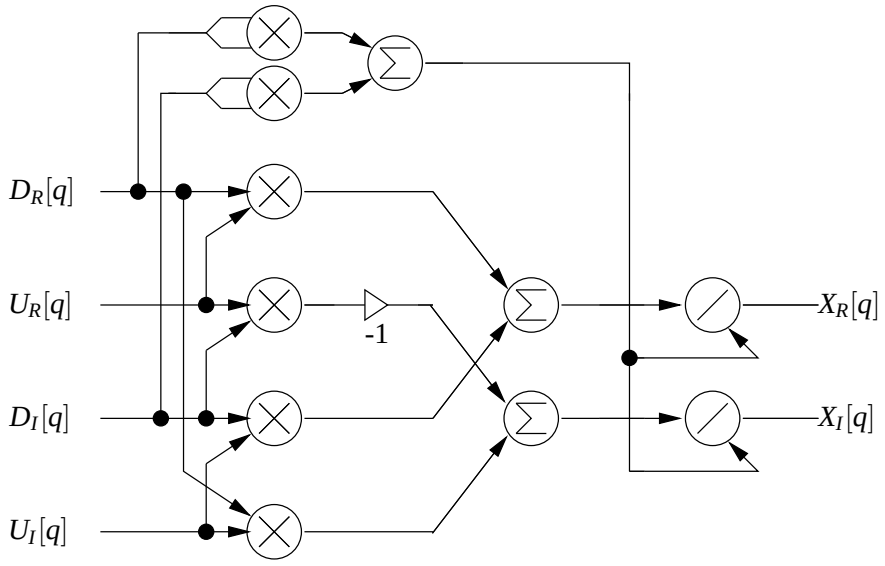

Figure 5.2: Signal flow graph of the amplitude and phase estimation algorithm.

The figure shows, that the calculation consists of six multiplications, three additions and two divisions. The figure also shows data dependency between the multiplications and additions, and between the additions and divisions.

The algorithm contains division, which makes it an object for instability. The instability arises, when the denominator approaches zero, which causes $X_R[q]$ and $X_I[q]$ to move towards infinity. To secure stability, the denominator has to be analysed. The denominator consists of the squared and summed spectrum. $D_R[q]$ and $D_I[q]$ origin from the current in the transmitting coil, which is

assumed to be constant and different from zero. This implies that the amplitude of $D[q]$ will be constant for all q. The amplitude of the spectrum is defined as,

$$|D[q]| = \sqrt{(D_R[q])^2 + (D_I[q])^2} \tag{5.23}$$

where it is seen, that the $|D[q]|$ and the denominator of Eq. 5.20 only differ by the square root operation. Due to the constant amplitude of $D[q]$, the square root operation can be expected to be a scale constant. The relation between the amplitude and the denominator can be expressed as,

$$\sqrt{(D_R[q]^2 + D_I[q]^2)} = v \cdot (D_R[q]^2 + D_I[q]^2) \tag{5.24}$$

where v is a scale constant. This implies that the denominator will be none zero and constant for all q. In the next section the algorithm of the DFT phase and amplitude estimator method is describe.

5.1.3 DFT Phase and Amplitude Estimation Algorithm

This section has the purpose of describing the steps in the amplitude and phase estimation algorithm, and the considerations necessary to get the correct performance of the algorithm. The algorithm is described for a block of $N = p \cdot m$ samples. The reason for not choosing values for p and m in this section is, that $p \cdot m$ determines the sampling of the spectrum in time, which the covariance based classifier depends on. The determination of p and m will be performed in Cha. 6. The first step in the algorithm is to compute the spectrum components $U_R[q]$, $U_I[q]$, $D_R[q]$ and $D_I[q]$, expressed as,

$$U_R[q] = \sum_{n=0}^{p \cdot m-1} u[n] \cdot w_R[mod(n,p)] \tag{5.25}$$

$$U_I[q] = \sum_{n=0}^{p \cdot m-1} u[n] \cdot w_I[mod(n,p)] \tag{5.26}$$

$$D_R[q] = \sum_{n=0}^{p \cdot m-1} d[n] \cdot w_R[mod(n,p)] \tag{5.27}$$

$$D_I[q] = \sum_{n=0}^{p \cdot m-1} d[n] \cdot w_I[mod(n,p)] \tag{5.28}$$

When the signals are converted to spectra, the rate of the spectra is expressed as,

$$F_c = \frac{F_s}{m \cdot p} \tag{5.29}$$

The sample frequency of the spectrum should be able to represent the spectrum changes in $X[q]$, which implies that Shannon's samplings theorem have to be meet. After the spectrum computations, the spectrum $X[q]$ has to be extracted. This is performed as,

$$DENUM[q] = \frac{1}{D_R[q]^2 + D_I[q]^2} \tag{5.30}$$

$$X_R[q] = DENUM[q] \cdot (U_R[q] \cdot D_R[q] + U_I[q] \cdot D_I[q]) \tag{5.31}$$

$$X_I[q] = DENUM[q] \cdot (U_I[q] \cdot D_R[q] - U_R[q] \cdot D_I[q]) \tag{5.32}$$

where $DENUM[q]$ is the constant denominator. The algorithm produces a spectrum estimate for every $m\dot{p}$ time domain samples, which is a consequence of no overlap between the blocks. For fixed point implementation, the gain of the spectra $U_R[q]$, $U_I[q]$, $D_R[q]$ and $D_I[q]$ has to be noticed, because it determines the word length of the accumulator. In the next section the algorithm for the linear regression classifier is developed and described.

5.2 The LRAD Classifier

This section describes the algorithm for the linear regression classifier with amplitude detection (LRAD). The section is parted into two sections covering training, residual error detection by the use of confidence interval. The training section focuses on the parameter estimation in the algorithm with is performed in non real-time. The residual error detection section concerns the actual computations in real time.

5.2.1 Training of LRAD

Training of the LRAD classifier consists of computing three parameters from a set of training data X_T of length M. The training data is chosen by comparing the absolute value of each new $X[q]$ with the noise floor threshold λ,

$$\text{if } |X[q]| > \lambda, \text{ then } X[q] \rightarrow X_T \tag{5.33}$$

The noise floor threshold λ is estimated in the beginning of the training session, by finding the maximum value when no objects are present in the detector head.
When the training data of length M is obtained, the maximum absolute value of the product is found. This is done by $M-1$ comparisons between the samples of the training data and $|X|_{max}$, where $|X|_{max}$ is assigned to the new absolute value if it is larger than the previously assigned value. The slope parameter α for the product is estimated from X_T as,

$$\begin{aligned}
S_{X_{T,R}X_{T,I}} &= \sum_{m=1}^{M}(X_{T,R}[m]-\overline{X}_{T,R})(X_{T,I}[m]-\overline{X}_{T,I}) && (5.34)\\
&= \sum_{m=1}^{M}(X_{T,R}[m]X_{T,I}[m]) - 2M\overline{X}_{T,R}\overline{X}_{T,I} + \overline{X}_{T,R}\overline{X}_{T,I} && (5.35)\\
&= \sum_{m=1}^{M}(X_{T,R}[m]X_{T,I}[m]) - M\overline{X}_{T,R}\overline{X}_{T,I} && (5.36)\\
S_{X_{T,R}X_{T,R}} &= \sum_{m=1}^{M}(X_{T,R}[m]-\overline{X}_{T,R})^2 && (5.37)\\
&= \sum_{m=1}^{M}(X_{T,R}[m])^2 - M\overline{X}_{T,R}^2 && (5.38)\\
\alpha &= \frac{S_{X_{T,R}X_{T,I}}}{S_{X_{T,R}X_{T,R}}} && (5.39)
\end{aligned}$$

where the subscripts R and I indicate the real and imaginary parts respectively, and the superscript $\overline{X}$ indicates the estimated mean value. To avoid that α approaches $\pm\infty$, a 45° angle is used as

threshold for mirroring the complex plane,

$$\text{If } |\alpha| \geq 1, \text{ then } \alpha = \frac{1}{\alpha} \text{ and swap the real and imaginary parts.} \tag{5.40}$$

The dynamic range of the parameter α is thereby reduced from $[-\infty : \infty]$ to $[-1 : 1]$.
The out-of-balance (OoB) component is removed by subtracting the mean value of the spectrum from each new $X_T[m]$. This means that the mean values used in the computation of α becomes zeros. This gives the following reduced equation for α,

$$\alpha = \left(\sum_{m=1}^{M} X_{T,R}[m] X_{T,I}[m] \right) / \left(\sum_{m=1}^{M} (X_{T,R}[m])^2 \right) \tag{5.41}$$

which also implies that the offset parameter β in the linear regression model becomes zero,

$$\beta = 0 \tag{5.42}$$

The last parameter is the normalisation factor (standard deviation estimate) $\widehat{\sigma}$,

$$SS_R = \sum_{m=1}^{M} (X_{T,I} - \alpha X_{T,R})^2 \tag{5.43}$$

$$\widehat{\sigma} = \sqrt{\frac{SS_R}{M-2}} \tag{5.44}$$

The parameter α is assumed to be constant, due to the constant phase observed in Cha. 2. The parameters $\widehat{\sigma}$ and $|X|_{max}$ cannot be assumed constant over all time, but the changes can be tracked by applying a lowpass filter on each using a large time constant. This will not be implemented in this master thesis.

5.2.2 Residual Error and Detection

Once the parameter training has been performed, metal detection is ready to be performed. If the α parameter was larger than one, the real and imaginary parts are swapped. Whether the parts should be swapped or not are known before the real time execution, which removes the swap decision from the execution in real time. The new spectrum $Y[q]$ which is swapped or not is used for detection. The algorithm for detection in real time is,

1. Compute mean value and remove it from $Y[q]$.
2. Check for amplitude detection.

$$\text{If } |Y[q]| > \left| X_{max} - \overline{X}_T \right|, \text{ then product not present(detection)} \tag{5.45}$$

3. Check the residual error.

$$\widehat{\epsilon}[q] = \frac{|Y_I[q] - \alpha Y_R[q]|}{\widehat{\sigma}} \tag{5.46}$$

After computation of $\widehat{\epsilon[q]}$, it is compared to the limit of the confidence interval,

$$\text{If} \quad \widehat{\epsilon[q]} \leq 1.96, \text{ then product is present,} \tag{5.47}$$

$$\text{else} \quad \text{metal is detected.} \tag{5.48}$$

In terms of computations the algorithm consist of one multiplication and three addition which is relative simple. The term $\left| X_{max} - \overline{X}_T \right|$ is precomputed as a part of the training. The usage of memory is M, which is the number of samples back in time used for the mean computation.

5.3 The Covariance and Bayes Classifier

The Covariance/Bayes classifier consists of a covariance based feature extractor and a classifier based on maximum likelihood decision rules. The two steps depend of each other, which means that the features from $X[q]$ has to be extracted, before decision can be made on $Y[q]$. A illustration of the algorithm can be seen in Fig. 5.3.

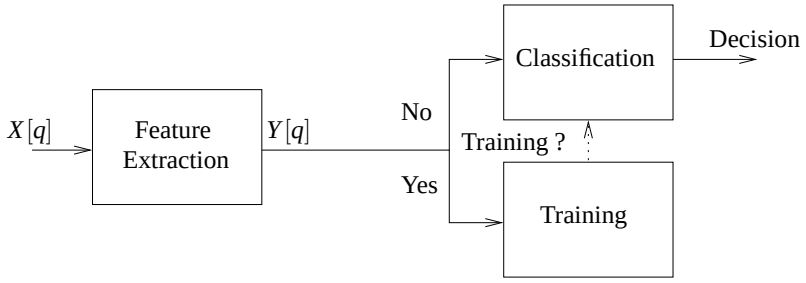

Figure 5.3: Illustration of the Covariance/Bayes classifier.

The section starts by describing the covariance based feature extractor algorithm, followed by the classifier algorithm. The training strategies will shortly be described in the end of the section.

5.3.1 Covariance based feature extractor algorithm

The covariance based feature extractor is grounded in the computation of normalised cross-covariance. The computation is expressed as,

$$Y[q] = \frac{1}{\sigma_R[q] \cdot \sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[q-n]^* \cdot (X[q-n] - \mu_X[q])) \tag{5.49}$$

where it is seen that the computation depends on the standard deviation of the matched filter R and the input X. The mean of X, μ_X, also has to be known. The match filter is known beforehand, which make it possible to normalise R such that $\sigma_R[q] = 1$ for all q. The filter can furthermore be chosen to be real instead of complex, which reduces the computations in Eq. 5.49. The new equation can be written as,

$$Y[q] = \frac{1}{\sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[q-n] \cdot (X[q-n] - \mu_X[q])) \tag{5.50}$$

where the real and imaginary part can be expressed as,

$$Y_R[q] = \frac{1}{\sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[q-n] \cdot (X_R[q-n] - \mu_{X_R}[q])) \tag{5.51}$$

$$Y_I[q] = \frac{1}{\sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[q-n] \cdot (X_I[q-n] - \mu_{X_I}[q])) \tag{5.52}$$

where it is seen that the computations of the real and imaginary parts are identical, but Eq. 5.51 and Eq. 5.52 still depends on the calculation of mean and standard deviation of $X[q]$. The standard deviation is expressed as,

$$\sigma_X[q] = \sqrt{\frac{1}{N-2} \cdot \sum_{n=0}^{N-1} (X[q-n] - \mu_X[q]) \cdot (X[q-n] - \mu_X[q])^*} \tag{5.53}$$

, where the complex conjugate secures that $\sigma_X[q]$ will be real and positive for all q. The term $(X[q-n]-\mu_X[q])$ exists in both Eq. 5.50 and Eq. 5.53, which make it an object for pre-computation. The pre-computed value $P[s]$ is expressed.

$$P[s] = X[q-s] - \mu_X[q] \quad for \quad s = \{0,1,2,3,..,N-1\} \tag{5.54}$$
$$P_R[s] = X_R[q-s] - \mu_{X_R}[q] \quad for \quad s = \{0,1,2,3,..,N-1\} \tag{5.55}$$
$$P_I[s] = X_I[q-s] - \mu_{X_I}[q] \quad for \quad s = \{0,1,2,3,..,N-1\} \tag{5.56}$$

Substituting $P[s]$ into Eq. 5.53 give rise to.

$$\sigma_X[q] = \sqrt{\frac{1}{N-2} \cdot \sum_{n=0}^{N-1} (P_R[n]^2 + P_I[n]^2)} \tag{5.57}$$

The algorithm for the covariance based feature extractor is now derived. In the following the computations will be listed as they should be performed for each q.

1. Compute mean of X
   $\mu_{X_R}[q] = \mu_{X_R}[q-1] + \frac{X_R[q]-X_R[q-N]}{N}$
   $\mu_{X_I}[q] = \mu_{X_I}[q-1] + \frac{X_I[q]-X_I[q-N]}{N}$

2. Compute P
   $P_R[s] = X_R[q-s] - \mu_{X_R}[q] \quad for \quad s = \{0,1,2,3,..,N-1\}$
   $P_I[s] = X_I[q-s] - \mu_{X_I}[q] \quad for \quad s = \{0,1,2,3,..,N-1\}$

3. Compute the standard deviation $\sigma_X[q]$
   $\sigma_X[q] = \sqrt{\frac{1}{N-2} \cdot \sum_{n=0}^{N-1} (P_R[n]^2 + P_I[n]^2)}$

4. Compute the covariance $Y[q]$
   $Y_R[q] = \frac{1}{\sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[n] \cdot P_R[n]))$
   $Y_I[q] = \frac{1}{\sigma_X[q] \cdot N} \cdot \sum_{n=0}^{N-1} (R[n] \cdot P_I[n]))$

The algorithm for the covariance based feature extractor consist of $4 \cdot N$ multiplications and $6 \cdot N+1$ additions, where N is the length of the filter R in samples. The pre computed $P[s]$ was introduced to make a reduction in additions. The cost for this reduction is and increase in memory usage. The memory usage of the algorithm is $4 \cdot N$ because the values both P and X has to be stored N samples back in time. Due to the real time execution of the algorithm the extra memory usage is assumed to be better, that adding more computations. In the next section the Bayes/maximum likelihood classifier algorithm is derived.

5.3.2 Bayes/maximum likelihood classifier algorithm

The classifier algorithm consists of implementing the three discriminant functions described in Sec. 3.2.2. The discriminant functions only differ in the parameters describing the class of interest, but the way they are computed is the same. In this section only two of the classes, which have to be

compared, are investigated. To reduce the computational complexity, the ratio between the two classes are investigated, because the ratio can be defined as in the case for metal present,

$$
\begin{aligned}
d^*_{metal}(Y[q]) &> d^*_{product}(Y[q]) && (5.58)\\
1 &> \frac{d^*_{product}(Y[q])}{d^*_{metal}(Y[q])} && (5.59)\\
ln(1) &> ln(d^*_{product}(Y[q])) - ln(d^*_{metal}(Y[q])) && (5.60)\\
0 &> ln(d^*_{product}(Y[q])) - ln(d^*_{metal}(Y[q])) && (5.61)
\end{aligned}
$$

where it is seen that the ratio can be rewritten to a subtraction between the natural logarithm to the two discriminant functions. To keep focus only one discriminant function is observed. The parameters to the classifiers is know beforehand which makes it possible to write the discriminant function as,

$$
\begin{aligned}
ln(d^*(Y[q])) &= ln(\frac{1}{2\cdot\pi\sqrt{|\sum_C|}}e^{(-\frac{1}{2}(Y[q]-\mu_C)^T\Sigma_C^{-1}(Y[q]-\mu_C))}) && (5.62)\\
ln(d^*(Y[q])) &= ln(\frac{1}{2\cdot\pi\sqrt{|\sum_C|}})\cdot(-\frac{1}{2}(Y[q]-\mu_C)^T\Sigma_C^{-1}(Y[q]-\mu_C))) && (5.63)\\
ln(d^*(Y[q])) &= l_1\cdot(-\frac{1}{2}(Y[q]-\mu_C)^T\Sigma_C^{-1}(Y[q]-\mu_C))) && (5.64)
\end{aligned}
$$

where it is seen that the constants can be precomputed. In Eq. 5.64 only one constant l_1 is shown for simplicity. The constants in Eq. 5.64 can be further reduced, but is held in the matrix notation, because the extended expression is complex.

The algorithm for the Bayes/maximum likelihood algorithm can be expressed in terms of discriminant functions.

$$
\begin{aligned}
\text{if } 0 &> ln(d^*_{product}(Y[q])) - ln(d^*_{metal_1}(Y[q])) \text{ or}\\
0 &> ln(d^*_{product}(Y[q])) - ln(d^*_{metal_2}(Y[q]) \text{ then metal present}\\
&\quad \text{else noise or product}
\end{aligned}
$$

The algorithm for the Bayes/maximum likelihood classifier is now derived. The formula for the discriminant functions are not fully extended, due to the complexity, but is done in the actual implementation. The discriminant function is kept on matrix form to keep simplicity. In terms of computational complexity one discriminant function uses 6 additions and 6 multiplications, which results in a total use of 18 multiplications and additions for all three discriminant functions. In the next section the training of the three classes and determination of the filter R are discussed.

5.3.3 Training of The Classifier

In this section the training of the classifier will be discussed. The actual parameter estimation process will not be handled, but the strategies in the training process of the different classes are in focus. The determination of the filter R and constrains will also shortly be discussed.

The strategies of training are different for the metal and noise classes. The difference lie in the selection of training data, but the parameter computation is identical. The parameter computation

will not be discussed. The noise training data has to belong to the class of noise and product, which implies that all data from the feature extractor can be used. The only condition, which has to be fulfilled, is that no metal must present in the detector head. The selection of training data for the metal classes are different because only near total correlation points has to be used. The reason for this selection is, that the product and metal objects are different in phase and amplitude trajectory. The filter R represents the amplitude and phase trajectory of a metal sphere, which correlate with all metal spheres. The selection of training data can be expressed as,

$$\text{if } |Y[q]| > \lambda \text{ then } Y[q] \longrightarrow T_{metal}$$

where T_{metal} is a vector containing training data for the class metal. λ is chosen to be 0.8.

In the previous section the filter was chosen to be real, which implies that the complex trajectory cannot be used directly. Instead of the complex spectrum, only the real values of complex spectrum are used, which does not change the trajectory. The only constrain on the filter selection is that the length for the filter N has to cover the complete trajectory. If the whole trajectory is not covered, the feature extractor cannot separate the difference in trajectory between the product and metal object. The filter has to be obtained once, because the trajectory does not change over time.

5.4 Summary of Algorithm Analyses

The chapter analysed the three algorithms in terms of computational complexity in preparation for either a software or hardware implementation. The chapter showed that the DFT method consisted of one multiplication and one addition for each real and imaginary part of each channel. It was found that the algorithm has a down conversion effect on the sample frequency, which means that the sampling rate of the spectra after the DFT method had a rate of $\frac{F_s}{m \cdot p}$. The spectrum sample rate has to obey Shannon's sample theorem, which implies that $m \cdot p$ cannot be chosen randomly. It was also found that if p is a natural number the coefficients will be periodic in p samples. The periodicity is seen as an advantage, because it reduces the number of coefficients for large values of $m \cdot p$.

The computational complexity of the Covariance/Bayes classifier was found to be,

- $4 \cdot N + 18$ multiplications
- $6 \cdot N + 1 + 18$ additions
- $4 \cdot N$ memory units for variables

which was found in the end of the sections, concerning the Covariance/Bayes algorithm. The complexity of the LRAD classifier is not taken into consideration, because of its simplicity. The complexity is used in Cha. 6 for the design of the system.

The classifier algorithms were analysed in terms of memory usage and computational complexity. It can be concluded that the covariance and byes classifier is most complex algorithm, both in terms of memory usage and computational complexity. The reason for this is, that the LRAD algorithm of only depends on the actual sample where the covariance and bayes classifier depends on window of N samples back in time, describing the trajectory.

Chapter 6

System Design

In this chapter the abstraction level is lowered for the purpose of partitioning the algorithms from Cha. 5 into digital hardware and software. Until this point, the design process has been kept on the functional level. The chapter is divided into three sections, where the first concerns the decision of boundary between hardware and software. In that section the preparation for lowering the abstraction level is initialised. In the two following sections, specification of hardware and software is performed.
The specification covers the four domains of the Rugby Meta Model, where the domains are Communication, Computation, Data and Time. The Rugby Meta Model is an expansion of the Y-Chart by Gajski, which covers only three domains - the Behavioral, the Structural and the Physical. The Y-chart was originally developed for hardware design alone, but the need for simultaneous design of software and hardware is often needed (for example in this report). Simultaneous design of hardware and software is called Hardware/Software Codesign. The first domain of the Y-chart covers the design of hardware on a high abstraction level (behavioral), which cover the behaviour/functionality of both the data path and the control path, which corresponds to forming an application. The second domain covers the structure of the hardware, which can be viewed as forming an algorithm for the application. The last domain is the physical part of the hardware, which is the target for the algorithm.
The idea of the Rugby Model is to expand the possibilities of using the Y-chart for simultaneous design of the hardware and software. Therefore the behavioral domain is parted into Communication and Computation, which cover the design of the behaviour/functionality. The Communication domain is focused on functionality of the interconnections and channels in the design, whereas the Computation domain is focused on computational functionality. To specify the channels of the Communication domain and the computational functionality of the Computation domain, the Data domain is devoted the specification of e.g. wordlengths and number representation, and the Time domain e.g. the timing requirements for the functionality.

The first section in this chapter concerns the parting of the algorithms into separate hardware and software components. The following two sections handle the separate hardware and software designs, where the Rugby Meta Model is used for structuring the design. The chapter is concluded by a summary of the System Design.

6.1 Hardware/Software Codesign

The boundary between hardware and software is decided from the mathematical operations in the algorithms, but also on basis of how static or dynamic the algorithms are. Once the transmitting frequency, sample frequency and the rate of complex spectra are fixed, the computations in the DFT become static, because of the repeated multiplications by the twiddle factors. The DFT method consists of two multiplications and two accumulator sums *for each* of the two A/D converter channels per sample. When N additions have been performed, the result is ready. Until this point in the algorithm the sample frequency has been $F_s = 3$ MHz, but after the accumulator sums the sample rate F_s is converted to the new sample frequency $F_c = \frac{F_s}{N}$ for the complex numbers of the DFT. The two pairs of sums, corresponding to the complex numbers for the reference and observation signals respectively, should be divided to obtain the amplitude ratio and phase difference. The real and imaginary parts of the complex division are given as,

$$X[q] = \frac{\widehat{X}_R[q] + j \cdot \widehat{X}_I[q]}{D_R[q]^2 + D_I[q]^2} \tag{6.1}$$

$$\widehat{X}_R[q] = (U_R[q] \cdot D_R[q] + U_I[q] \cdot D_I[q]) \tag{6.2}$$

$$\widehat{X}_I[q] = (U_I[q] \cdot D_R[q] - U_R[q] \cdot D_I[q]) \tag{6.3}$$

In these equations $U_R[q]$, $U_I[q]$, $D_R[q]$ and $D_I[q]$ are the four accumulator sums. The denominator of Eq. 6.1 is seen to be a sum of squared samples ($D_R[q]^2 + D_I[q]^2$). If the wordlength of each sample is denoted B_{accu}, the wordlength of the result of the squaring operation becomes $B_{sq} = 2 \cdot B_{accu} - 1$ for the case of signed integer data type. Once the results of the two squarings are obtained, they are summed and used in the division. The numerator consists of sums of multiplied samples, which can also result in double wordlength. Both the partial products and the result of the division gives a large dynamic range in the values. This indicates that floating point representation of the data is an advantage in the complex division.
Once the resulting complex number $X[q]$ is obtained, classification is performed. In this report, two strategies have been chosen (LRAD and Covariance/Bayes). For this reason, it is obvious to implement the classifiers in software, such that these can be easily swapped, when the need arises. For this to be possible, the boundary between hardware and software should be chosen, such that the channel in between, becomes independent of what is implemented on both sides. The choice is to place the boundary between the DFT (MAC) and the complex division. After the complex division, both classifiers can be implemented, while the interface between the hardware and software is fixed. The DFT can be implemented in fixed point, because it consists of *a known and fixed* number of products and sums, while the complex division was seen to have large dynamics in both partial and final results. Also, this boundary results in a down conversion of the sample rate, which reduces the timing requirements for the following computations. The down conversion changes the sample rate to $F_c = \frac{F_s}{N}$. The system model including the boundary between hardware and software is given in Fig. 6.1, which also includes the interface to the A/D converters.
From the system model two interfaces are seen, and these are discussed separately in the following.

The ADC/HW Interface covers the sampled signals $u[n]$ and $d[n]$, which are quantised into 12 bits. In Cha. 4 the sample frequency F_s was chosen to be a factor of 100 larger than the transmitting frequency f_t, meaning that $F_s = 3$ MHz per channel.

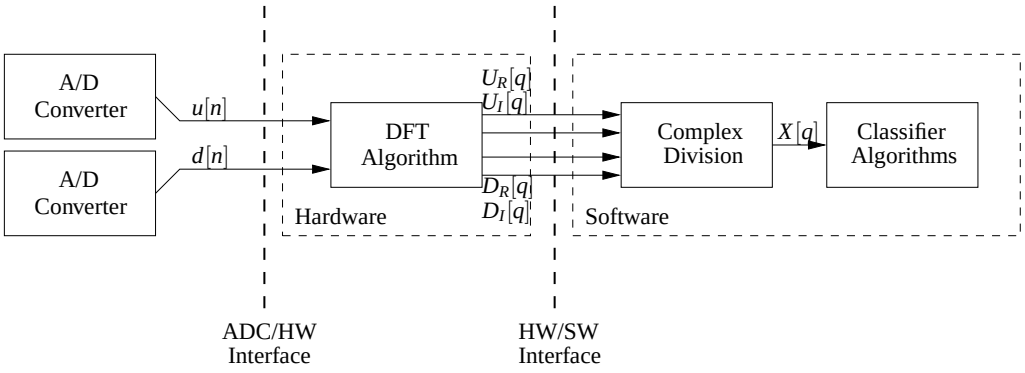

Figure 6.1: System model illustrating the hardware/software boundary and interface to the external A/D converters (ADC).

The HW/SW Interface covers the real and imaginary parts of $U[q]$ and $D[q]$, corresponding to the frequency spectrum of $u[n]$ and $d[n]$ in the transmitting frequency f_t. The rate of which the complex numbers pass the interface, F_c, was defined in Cha. 4 to be larger than 200 Hz , when the speed of the conveyor belt was set to approximately 25 meters per minute. A specific value for F_c is chosen later in this chapter under the hardware design, based on the decision of value for N.
The HW/SW interface can be implemented by polling the hardware for new results or by hardware interrupt, whenever new results from the DFT are ready. The interface is chosen to be based on interrupt, because it minimises the use of the bus in the interface. This is discussed in Cha. 7.

The partitioning into hardware and software has now been performed, and the rest of the chapter concerns the design and specification of hardware and software.

6.2 Hardware Design

In Cha. 4 the target hardware architecture was chosen to be the Altera Cyclone-II FPGA mounted on the DE2 Development Board. To develop hardware for the FPGA, the software tool Quartus, provided by Altera is used. Quartus is an Integrated Develop Environment (IDE), in which hardware design can be performed, implemented and compiled to the chosen target architecture. Together with Quartus is the SOPC Builder, which is a tool for developing the NIOS II softcore (processor), including debugging interface. The development of the processor is done according to specific requirements, such that it can be fitted to a specific need (e.g. floating point hardware). Also in SOPC Builder, external interfaces to the processor are defined, such that custom hardware can be connected. A block diagram of the hardware DFT and its interfaces is shown in Fig. 6.2.
In the block diagram the DFT is seen as two parallel data paths, where each data path corresponds to the DFT of the data ($u[n]$ and $d[n]$) from the A/D converter input channels. The input channel is given as the interface between the A/D converters and the hardware DFT. The output channel of the hardware is given as the interface between the hardware and software. The functionality of the channels form the Communication domain of the Rugby Model. The data paths (Multiplier and Accumulator) for computing the real and imaginary parts of the DFTs are equal, except from the multiplier inputs. The multiplier inputs differ, as the real part is computed by multiplication with a cosine ($w_R[n]$) and the imaginary part by multiplication with a sine ($w_I[n]$). The figure also

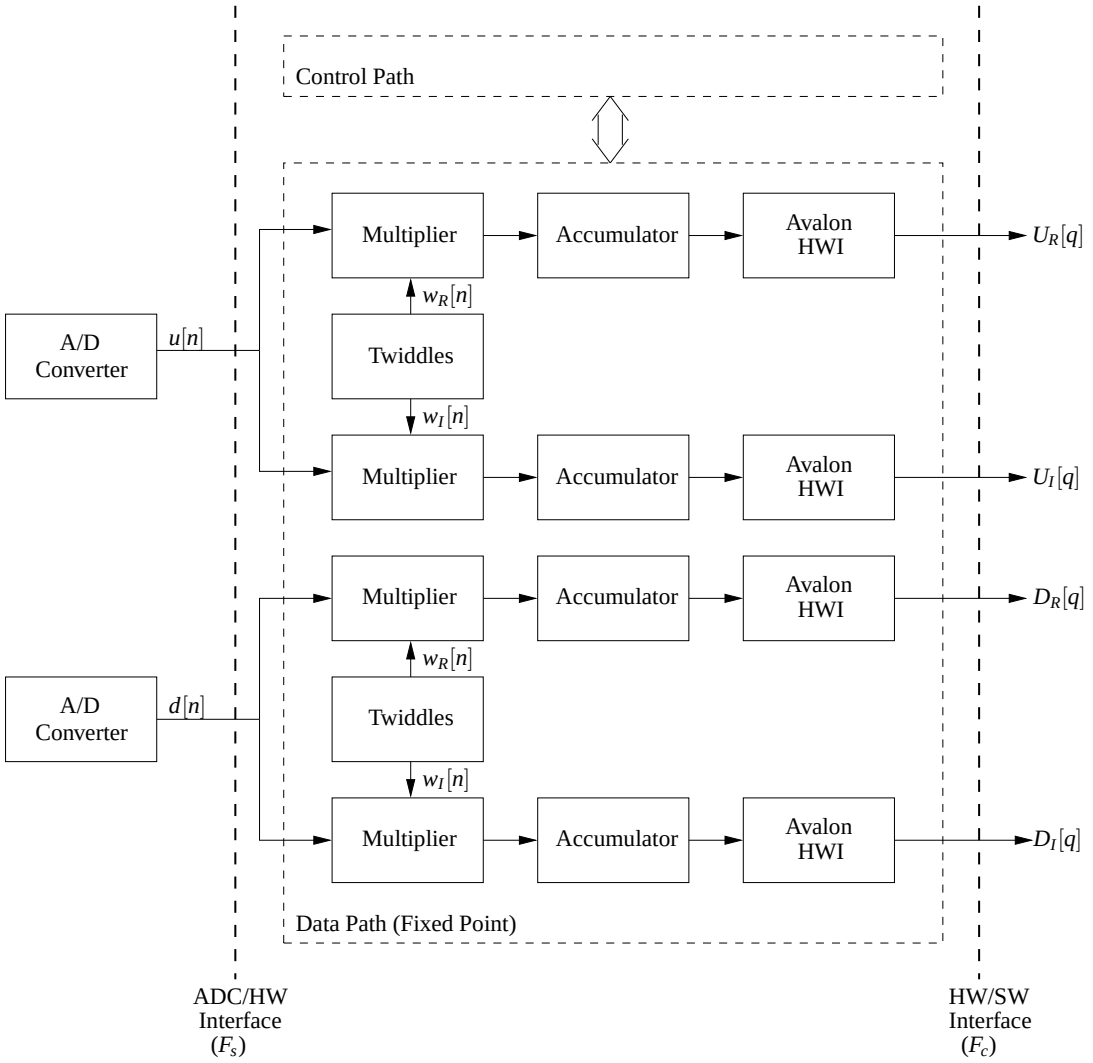

Figure 6.2: Block diagram of hardware, implementing the DFT and interfaces.

shows two blocks of Twiddles, but these should be implemented as the same for both channels. The functionality of the DFTs form the Computation domain of the Rugby Model.
The purpose of the control path is to synchronise the partial elements of the hardware and the channels. To synchronise the elements of the hardware, a master clock is used. The master clock is not shown in the block diagram, but considered as a vital part of the control path. The control path is not specifically treated in the Rugby Model, so this is considered as part of both the Communication and the Computation Domain, but is dedicated a section of its own.
The last two domains of the Rugby Model - the Data and Time domain - are considered as *specification of requirements* for the Communication and Computation domain on a lower abstraction level. In the following the control path is designed.

Control Path

The control path is responsible for the two interfaces to the data path, but also for the control of the multiplier and accumulator. The ADC/HW interface depends on the specific A/D converter

(the Analog Devices AD9220), but the sample frequency of the converter is the same as the master clock of the hardware, such that a new sample is multiplied with the twiddles and accumulated in the same clock cycle. If the propagation delay through the multiplier and into the accumulator sum is longer than the time between each sample, a pipeline can be introduced in the data path. The control of the ADC/HW interface is described in Sec. 7.1.1.
The multiplier is connected to the master clock, and the control path is responsible for supplying the correct twiddle factors to one of the inputs of the multiplier. The twiddle factors are placed in read only memory (ROM), because they are constant, so the control path should generate the correct address in the ROM in the correct state. If the twiddle factors are *not* chosen periodic, the ROM increases from p to N samples, which also increases the complexity of the control path, because the control path then should be able to address N addresses instead of p. The state machine for this part of the control path, becomes a Moore Machine, because the next state only depends on the present state and not the control input. This part of the control path only controls the addressing of the ROM.
The control structure for the accumulator consists of counting the number of samples accumulated and when it reaches N, the result should be latched and the accumulator reset. The state machine for the accumulator depends on the present state only, so it is a Moore Machine.
When the result of the accumulator is latched, the control path is responsible of notifying the HW/SW interface, such that the result can be transferred through the Avalon interface. The control path of the HW/SW interface controls transfer of data which requires that the control path supports a read cycle in the Avalon interface. The interrupt sequence and read cycle require control signals, defined by the Avalon interface, which include handshake signals for the interrupt. The handshake signals results in the state machine being a Mealy Machine, because the output of the state machine depends on the input directly(interrupt acknowledge).

In the following the four domains of the Rugby Model are used to structure the design of the data path. The control path is taken into discussion, when links are necessary.

Communication

The input channel is given between the A/D converters (Analog Devices AD9220) and the hardware. The converters are external components, which should be connected to the hardware design in the FPGA. The external pins of the FPGA package is connected internally to the gate array, such that the hardware design in Quartus can be connected to external hardware components. Connection of the hardware design and the converters is performed in the Pin Planner of Quartus, which is the tool for assignment of pins to input/output nodes of the hardware design.

The interface to the software (HW/SW) is determined by the Avalon Interconnection Fabric from Altera, which is interfaced to both DFT hardware and the NIOS II softcore. The Avalon hardware interface is written in VHDL, and added as a component for the processor in SOPC Builder. In the VHDL code for the interface, the hardware design is defined as a slave on the Avalon bus, where the slave is able to perform interrupt requests. Once SOPC Builder has exported the processor, including the Avalon interface, to a NIOS II *building block*, it is included in the Quartus hardware design schematic and will have the interface connectors, that can be used together with the custom hardware design.

The Avalon interface is defined to require interrupt request (IRQ) from the slave. Also the four accumulator results $D_R[q]$, $D_I[q]$ and $U_R[q]$, $U_I[q]$, should be transferred separately through the channel in the interface, which calls for a reading interface. The interrupt sequence is determined by the Avalon interface, and requires an IRQ signal, which should be kept asserted until the interrupt receiver has acknowledged.
The data transfer through the Avalon interface is given as a read cycle, which requires the following signals,

- Address (one for each accumulator result)
- Data (one for each accumulator result)
- Chip Select
- Output Enable
- Read

Details about the implementation is found in Cha. 7 or by consulting [Altera, 2008b].

Computation

Hardware design in Quartus is performed on the RTL abstraction level and below, but using the Megacore Independent Platform (IP) Library from Altera, large scale operations (Multipliers, MACs, FFTs, etc.) can be included in the design to shorten the development time. In this project the Megacore IP Library is not used extensively, because the authors want to develop skills in RTL/Gate level implementation through a hardware description language (VHDL). Operations on the RTL level is implemented using Quartus and Gate level operations using VHDL programming. Which parts of the design being implemented on the RTL level, Gate level or by Megafunctions is decided once the implementation of the respective block is performed.
Designs in Quartus on the RTL level are synthesised into a *netlist* for the target architecture, but VHDL code should be synthesised separately before included in the design. Synthesising the VHDL code is also done in Quartus, and once completed, Quartus makes a building block, which can be included in the hardware design schematic.

Data

The number representation in the input to the multiplier is defined as Two's Complement integer in 12 bits, which is supported by the A/D converters. The data type and precision of the twiddle factors is also chosen to this, which is argued in App. F. It is recommend that the reader consults this Appendix. Multiplication of two Two's Complement number in 12 bits, gives 24 bits with double sign bit. With the extra sign bit removed, the precision after the multiplier becomes $B_{mult} = 23$ bits. As argued in Cha. 5 and App. F, truncation/rounding before the accumulator results in accumulating the errors, which is undesirable.
To specify the size of the accumulator, the number N is needed. Earlier the sample frequency was determined to be $F_s = 3$ MHz, and the rate of complex spectra given as $F_c \geq 200$ Hz . The gain

in the algorithm was found to be $\frac{N}{2}$ in Cha. 5. The accumulator size B_{accu} becomes,

$$B_{accu} = B_{mult} + \log_2\left(\frac{F_s}{2 \cdot F_c}\right) = B_{mult} + \log_2(7500) = 35.87 \text{ bits} \tag{6.4}$$

which is rounded to 36 bits. This result should fit to the Avalon interface, in which the precision can be chosen from 8 1024 in powers of two. The full result from the accumulator is chosen to be truncated to 32 bits, corresponding removing the 4 least significant bits. Rounding can be performed, but not done because the lowest bits are expected to be dominated by noise. Another possibility is to use the *block floating point* data type, where the mantissa is defined to a fixed length and the exponent is determined by counting the number of overflow bits after the sum. However, the 32 bits integer is considered as sufficient.

The control path is responsible for counting to N, corresponding to 15000, with the above given requirements. However the complexity of the control path can be reduced, by choosing the number as powers of two. This is because the counter can be made, by simply adding one to the count each time the clock ticks, and when the counter reaches N, it is reset automatically due to counting in a limited wordlength. Therefore N is rounded up to $N = 2^{14} = 16384$. The new N results in $F_c = \frac{F_s}{N} = 183.11$ Hz , which is below the requirement of 200. Therefore the over sampling rate is changed from 50 to 64, giving a new A/D sample frequency of,

$$F_s = 64 \cdot (2 \cdot f_t) = 3.84 \text{ MHz} \tag{6.5}$$

Thereby $F_c = \frac{F_s}{N} = \frac{3840000}{16384} = 234.375$ Hz , which is above the required limit. With $p = 128$ (twice the over sampling factor), the number of periods in the twiddle factors becomes $m = 128$, which is a positive natural number as required for twiddles to be periodic, see Cha. 5. To secured that the accumulator do not overflow because the factor N was changed Eq. 6.4 is recalculated which implies that $B_{accu} = 36$. This secures that the accumulator do not overflow.

Time

The critical path in the hardware is through the multiplier and into the accumulator sum. The time t_{cp}, from a sample is obtained from the A/D converter to the multiplier result being accumulated, should be less than,

$$t_{cp} = \frac{1}{F_s} = 260.42 \text{ ns} \tag{6.6}$$

If computations take longer than t_{cp}, it may be necessary to include a *pipeline* in the MAC design, but is analysed when implemented. The rate of complex spectra F_c, corresponding to the number of interrupts of the NIOS II core per second, is given as,

$$F_c = \frac{F_s}{N} = 234.375 \text{ Hz} \tag{6.7}$$

6.3 Software Design

The NIOS II softcore is used as platform for the software, and it is included in the hardware design by use of SOPC Builder. For software development the NIOS II Integrated Development

Environment (IDE) is used, which includes C/C++ compiler. For the compiler to optimise the software for the specific NIOS II processor, it needs the SOPC Builder system file (.sopc) generated in SOPC Builder. The system file holds information about the core of the processor, which can be internal/external memory, floating point hardware, debugging hardware, etc. Using the Hardware Abstraction Layer (HAL) API, together with the debugging capabilities in the NIOS2-IDE, the *stdio* C library is supported by the software under development. Using the library, debugging can be performed by the use of *printf()*, while a program is executed on the NIOS II processor. In Fig. 6.3 a block diagram of the software is given.

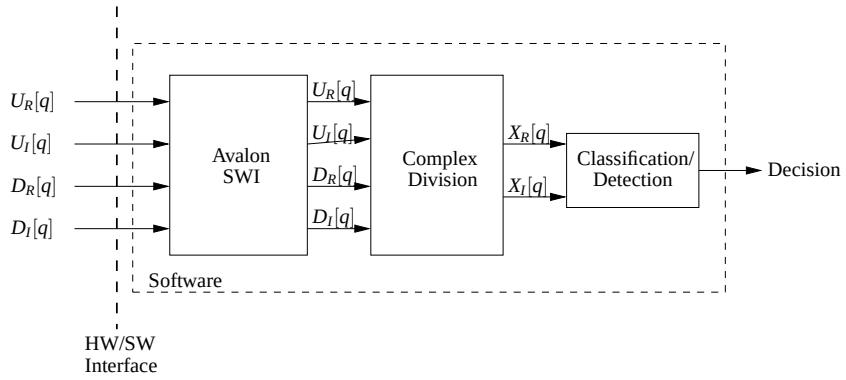

Figure 6.3: Block design of software, implementing the complex division, classification/detection and interfaces.

In the block diagram of the software, the hardware/software interface is seen, together with the complex division and classification/detection blocks. The software design is also performed according to the Rugby Model.

Communication

Communication with the hardware DFT is performed by the use of io.h. The library delivers C functions, which can be used in the software for reaching the hardware. The .sopc system file for the NIOS II processor contains information about the address space in which the hardware is placed. Also in the file, the interrupt number given to the hardware is found.

The debug tool in the NIOS2-IDE is used, together with HAL API in the NIOS II processor, for communication between the processor and the PC. The communication is performed through the USB interface, which is also included on the DE2 Development Board. The debug interface is used as target for the *decision*, see Fig. 6.3, made by the classifiers, which is done by use of *printf()*. In NIOS2-IDE a console is available, which is defined as output for *stdout*. This is only done for the prototype of the system, as the metal detector should activate a mechanism for removal of the product. As argued in Cha. 4, this is not treated in this report.

Computation

The complex division and detection algorithms are initially implemented in C for the I386 architecture (PC), and then later ported to the NIOS II processor. Compilation for the PC is done by use of the gcc compiler. Programming the software in C and then porting it to the specific architecture

has the advantage, that it can be fit to other architectures later if needed. The C/C++ compiler in NIOS2-IDE is used for the compilation of the executable code, and to transfer the code to the program memory of the processor through the USB interface.
The NIOS II processor does not feature a pipeline for the floating point computations, so no optimisations to utilise pipeline should be done in the code. Instead, the NIOS II processor on the FPGA can be extended to include custom hardware instructions, which are made on grounds of for example a block of code (e.g. a product sum) or by mathematical functions (e.g. tangent). Implementing such custom instructions, makes it possible to perform the operations faster due to the dedicated hardware, which in case decreases the execution time on the expense of more hardware and power consumption.

Data

The complex division consists of a sum of multiplications of two 32 bits numbers, which is divided by a sum of squared 32 bits numbers, see Eq. 6.1. If a 32 bit signed integer is chosen as data type, the dynamic in the result becomes 64 bits, depending on the values multiplied. The floating point data type is better suited for this computation, and it is supported by the NIOS II processor. Also custom hardware for this data type is available in SOPC Builder. The 32 bits floating point data type is chosen for data in the software.

Time

The real time requirement for the algorithms is given by the time between the interrupts, which is,

$$t_{rt} = \frac{1}{f_c} = 4.267 \text{ ms} \tag{6.8}$$

To determine whether floating point hardware or custom hardware instructions for time consuming operations in the algorithms is necessary, the NIOS II cycle count is held against the number of additions and multiplications in the algorithms. The number of multiplications and additions per complex spectrum $X[q]$ for the Covariance/Bayes classifier was derived in Cha. 5 and by inserting the filter length 250, it becomes,

- $4 \cdot 250 + 18 = 1018$ multiplications
- $6 \cdot 250 + 19 = 1519$ additions

The rate F_c is the number of $X[q]$ per second, which gives the total instructions per second,

- $F_c \cdot 1018 = 234.375 \cdot 1018 \approx 238594$ multiplications per second
- $F_c \cdot 2013 = 234.375 \cdot 1519 \approx 356015$ additions per second

The complexity of the LRAD classifier is not considered, because of the low complexity. On the Cyclone II FPGA the maximum clock speed for the NIOS II processor is approximately 50 MHz, giving the following number of multiplications and additions per second, [Altera, 2006],

- Multiplication

  153846 instr. per second for software implementation

  4166667 instr. per second for hardware implementation

- Addition

  163399 instr. for software implementation

  3571429 instr. for hardware implementation

Comparing the number of operations per second for the Covariance/Bayes classifier and the possible operations in the NIOS II processor, it is evident that the floating point custom instructions are needed to meet the real time requirements.

6.4 Summary of System Design

The chapter showed the overall structure of the hardware and software design, where interfaces between the hardware and software were defined. The considerations made in the design process of each block in the hardware were explained. The requirements were redefined to be optimal in a implementation point of view, but still obey the specification in Cha. 4. The software platform was investigated, and it was found that it was necessary to add custom floating point hardware to be able to meet the real time requirement of the Covariance/Bayes classifier.

In the following chapter, the system design is implemented, by the hardware and software parts seperately. The design is also discussed along the implementation.

Chapter 7

System Implementation

This chapter presents the implementation of the metal detector, which consists of a hardware part, a hardware/software interface part, and lastly a software part. The chapter is organised according to these three parts, and the chapter is ended by a summary. The design of the complete system was presented in Cha. 6. This chapter moves the design to the actual implementation of the system, where the implementation aspects and considerations are discussed. The chapter starts describing the hardware implementation of the DFT preprocessor together with the interface to the A/D converters, which is followed by the HW/SW interface where the actual protocol for communication is explained. This leads to the SW implementation. It is chosen to describe the software platform, wherein the classification algorithms is implemented. The first section concerns the hardware implementation.

7.1 Hardware Implementation

This section presents the implementation of the DFT preprocessor in fixed point hardware. The implementation of the data path is divided into three parts, concerning the A/D converter interface, the 12 bits Multiplier and the 36 bits accumulator. The implementation of the Avalon hardware interface to the NIOS II processor is presented in Sec. 7.2.

The master clock for both the A/D converter and the hardware DFT is derived from the 50 MHz oscillator of the DE2 Development Board, as,

$$CLK_{HW} = \frac{50}{13} \approx 3.846 \text{ MHz} \tag{7.1}$$

which results in a transmitting frequency of approximately 30048 Hz . The NIOS II processor implemented on the Cyclone II FPGA, is connected directly to the 50 MHz clock. In the following the implementation of the A/D converter and the interface to the hardware is presented.

7.1.1 A/D Converter and Interface

The analog to digital conversion is performed by the Analog Devices AD9220, which has a maximum sample frequency of 10 MHz in 12 bits. The number representation can be set to Two's Complement, and the relatively simple interface, makes it a good choice for interfacing the hardware DFT. In the following the operation of and interface to the converter is described. The circuit schematic

for the A/D converter is found in App. G. Operating the AD9220 in Two's Complement number representation is done by inverting the MSB, corresponding to the 11'th bit counting from zero. The operation of the converter is best described from its timing diagram, see Fig. 7.1.

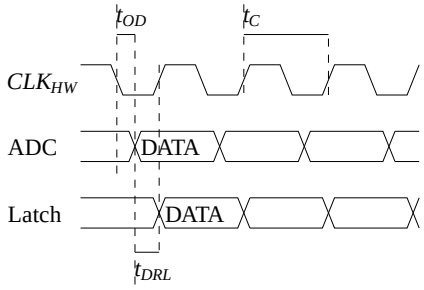

Figure 7.1: Timing diagram for the AD9220 A/D converter. The diagram is copied from Figure 1 in [Devices, 2003], and extended with the data latch. The pipeline delay is also left out.

The A/D converter is simple to interface, because the output data is latched out on every high clock. The converter has a pipeline of three samples, but this does not influence the performance of the system. The output delay t_{OD}, from the positive edge of the clock to the data being ready, is given as [Devices, 2003],

$$t_{OD} = 8 \text{ ns} \tag{7.2}$$

The clock cycle time t_C is determined from the sample frequency,

$$t_C = \frac{1}{F_s} = 260 \text{ ns} \tag{7.3}$$

and the duty cycle is 50 %. The control path is responsible for latching the data, whenever it is ready. To secure that the data is ready when latching, it is latched on every transition from high to low clock. As long as $t_{OD} < \frac{t_C}{2}$, this approach is safe, which implies that data is stable before latching. In Fig. 7.2 a block diagram for the A/D converter is given.

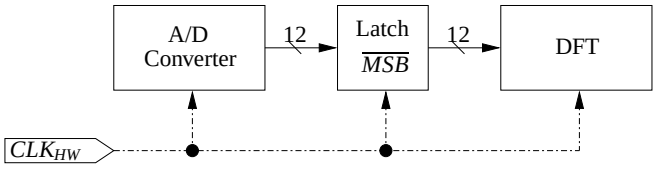

Figure 7.2: Block diagram for the connection between AD9220 and hardware DFT *in a single channel*. The latch in the interface, includes the inversion of the MSB of the A/D converter data, to obtain the Two's Complement number representation. The hardware clock CLK_{HW} is given as $F_s = 3.846$ MHz.

In Fig. 7.1 the timing diagram including the latch is presented, which includes the delay from data is ready to it is latched t_{DRL}. The time from the A/D converter data is ready, to the data is latched, is given as,

$$t_{DRL} \approx \frac{t_C}{2} - t_{OD} \approx 122 \text{ ns} \tag{7.4}$$

The timing requirement for the complete hardware DFT, becomes t_C given in Eq. 7.3. In this time the multiplication should be performed and the result accumulated. In the following the twiddle factors are implemented.

7.1.2 Twiddle Factors

Based on the choice of periodic twiddle factors, the number of constant values becomes $p = 128$, instead of the complete DFT length of $N = 16384$. The real and imaginary parts of the twiddle factors are a cosine and sine respectively. These are calculated in MATLAB for the *a priori* transmitting frequency f_t, and placed in a Look Up Table (LUT), which corresponds to ROM. The twiddle factors should be supplied to the Multiplier, synchronous with the sample from the A/D converter. Therefore the twiddles are latched synchronous with the samples from the A/D converter. In Fig. 7.3 a block diagram for the implementation of the twiddles is presented.

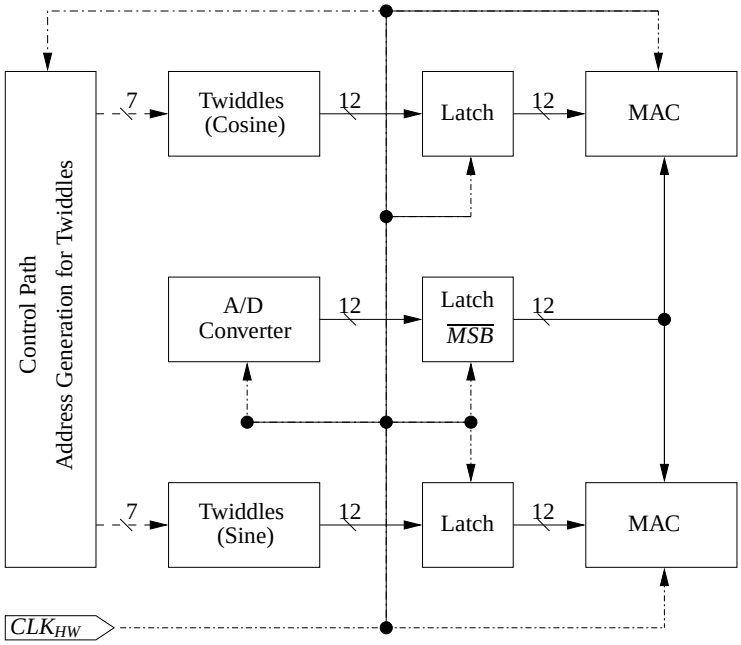

Figure 7.3: Block diagram for supplying the twiddles to the multiply-and-accumulate computation (MAC) *in a single channel*. Also the A/D converter blocks are included to show the overview. The solid lines represent data communication, the dashed lines represent address communication and dashed-dotted lines represent the clock signal.

The A/D converter data and the twiddles propagate simultaneous to the multiply-and-accumulate (MAC) computation, due to the latches. The timing diagram is given in Fig. 7.4.
The timing diagram includes the address generation in the control path (Address), which consists of the supplying a 7 bit address (Addr) to the LUT. The time t_{ADR} from address generation, to the twiddles (DATA) are ready, depends on the time used for the output of the LUT to be stable. The time from stable output of the LUT and to it is latched on the following falling edge, is denoted t_{DRL}. The time of address generation, to stable output of the LUT, is constrained by half a clock period ($\frac{t_c}{2}$), because DATA is latched on the following falling edge of the clock,

$$t_{ADR} + t_{DRL} < t_C \tag{7.5}$$

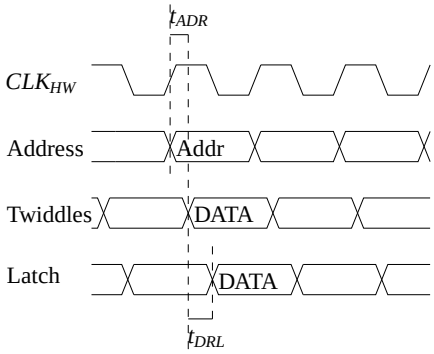

Figure 7.4: Timing diagram for the implementation of the twiddle factors. On the rising edge of CLK_{HW} the address for the twiddles (cosine and sine) is setup, and t_{ADR} later the twiddles (DATA) are ready. The DATA is latched t_{DRL} later.

The control path and LUT are implemented in VHDL, so an exact timing calculation cannot be performed, due to the synthesis of the code in Quartus. This is because it is not known how Quartus, complies the VHDL code to a ROM unit. The timing can however be evaluated and verified by use of the results of Quartus' *Timing Analyser*. The propagation delay from the control path clock input to data is stable at the output of the LUT was found to be 13.531 ns. This implies that the timing constrain Eq. 7.5 is satisfied. In the next section the implementation of the multiplier and accumulator is described.

Multiply-and-Accumulate (MAC)

The fundamental operation of the DFT is the MAC. The MAC consists of two elements, which is the 12 bits Multiplier and the 36 bits Accumulator. The determination of word sizes is done in Cha. 6. The implementation of each of these is presented in the following.

12 Bits Multiplier

To implement the 2's Complement multiplication, the arithmetic computation for a multiplication is analysed. A B-bits, 2's Complement number in finite precision, which is scaled to the range $[-1:1[$, is represented as [Oppenheim et al., 1999],

$$a = -a_{B-1} + \sum_{i=0}^{B-2} a_i \cdot 2^{-i+1} \tag{7.6}$$

where a_{B-1} is the signbit (MSB). The number a can also be written as a bit pattern, giving $a = [a_{B-1}\ a_{B-2}\ ...\ a_1\ a_0]_{LSB}$. Comparing the term 2^{-i+1} in Eq. 7.6, with the bit pattern, it is seen that each 2^{-i+1}, corresponds to a position in the bit pattern. Another number b is introduced, and the multiplication between a and b is expressed as,

$$a \cdot b = \left(-a_{B-1} \cdot 2^0 + \sum_{i=0}^{B-2} a_i \cdot 2^{-i+1}\right) \cdot \left(-b_{B-1} \cdot 2^0 + \sum_{i=0}^{B-2} b_i \cdot 2^{-i+1}\right) \tag{7.7}$$

$$= -a_{B-1} \cdot \left(-b_{B-1} + \sum_{i=0}^{B-2} b_i \cdot 2^{-i+1}\right) \cdot 2^0 \tag{7.8}$$

$$+ a_{B-2} \cdot \left(-b_{B-1} + \sum_{i=0}^{B-2} b_i \cdot 2^{-i+1} \right) \cdot 2^{-1} \tag{7.9}$$

$$+ \ldots \tag{7.10}$$

$$+ a_1 \cdot \left(-b_{B-1} + \sum_{i=0}^{B-2} b_i \cdot 2^{-i+1} \right) \cdot 2^{-B+2} \tag{7.11}$$

$$+ a_0 \cdot \left(-b_{B-1} + \sum_{i=0}^{B-2} b_i \cdot 2^{-i+1} \right) \cdot 2^{-B+1} \tag{7.12}$$

Eq. 7.7 expresses, that each single bit of $a = [a_{B-1}\ a_{B-2}\ \ldots\ a_1\ a_0]_{LSB}$ is multiplied (AND'ed) with each bit of the number b, which is bit shifted by $i+1$ bits, corresponding to the position of the bit a_i. The shift corresponds to putting different weights on the partial products, which are summed to obtain the right result for the multiplication. [Wikimedia Foundation, 2008] presents a method for multiplication, where the sign extension is saved. This method is chosen for implementation. To show how the multiplication is implemented, a 4 bit example is presented in Tab. 7.1.

line 1						b_3	b_2	b_1	b_0
line 2					$(\times)$	a_3	a_2	a_1	a_0
line 3					1	$!(a_0b_3)$	a_0b_2	a_0b_1	a_0b_0
line 4				$(+)$	$!(a_1b_3)$	a_1b_2	a_1b_1	a_1b_0	
line 5 (Stage 0)				$R_0[4]$	$R_0[3]$	$R_0[2]$	$R_0[1]$	$R_0[0]$	
line 6			$(+)$	$!(a_2b_3)$	a_2b_2	a_2b_1	a_2b_0		
line 7 (Stage 1)			$R_1[4]$	$R_1[3]$	$R_1[2]$	$R_1[1]$	$R_1[0]$		
line 8	$(+)$	1	a_3b_3	$!(a_3b_2)$	$!(a_3b_1)$	$!(a_3b_0)$			
line 9 (Result)	$R[7]$	$R[6]$	$R[5]$	$R[4]$	$R[3]$	$R[2]$	$R[1]$	$R[0]$	

Table 7.1: Illustration of 2's Complement multiplication of two 4 bit numbers a and b. The source for the method is [Wikimedia Foundation, 2008].

The 4 bit multiplication consists of two partial results, which are denoted $R_0[4]..R_0[0]$ in line 5 and $R_1[4]..R_1[0]$ in line 7. The complete result of the multiplication is denoted $R[7]..R[0]$ in line 9. The multiplication consists of *three stages*, which are seperated by single solid lines. The multiplier and multiplicand are seen in lines 1 and 2, and the multiplication result in line 9. The partial results of stage 0 (line 5) and stage 1 (line 7), and the complete result (line 9) are given in Tab. 7.2.
The first result bit $R[0]$ is equal to a_0b_0. In stage 0 the first partial result $R_0[0]$, becomes $a_0b_1+a_1b_0$. If the individual products of the sum $a_0b_1 + a_1b_0$ are both 1, the partial result $R_0[0]$ becomes 0, and a carry bit $C_0[0]$ is added to the next partial result $R_0[1]$, giving $R_0[1] = a_0b_2 + a_1b_1 + C_0[0]$. This is repeated until the last bit is obtained in the partial result of stage 0, which is $R_0[4]$. The first stage differs from the others by not having a partial result from a present stage.
The second stage (stage 1) consists of bitwise product (AND), between a_2 and the bits of the number b. Each of the bitwise results, are added (OR) to the partial result from stage 0, to obtain the next partial results $R_1[4]..R_1[0]$ of stage 1. The result stage is obtained by bitwise AND between a_3 and b, and the bitwise results are added to $R_1[4]..R_1[1]$ (with carry). The result is denoted $R = [R[7]\ R[6]\ ..\ R[0]]$.
The number 1 seen in the right side of line 8 of Tab. 7.1, is present to obtain the correct repeated sign bit (MSB), but it can be removed, such that the result is 7 bits with a single sign bit, instead

Partial Stage 0	**Partial Stage 1**	**Result**
empty	*empty*	$R[0] = a_0b_0$
$R_0[0] = a_0b_1 + a_1b_0$	*empty*	$R[1] = R_0[0]$
$R_0[1] = a_0b_2 + a_1b_1 + C_0[0]$	$R_1[0] = R_0[1] + a_2b_0$	$R[2] = R_1[0]$
$R_0[2] = !(a_0b_3) + a_1b_2 + C_0[1]$	$R_1[1] = R_0[2] + a_2b_1 + C_1[0]$	$R[3] = R_1[1] + !(a_3b_0)$
$R_0[3] = 1 + !(a_1b_3) + C_0[2]$	$R_1[2] = R_0[3] + a_2b_2 + C_1[1]$	$R[4] = R_1[2] + !(a_3b_1) + C[3]$
$R_0[4] = C_0[3]$	$R_1[3] = R_0[4] + !(a_2b_3) + C_1[2]$	$R[5] = R_1[3] + !(a_3b_2) + C[4]$
empty	$R_1[4] = C_1[3]$	$R[6] = R_1[4] + a_3b_3 + C[5]$
empty	*empty*	$R[7] = 1 + C[6]$

Table 7.2: Partial results of stage 0 (left), 1 (middle), and complete result (right) of 2's Complement multiplication, where 4 bit numbers are used. The C's are carry bits.

of 8 bits with double sign bit. In line 3 another 1 is seen, which together with the negation of the MSB of the partial sums, and the negation of line 8, excluding the MSB, compensates for not sign extending.

The three stages in Tab. 7.1 are all different. If the numbers a and b are extended to wordlengths longer than 4 bits, the computation of the middle stages, becomes identical to stage 1, but the first and last are obtained according to Tab. 7.1. In the case of 4 bits, the number of stages was seen to be three, so when 12 bit numbers are used, the number of stages becomes 11. Of the 11 stages, the nine middle stages are computed as stage 1 in Tab. 7.1. A middle stage is shown in Fig. 7.5.

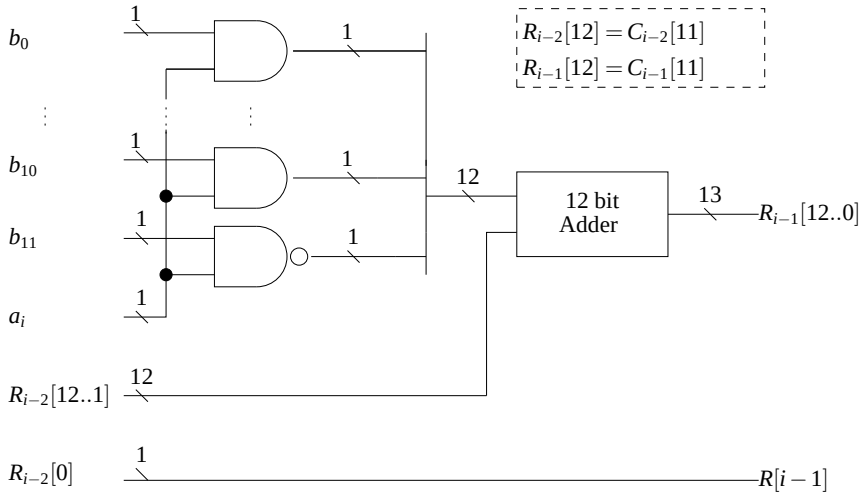

Figure 7.5: Illustration of computation of partial results for the middle stages. The computation is part of a multiplication of two 12 bit numbers $a = [a_{11}\ a_{10}\ ..\ a_0]$ and $b = [b_{11}\ b_{10}\ ..\ b_0]$. The partial results are denoted $R_{i-2}[12..0]$, the result bit is denoted $R[i-1]$ and the Cs are carry bits, corresponding to the 12th bit of the partial sums, when counting from zero.

The figure shows the bitwise AND, between a_i and $b_{11..0}$. The bitwise results are feed into a 12 bit Adder, together with the partial result from the previous stage ($R_{i-2}[12..1]$). The LSB of the previous partial result $R_{i-2}[0]$ is equal to the final result $R[i-1]$. After the 12 bit Adder the next partial result is obtained, which is denoted $R_{i-1}[12..0]$. The LSB $R_{i-1}[0]$ corresponds to the next

result $R[i] = R_{i-1}[0]$ and the MSB ($R_{i-1}[12]$) is the carry out from the 12 bit Adder.
A 12 bit Ripple Carry Adder (RCA) can be implemented by concatenating 1 bit Full Adders. A Full Adder computes the sum of two bits, and generate a carry out, if both bits are logical 1, or if one bit is 1 and a carry in is presented. If there is no carry in, the Full Adder reduces to a Half Adder. The logic for a Half and Full Adder can be seen in Fig. 7.6.

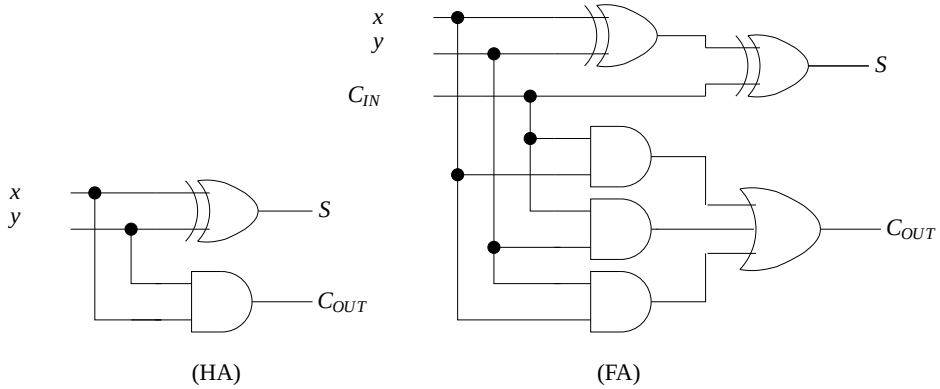

Figure 7.6: Example of logic for a Half (HA) and Full (FA) Adder. The bits x and y are those to be added, C_{IN} is the carry in from an eventual previous Half or Full Adder, and C_{OUT} is the carry out from the present Half or Full Adder. The result is denoted S. The logic schematic is from [Wakerly, 2001].

The propagation delay for the signal and carry path in a 1 bit Half and Full Adder is,

$$t_{pd,HA} = \{signal, carry\} = \{t_{XOR}, t_{AND}\} \tag{7.13}$$

$$t_{pd,FA} = \{signal, carry\} = \{(2 \cdot t_{XOR}), (t_{AND} + t_{OR})\} \tag{7.14}$$

where t_{XOR} is the propagation delay through a XOR gate, t_{AND} is the delay in a AND gate, and t_{OR} is the delay of a OR gate. In App. H propagation delays are simulated for the Cyclone II FPGA. Inserting the largest delay of the simulation results gives,

$$t_{pd,HA} = \{9.423, 9.381\} \text{ ns} \tag{7.15}$$

$$t_{pd,FA} = \{18.846, 18.218\} \text{ ns} \tag{7.16}$$

Four Full Adders are concatenated to obtain a 4 bit Ripple Carry Adder is denoted the name, because the carry ripples between the Full Adders. The critical path of such Adder, is thereby the carry path. Assigning x to the bitwise results of AND'ing a_i with $[b_{11}..b_0]$ in Fig. 7.5, and assigning y to the previous partial results $R_{i-2}[12..1]$, a 12 bit Ripple Carry Adder can be obtained, as seen in Fig. 7.7.
The critical path is through the ripple carry path, because the last partial result, requires the carry from the previous Full Adders. The critical path for the 12 bit Ripple Carry Adder (RCA) is given as,

$$t_{pd,RCA} = t_{pd,HA} + 11 \cdot t_{pd,FA} = t_{XOR} + 11 \cdot (t_{AND} + t_{OR}) = 209.821 \text{ ns} \tag{7.17}$$

where the carry path propagation delay for $t_{pd,HA}$ and $t_{pd,FA}$ is used. Including the bitwise AND propagation delay in Tab. 7.1, the total propagation time for a stage $t_{pd,stage}$, becomes,

$$t_{pd,stage} = t_{cp,RCA} + t_{AND} = t_{XOR} + 11 \cdot (t_{AND} + t_{OR}) + t_{AND} = 219.202 \text{ ns} \tag{7.18}$$

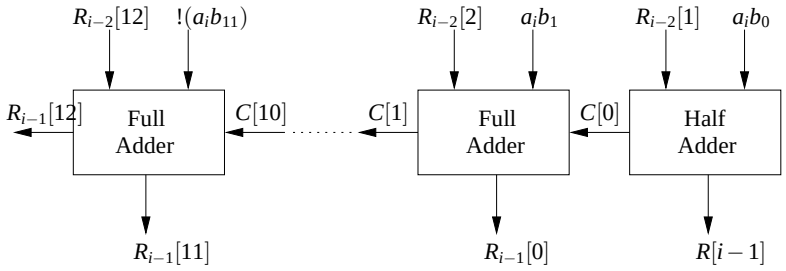

Figure 7.7: Illustration of 12 bit Ripple Carry Adder, consisting of one Half Adder, followed by 11 concatenated Full Adders.

The propagation delays for the 12 bits Adders in the last nine stages are equal. The result stage differs by requiring negation of all bits, except for the MSB, but this does not influence the ripple carry delay. The delay in the initial stage $t_{pd,init}$ is also equal to the stage delay, because the bitwise AND's can be performed in parallel, corresponding to lines 3 and 4 in Tab. 7.1.
The critical path for the complete 12 bit Multiplier is needed, to see whether optimisations have to be performed. Again the ripple carry is followed through the complete Multiplier. In the initial stage the output carry is ready after $t_{pd,init} = t_{cp,RCA} + t_{AND}$. This carry becomes the MSB of the partial result in the next stage. This gives a Full Adder propagation delay $t_{pd,FA}$ per stage. A total of 10 stages after the initial stage, gives the total propagation delay $t_{pd,mult}$ in the 12 bit Multiplier,

$$t_{pd,mult} = t_{pd,RCA} + 10 \cdot t_{pd,FA} = t_{pd,HA} + 11 \cdot t_{pd,FA} + 10 \cdot (t_{AND} + t_{OR}) \tag{7.19}$$

where the carry progation delay of $t_{pd,HA}$ and $t_{pd,FA}$ is used, and the signal progation delay of $t_{pd,FA}$ is used. With values inserted, $t_{pd,mult} = 392$ ns. With the propagation delay of the critical path in the Multiplier known, the maximum throughput F_{mult} in MegaSamples per second ($\frac{\text{MS}}{\text{s}}$) is obtained,

$$F_{mult} = \frac{1}{392 \cdot 10^{-9}} = 2.551 \ \frac{\text{MS}}{\text{s}} \tag{7.20}$$

Comparing this to the required throughput according to Cha. 6, which was 3.84 $\frac{\text{MS}}{\text{s}}$, optimizations according to speed are needed in the Multiplier.

The obvious choice for initial optimisation, is the 12 bit Ripple Carry Adder. This is because, that half of the critical path of the complete Multiplier is the ripple carry in the initial Adder ($t_{pd,RCA}$), which is seen in Eq. 7.19. However, the propagation delay in the other Adders, does also need the optimisation, such that their carry is ready, whenever the MSB of the partial results (the carry out) has rippled through the stages. A way to optimise the Adders is to used another type called Carry Lookahead Adders (CLA). The carry lookahead logic is obtained by writing the boolean equations for each carry. The boolean equations become extensive, the longer the wordlength becomes. In [Wakerly, 2001] it is stated, that carry lookahead is performed in 4-bit Adder blocks, with ripple carry in between. Based on p. 435 in [Wakerly, 2001], the delay for computing the carry bits, become three gate delays. Another delay is introduced for computing the result. This gives a total of four gate delays, for computing the sum of two four bits numbers,

including the carry out bit. The total propagation delay $t_{pd,CLA}$ for the critical path of a 12 bit Carry Lookahead Adder becomes,

$$t_{pd,CLA} = (3 \cdot 4) \cdot t_{gate} \tag{7.21}$$

$$12 \cdot t_{NAND} \leq t_{pd,CLA} \leq 12 \cdot t_{XOR} \tag{7.22}$$

$$65.04 \text{ ns} \leq t_{pd,CLA} \leq 113.076 \text{ ns} \tag{7.23}$$

which compared to the propagation delay for the RCA Adder, is between an approximate factor of two to four faster depending on the type of logic, but with the expense of using more logic for carry lookahead generation. The speed of the complete 12 bit Multiplier is however only increased by approximately 25 %, because the MSB of the partial results still has to ripple through all the stages,

$$t_{pd,CLAmult} = \frac{t_{pd,RCA}}{2} + 9 \cdot t_{pd,FA} \tag{7.24}$$

Another possibility for optimising the 12 bit Multiplier is to introduce latches between each of the stages. Between the stages both multiplier and multiplicand, together with the partial result, are latched. This corresponds to introducing a *pipeline* in the Multiplier, which requires the master clock for controlling the latches. In between the latches, the logic is unchanged. The pipeline reduces the propagation delay for the critical path by approximately two, which is because the ripple of the MSB of the partial results between the stages is removed. This results in a propagation delay $t_{pd,pmult}$ in the Pipelined Multiplier, given as,

$$t_{pd,pmult} \approx t_{pd,RCA} \quad \text{for the case of Ripple Carry Adders} \tag{7.25}$$

$$t_{pd,pmult} \approx t_{pd,CLA} \quad \text{for the case of Carry Lookahead Adders} \tag{7.26}$$

The pipeline is 11 stages deep, corresponding to a wordlength of 12 bits. Another consequence by introducing the pipeline is, that more logic elements (memory bits) are needed for the latches. Since the pipeline completely removes the ripple of the MSB of the partial results, the gain in speed becomes approximately 50 %, because it corresponds to removing the Full Adder delays $10 \cdot (t_{AND} + t_{OR})$ in Eq. 7.19.

Combining both the Carry Lookahead Adder and the Pipeline optimisations, will result in the propagation delay of the critical path being reduced by approximately a factor of four. *Not both optimisations are implemented* in this project, so the Pipeline optimisation is chosen. The theoretical maximum throughput of the Multiplier becomes,

$$F_{max,pmult} \approx \frac{1}{t_{pd,RCA}} = 4.765967 \ \frac{\text{MS}}{\text{s}} \tag{7.27}$$

where the latch delay (setup time) is excluded. If both optimisations are included, the maximum throughput will approximately be doubled, corresponding to $F_{max,both} \approx 9 \ \frac{\text{MS}}{\text{s}}$.

Synthesising of the Multiplier in the Quartus, including optimizations for speed, the propagation delay for the 12 bits Adder can be obtained. Using the result of the Timing Analyser for the critical path, the following delay is resulted,

$$\widehat{t}_{pd,Adder} = 16.427 \text{ ns} \tag{7.28}$$

meaning that the throughput of the multiplier is $\widehat{F}_{mult} = \frac{1}{\widehat{t}_{pd,Adder}} = 60.875388 \frac{\text{MS}}{\text{s}}$. The optimiser in the synthesiser of Quartus has optimised (the inefficient) RCA Adder, such that the throughput is increased approximately 13 times. Comparing $\widehat{t}_{pd,Adder}$ with the gate delays in App. H, it is noted that only 1.5 AND gate delays or three NAND gate delays can be present in the resultant Adder.

In the following the implementation of the 36 bits Accumulator is presented.

7.1.3 36 Bit Accumulator

The accumulator has the purpose of the accumulating the product from the multiplier. Basically the Accumulator is an Adder, where the output is connected to the input trough a latch. A block diagram for the 36 bits Accumulator is presented in Fig. 7.8.

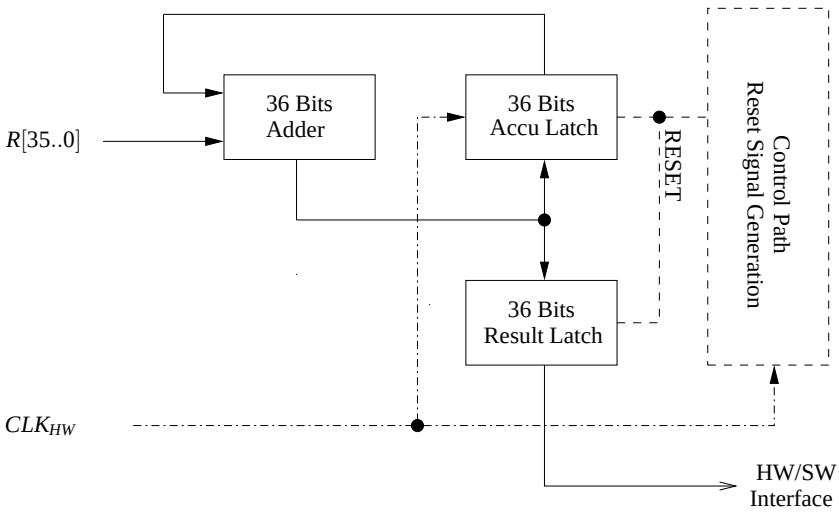

Figure 7.8: Block diagram for 36 bits Accumulator.

The 36 bits Accumulator should once per clock cycle, add a 23 bit number (R[22..0]) to the 36 bits accumulator sum. To perform the addition R[22..0] has to be sign extended to 36 bits R[35..0], to get the right number at the output. If a Ripple Carry Adder is used in the Accumulator, the worst case propagation delay becomes,

$$t_{pd,36bitRCA} = t_{pd,HA} + 34 \cdot t_{pd,FA} + 2 \cdot t_{XOR} = 3 \cdot t_{XOR} + 34 \cdot (t_{AND} + t_{OR}) = 647.68 \text{ ns} \quad (7.29)$$

where the multiplication with 34 is for the ripple carry, and the term $2 \cdot t_{XOR}$ is for the signal path in the last Full Adder. The signal path is the last term in the critical path, because the carry in is needed to obtain the last bit in the result, and no carry out occurs. The maximum throughput of this RCA Accumulator becomes,

$$F_{36bitRCA} = \frac{1}{t_{pd,36bitRCA}} = 1.544 \frac{\text{MS}}{\text{s}} \quad (7.30)$$

The RCA Adder is described in Sec. 7.1.2. Comparing the throughput, with the sample frequency of the A/D converter ($F_s = 3.84$ MHz), also optimisations are needed here.

Instead of implementing an optimised Adder for the Accumulator, a Megafunction made by Altera is used. Megafunctions are operations optimised for the FPGA architecture. The Megafunction is called LPM_ADD_SUB, and it is added in Quartus through the Symbol Tool. The MegaWizard Plug-In Manager is used for setting up the Megafunction, such that e.g. the wordlength is correct.

The output of the Accumulator is truncated to 32 bits, which was determined in Cha. 6. After the truncation, the 32 bit word is passed through the Avalon interface to the software in the NIOS II processor. The implementation of the HW/SW interface is presented in the following.

7.2 HW/SW Interface

The HW/SW interface is the channel between the hardware part of the system and the software part. The section focuses on the communication protocol between the hardware and software, and how the protocol is implemented in the system. The section starts describing the Avalon Interconnection Fabric and the protocol used for communication. Afterwords the actual implementation and considerations made in form of timing is explained. The section is based on [Altera, 2008b].

7.2.1 The Avalon Interconnection Fabric

The Avalon Interconnection Fabric is a bus structure implemented in Altera's FPGAs. The Avalon Interconnection Fabric enables developers to connect peripheral units to the System On Chip (SOC) processors, such as the NIOS II processor. The Avalon interface has many functions and modes, but in this report the focus is placed on the Memory Mapped mode. The interface is called memory mapped, because the registers in the peripheral units are mapped into the address space of the master unit. Physically the registers are placed in the peripheral unit, but when the NIOS II processor reads the address, the register in the peripheral unit is read. The system on chip design used for this project is shown in Fig. 7.9.

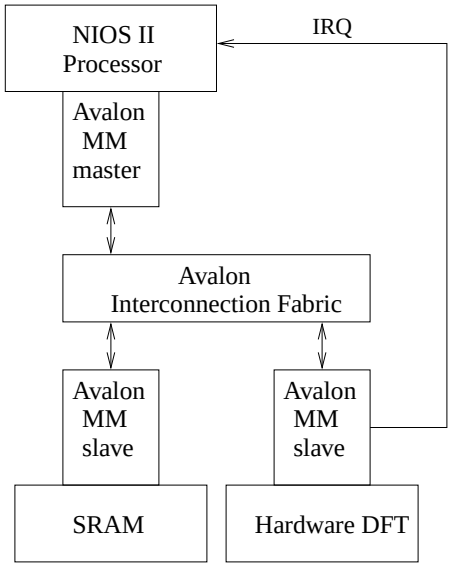

Figure 7.9: Figure of the Avalon Interconnection Fabric with connected units.

The figure shows the Avalon Interconnection Fabric with a SRAM and the DFT unit connected.

The NIOS II processor is seen as the master unit and the peripherals as the slave units. The term master specify the unit, that initiate a read or write cycle at the Avalon Interconnection Fabric. The term slave specify a unit, which replies at the requests from the master. The HW/SW interface is a read only interface, because the NIOS II processor only has to read from the DFT unit. In the next section the protocol for communication through the HW/SW interface is presented.

7.2.2 HW/SW Interface Protocol

The HW/SW interface protocol section concerns a read cycle and the interface constrains inherit from the master unit. The HW/SW interface controls the communication in the channel at the slave side. The HW/SW interface will in the following be named the slave unit. In the following steps, the interface protocol for a read cycle is specified.

1. The slave unit interrupt requests the NIOS II master unit.
   The slave unit has to interrupt the NIOS II master unit to notify the master, that it wants to be served. The slave unit cannot write the data to the master unit directly, because it is the master unit, that controls the Avalon Interconnection Fabric. The interrupt request has to be held until the master starts to service the slave. If not held, it cannot be secured, that the master unit serves the slave.

2. The master unit sets chipselect, address bus and read strobe through the Avalon Interconnection Fabric.
   The read strobe tells the slave unit, that the master unit wants to read the address pointed to by the address bus. The slave does not decide, which address should be read, because the master determines this by the address bus in the Avalon Interconnection Fabric.

3. The slave sets the data determined by the address bus at the data bus.
   The data at the data bus is read by the master unit, and the slave unit does not need to remove it again, because the Avalon Interconnection Fabric does not require this.

The protocol is now defined for communication between the NIOS II processor and the spectra computation unit. Constrains set to the HW/SW interface is inherit from the NIOS II processor, because it is the master unit. The constrain is that the data has to be placed at the data bus in a little endian format, because the NIOS II processor is little endian. If the data is not placed in little endian order, data will not represent the correct value in the NIOS II processor, because the bits are swapped from the NIOS II processor point of view. The data bus size of the NIOS II processor is 32 bit. To fit the bus of the slave to a master unit, the slave is also set to 32 bit bus width. The bus width of the slave unit can also be set to 8,16,64,128,256,512,1024 bits, but it will cause for larger bus sizes, multiple read cycles controlled by the Avalon Interconnection Fabric, and for smaller sizes the word read will be copied to fit the width of the master. The optimal is to fit the slave bus width to the master bus width to avoid multiple copies of the word in the NIOS II processor address space, or multiple read cycles at the bus. The copies occur, because the processor expects to get 32 bits every time the slave unit is read. If this is not the case, the word fetched is copied until it is 32 bits. In the next section the implemented HW/SW interface is explained from the slave side. The master side is controlled by the NIOS II processor and the software executed at the processor.

7.2.3 HW/SW Interface From Hardware Side

The HW/SW interface has to follow the protocol described in the previous section, but there are also timing constrains to care about. The section explains the implementation in form of enumerations, where the steps in the protocol, and implementation issues, are discussed.

1. Interface preparations.
   The interface preparations are to receive the DFT spectra from the computation unit and place them in registers, which should be transferred via the Avalon Interconnection Fabric. The registers are four separate 32 bit registers, which are addressable by the master unit. The spectra are placed in a little endian order, to satisfy the endian constrain of the master unit. The reason for latching the results in registers is, that the computation unit and the Avalon Interconnection Fabric is not synchronised. The latching makes it possible to synchronise the communication between the DFT computation unit and the Avalon Interconnection Fabric. The HW/SW interface is notified by the DFT computation unit, when new spectra are placed in the registers.

2. The HW/SW interface interrupt requests the NIOS II master unit.
   The interface consists of two processes. A process which is synchronised to DFT computation unit, and handles the interrupt request and handshaking with the master unit. The other process handles the communication via the Avalon Interconnection Fabric and is synchronised to the Avalon Interconnection Fabric. The interrupt process basically latches the new spectra, and sets the interrupt request, and holds it, when notified by the DFT computation unit. The interface is now waiting for the read cycles from the master unit.

3. The master read cycle from the Avalon Interconnection Fabric.
   The timing for a read cycle is shown in Fig. 7.10.

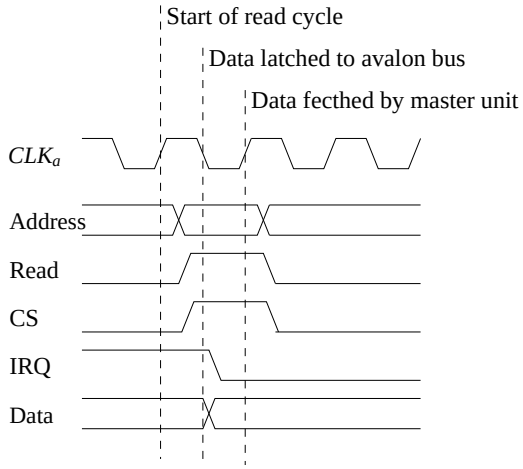

Figure 7.10: Timing diagram of a Avalon master read cycle.

The figure shows that a read cycle at the Avalon Interconnection Fabric starts at the rising edge of the Avalon Interconnection Fabric clock called CLK_a. After the rising edge, the master unit sets the address bus, and asserts read and chipselect to notify the HW/SW interface, that it wants to read the register pointed to by the address bus. At the falling

edge, the process controlling the Avalon communication sets the data wanted at the Avalon Interconnection Fabric data bus. The interrupt process removes the interrupt request when the chipselect and read are set, because this ensures that the master unit has started serving the interface. The read cycle is performed four times because all spectra have to be fetched from the interface. The interface does not interrupt the master unit for each register fetched, because the software in the NIOS II processor knows, that it has to fetch all four register at once.

The timing of a read cycle between the NIOS II processor and the HW/SW interface is important, because the data set at the data bus at the falling edge, has to be stable at the rising edge of the clock, where the NIOS II processor reads the data bus. The propagation from the internal registers to the data bus has to be less than 10 ns, because the Avalon Interconnection Fabric clock is 50 MHz. If the timing constrain is not satisfied, the master unit does not fetch the right data at the data bus. It is assumed, that the timing constrain is satisfied, because the registers are placed in the FPGA, together with the Avalon Interconnection Fabric. The delay from a register to the Avalon Interconnection Fabric is not specified by the manufacturer, but a solution, if the timing does not hold, is to insert a wait state. The wait state forces the master unit to wait one clock cycle, before it fetches the data. The timing constrain is tested in the system evaluation Cha. 8.

7.3 Software Implementation

In this section the software implementation is described. The section concerns the scheduling of the algorithm, and the mechanisms used to synchronise the real-time scheduling and the shared memory handling, between the two routines in Fig. 7.11. The section starts describing the hardware interrupt routine, where the software accesses the channel and places the new spectra in shared memory. Afterwords the main routine is described, where terms as accessing the shared memory and scheduling are discussed. The section is based on [Altera, 2008a]

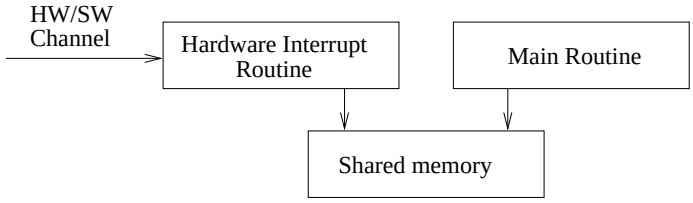

Figure 7.11: Figure of the general implantation of the software.

The first section handles the implementation of the hardware interrupt routine.

7.3.1 Hardware Interrupt Routine

This section handles the hardware interrupt routine. It was chosen in Cha. 6 to use hardware interrupt. This implies the software to have a hardware interrupt routine, which services the hardware interrupt from the DFT unit. The purpose of the hardware interrupt, is to read the spectra from the Avalon interface. Due to the hardware interrupt, the hardware secures, that new spectra are ready, when the NIOS II processor is interrupted. The difference between an interrupt and polling interface is, that a polling interface has to ask the hardware, whether new spectra are ready or

not. This leads to a continuous use of the Avalon Interconnection Fabric, where other units on the bus cannot use the bus. If no other units want to use the Avalon Interconnection Fabric, it does not have any influence on the actual execution. The interrupt is chosen, because it can be used as a synchronisation of the real-time execution, and does not occupy the Avalon Interconnection Fabric, as it is with the polling interface. The polling interface is as good for synchronisation as the interrupt routine, the difference lie in the use of the Avalon Interconnection Fabric.

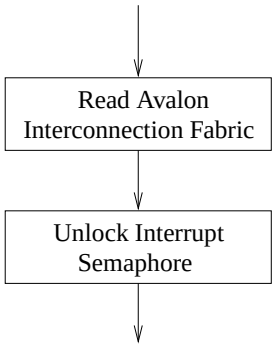

Figure 7.12: Flowchart of the hardware interrupt routine.

The flowchart of the hardware interrupt routine is shown in Fig. 7.12. The flowchart shows that the interrupt routine fetches the new spectra from the hardware, and unlocks the interrupt semaphore. The read of the new spectra is done from four addresses, which are mapped from the hardware to the memory map of the processor. Physically the registers read are placed at the hardware interface, but when read, the data is transfered via the Avalon Interconnection Fabric to the shared memory in the NIOS II processor. The interrupt semaphore serves as a blocking mechanism for other routines, in this case the main routine. The semaphore is the main part of the synchronisation, because the main routine blocks until the hardware interrupt routine has unlocked the semaphore. This secures that the main thread only execute, when new spectra are available in shared memory.

In the next section the implementation of the main routine is described. The actual implementation of the algorithms, will only be performed on a block basis, because the algorithms are implemented according to Cha. 5.

7.3.2 Main Routine

The main routine serves as the routine, where the execution of the complex division, feature extraction and classification algorithms are executed. The routine also serves as the routine, which initialises the classifier in terms of computation of parameters and buffer initialisation. The training part of the classifiers is not covered in this section, because it is not seen as a part of the real time execution.

The structure of the main routine has to be flexible, such that it can be used for both the LRAD and the Covariance/Bayes classifier. The main thread can be interpreted as a software platform, where modules in the platform can be switched easily, without changing the structure of the platform. The flexible software platform is wanted, because two algorithms have to be setup and tested in terms of classification performances.

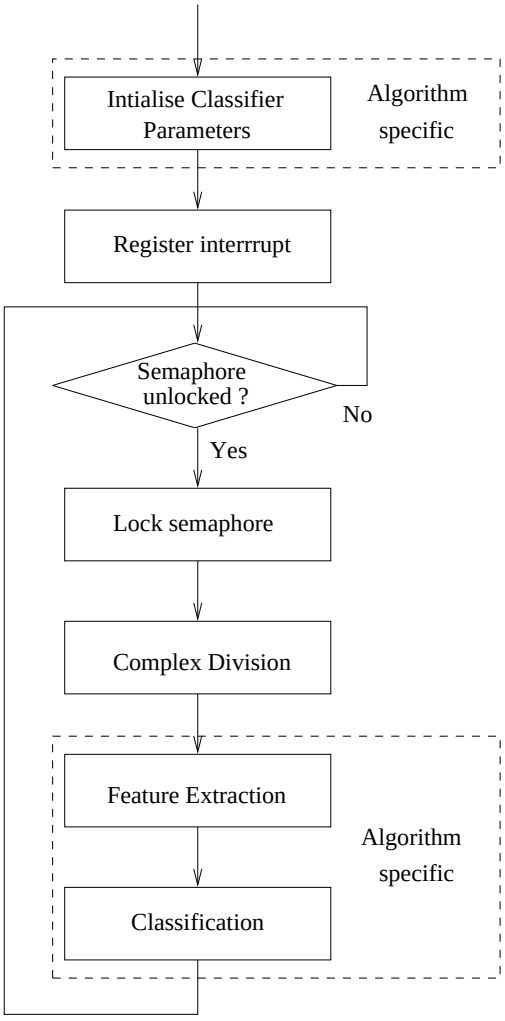

Figure 7.13: Flowchart of the main routine.

In Fig. 7.13 the software platform is shown. The figure shows, that the design consists of some algorithm specific blocks, which can be switched, depending on the algorithm up for test. The algorithm specific blocks are surrounded by dashed lines. The non-algorithm specific blocks in the design is the basis for software platform. In the following the software platform will be described. Afterwords the algorithm specific blocks will be shortly described for each algorithm.

Register Interrupt

The register interrupt block is a part of the initialisation of the software platform. The register interrupt block initialises the hardware interrupt. To perform the initialisation the HAL API is used, which is a part of the NIOS2-IDE Development Environment. The HAL API consist of some hardware specific C functions, which enables the software developer to access the NIOS II hardware registers. The function used to register the interrupt is `alt_irq_register(alt_u32 id, void * context, void (*isr)(void * ,alt_u32 ))`. The function looks complicated at first sight, but basically the argument `id` is the interrupt number and `*isr` is a pointer to the address of the interrupt service routine, which in this case is the hardware interrupt routine. Processors

in general have a interrupt vector table, which is placed in a specific memory location, containing the address to jump to, when a specific hardware interrupt occur. The HAL API enables the programmer to abstract from the placement of the vector table, by passing the arguments to the function, which fills the location match to `id`, with the address of `isr`. The argument `context` is a pointer to the argument of the hardware interrupt routine, which in this case is the shared memory, enabling the users to pass arguments to the hardware interrupt routine. Basically the block registers the interrupt, and sets up the interrupt vector table of the NIOS II processor.

Semaphore Unlocked/Semaphore Lock

The semaphore unlock question is the main synchronisation unit of the main routine. The decision synchronises the main routine with the hardware interrupt routine, because the only routine, which can unlock the semaphore, is the hardware interrupt routine. This blocking mechanism secures, that the main routine only perform one classification for each new spectra.

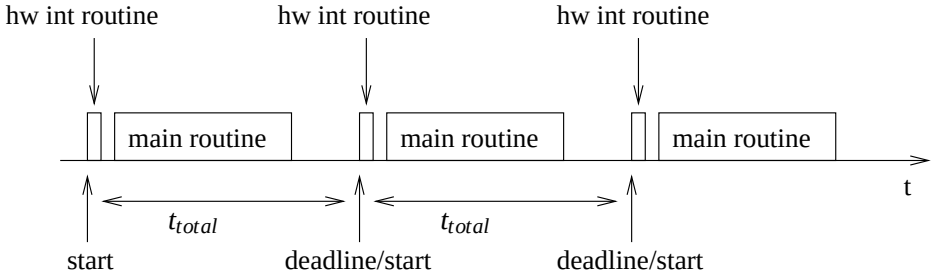

Figure 7.14: Timing diagram of the hardware interrupt routine.

In Fig. 7.14 the timing diagram of the hardware interrupt and main routine tasks are shown. The figure shows that the main routine task has to be executed after the hardware interrupt routine. The scheduling between the tasks are done by the semaphore, because the hardware interrupt unlocks the semaphore, and let the main routine execute the complex division, feature extraction and classification steps once. The execution of the hardware and main routine, have to meet the requirement, that the execution of both routines have to be finished, before a new interrupt occur. If this is not the case, the algorithm cannot be executed in hard real time. The real time constrain is important for the Covariance/Bayes classifier, because the algorithm depends on the trajectory of the signal. If the real time constrain is not satisfied, it will give rise to missing samples in the trajectory, which will degrade performance of the algorithm, because the matched filter does not fit anymore. In Fig. 7.14 points in time named start and deadline are marked for each execution loop. The point start symbolises the hardware interrupt, and deadline symbolises the time, where the two task should be finished. The time between start and stop, is the time requirement for the execution time, marked as t_{total}. The timing requirement for the software execution was found to be $t_{total} = 4.267$ms in Cha. 6. This leads to the following inequality,

$$t_{total} \geq t_{int} + t_{main} \tag{7.31}$$

where t_{int} is the execution time of the hardware interrupt routine and t_{main} is the execution time of the main loop. If the inequality Eq. 7.31 does hold, the real time requirement of the software execution is meet. In the next section the first block in the main loop is described.

Complex Division

The complex Division is based on the exact computation of Eq. 5.30 to Eq. 5.32. The block return the complex spectrum as the real and imaginary parts. Before the computation the spectra are type casted from 32 bit signed integer to 32 bit floating point. The type cast reduces the SNR of the 32 bit word, but is seen as insignificant.

Feature Extraction/Classification

The feature extraction and classification blocks are the algorithm specific blocks in the main loop. The blocks are implemented exactly as described in Sec. 3.2.1 and Sec. 3.2.2, because the recipes are mapped directly to the C code. Both algorithms have to interface the complex division, such that the real and imaginary parts of the complex spectrum is obtained in the right order.

Classifier Initialisation

The initialisation block setup the classifiers. For the LRAD method it fetches the α and $\widehat{\sigma}$, and computes the term $|X_{max} - \overline{X}|$. It also sets up two circular buffers for the mean value computation, because the method has to know N samples back in time to calculate the mean value.

The Covariance/Bayes classifier computes the parameters for the Bayes classifier and fetches the filter R. The method sets up four circular buffers. Two for the mean computation, and two for the precomputed values of $P[s]$.

In the next section the points drawn in the previous sections are summarised and concluded.

7.4 Summary of System Implementation

The implementation of the hardware DFT was presented in Sec. 7.1. The implementation of the hardware part of the metal detection system, consisted of the A/D converter interface, the twiddle factors, the 12 bits Multiplier and the 36 bits Accumulator. Analysis of the critical path of the Multiplier revealed, that optimisations were needed to meet the timing requirements. Two possible optimisations were discussed, and the Pipelined Multiplier solution was chosen, because it was sufficient to meet the timing requirements. After implementing the Pipeline Multiplier in VHDL, it was synthesised in Quartus. The synthesiser in Quartus includes an optimiser, which resulted in a factor of 13 in reduction of the propagation delay compared to the theoretical computed propagation delay. After the Multiplier, a Accumulator was present, which was implemented by the use of the Altera's Megafunctions. The Megafunction was chosen, due to the timing requirements.
The implementation of the HW/SW interface was presented in Sec. 7.2. The protocol for communication through the channel was explained, and it was found that the optimal bus width of the spectra computation unit was 32 bits. It was chosen to avoid multiple read cycles at the interface and copied words in the address space of the NIOSII processor. The timing of a read cycle was explained, but could not be checked by data sheets from the manufacture of the FPGA. The timing of the interface is verified in Cha. 8 to secure the interface is stable.
The implementation of the software platform was presented in Sec. 7.3. It was chosen to develop

a flexible software platform, which could cope with the implementation of two classification algorithms. The software platform consists of two routines, a main and hardware interrupt routine. The scheduling of the routines were implemented with a semaphore, which blocks the main loop until the interrupt has occurred. The semaphore also secures, that the main loop not accesses the shared memory, which contain the spectra from the HW/SW interface, before new spectra have arrived. It was found, that the execution time of the main loop and hardware interrupt routines had to satisfy the inequality of Eq. 7.31 to meet the real time requirements. The real time requirement is evaluated in Cha. 8.

In the following chapter the implemented metal detection system is evaluated.

Chapter 8

System Evaluation

In this chapter the digital metal detection system will be evaluated. The section starts summarising the test performed on the on the hardware adder in the multiplier, see Sec. 7.1. Afterwards the HW/SW interface is tested, to see whether the timing constrain in the interface holds. The documentation of the HW/SW interface is placed in Sec. 7.2. The section ends up evaluating the classification methods. The question in Cha. 3 was whether it was best to search for the metal object, or to search for the product component, to be able to detect, when something else is present in the detector head. The answer to this question will be stated last in this chapter through the system evaluation.

8.1 Hardware Verification

The timing constrain of the hardware adder was setup up in section Sec. 7.1 and tested. The adder is the basic building block in the multiplier. The test showed that the adder was 13 times faster than estimated, based on gate delays. It is assumed that the speedup is caused by optimisations performed by the synthesiser in Quartus. The speedup is not seen as a disadvantage, because it increases the throughput of the multiplier.

In Sec. 7.1 the timing of the twiddles to the multiplier was measured, and it was concluded that the timing constrain was satisfied, which implies, that the twiddles supplied to the multiplier, also are latched by the multiplier.

In this section the complete test of the hardware DFT preprocessor is performed. The test is based on the knowledge, that the timing of the twiddle factors is correct. The test uses the twiddle factors for stimuli for the DFT preprocessor, which means that the sine twiddles are used for the input to the multipliers. The reason for using the sine twiddle factors, are that the twiddles are in phase with the computation of the imaginary parts of the spectra, and orthogonal to the computation of the real parts of spectra. This gives the opportunity to fully utilise the accumulator in the computation of the imaginary part, and zero it in the real part computation. The setup in Quartus is placed on the DVD, attached to the backside of the report. The test is performed for a window of 16384 samples. The clock frequency applied to the DFT preprocessor is 50MHz. The clock frequency is chosen to shorten simulation time, but do not influence the output of the simulation, because the timing of the multiplier and adder are still fulfilled. The truncated outputs

from the hardware part of the DFT preprocessor, are shown in table Tab. 8.1

channel	Measured	Calculated	Utilised bits in accumulator
channel 1 real	0	0	0
channel 1 imaginary	2145300592	2145300592	30.999
channel 2 real	0	0	0
channel 2 imaginary	2145300592	2145300592	30.999

Table 8.1: Results from the simulation of the Hardware part of the DFT preprocessor

The table shows, that the computed values in MATLAB, equal the measured ones. To compute the values in MATLAB, the quantised twiddles are used, to compensate for the quantisation of the twiddles. The results from MATLAB are also truncated to 32 bits, by dividing with 2^4. The similarity between the values from MATLAB, and the measured ones, shows that the computation performed in hardware has the expected functionality. It shall be noted, that an error less than 2^4 cannot be observed, due to the truncation of the word from the accumulator. The imaginary parts of the computation show that an in phase signal utilises the accumulator to 30.999 bits as expected. This indicates that the accumulator will never overflow regardless the signal applied. In the next section the HW/SW interface is verified.

8.2 HW/SW Interface Verification

In the previous section, the hardware was verified. In this section the verification of the HW/SW interface is performed. The section seeks to verify, whether the timing of the interface is satisfied. In section Sec. 7.2 it could not be stated, whether the timing of the interface was satisfied, because the propagation delay from the register to the Avalon Interconnection Fabric, was not stated by the manufacture. It was assumed, that the propagation delay was less than 10 ns, to satisfy the timing for a read cycle. This section verifies this assumption. The test setup is a hardware platform, where the twiddles are shipped to the interface, at the hardware side. At the software side, the twiddles are read and stored in a file. The test setup in Quartus is located at the DVD, which is attached on the backside of the report. The reason for using the twiddles, is that they are known beforehand, which makes it possible to compare the signals, fetched at the SW side of the interface. The test is performed for each channel simultaneously, where the cosine twiddles are applied to both the real and imaginary part of channel 1. The sine twiddles are applied to the real and imaginary part of channel 2. The two equal signals applied at each channel, makes it possible to see if the data of the real or imaginary part, is delayed in the interface. It is not possible to see, if all four signals are delayed. This is not seen as a problem, because it is the timing between the spectra parts, which is important, to obtain the correct complex spectrum. The test is performed for 16384 samples, which is the limit due to the memory space in the NIOS II processor. This means that 16384 a priori known data, are fetched through the interface, and stored at the SW side of the interface. For the test to be successful, all data shipped at the hardware side of the interface, have to be the same, seen from the software side of the interface. This implies that there is no delay between the real and imaginary parts and the channels in between. The signals measured, are tested in MATLAB. The test is performed as a model based test, where the sine and cosine twiddles are generated and quantified. The quantified twiddles are compared with the measured signals in terms of residual

error. The test is expressed as,

$$e_{ch1_R} = E\left[(ch1_R - Q_{cos})^2\right] \tag{8.1}$$

$$e_{ch1_I} = E\left[(ch1_I - Q_{cos})^2\right] \tag{8.2}$$

$$e_{ch2_R} = E\left[(ch2_R - Q_{sin})^2\right] \tag{8.3}$$

$$e_{ch2_I} = E\left[(ch2_I - Q_{sin})^2\right] \tag{8.4}$$

where e denotes the residual error of, for example the real part of channel 1 $ch1_R$. Q is the quantified sine or cosine. If there is missing samples or delay between the channels, the residual errors will be different from zero. All residual errors are zero, which implies that there is no errors in the interface. It can be concluded, that the timing requirements of the interface is satisfied, and that there are no delay in between the channels. In the next section, the real time requirement of the software platform is verified.

8.3 Software Platform Verification

In Sec. 7.3 the implementation of the software platform was explained. This section verifies the timing constrains between the main loop routine, and the hardware interrupt routine. The section also verifies the real time execution of the Covariance/Bayes classifier, to see whether it meets the real time requirement. The real time demand was set as the time between two successive interrupts, which is 4.267 ms. The test of the real time requirement is performed with a real time debugging interface. The interface was developed to see whether real time execution was obtained. The interface consists of three outputs from the NOIS II processor, which are connected to the GPIO at the DE2 board. From the software side, it is possible to set or reset the pins. In the real time test, one of the pins is set at the start of the hardware interrupt routine, and reset at the end. Another pin is set in the start of the main loop, and reset at the end. The data from the two pins are sampled in a logic analyser. In Fig. 8.1 part a and b, the scheduling and execution times for the hardware interrupt routine and main loop routine are shown.
The figure shows that the scheduling between the routines are as expected, because the hardware interrupt is executed before the main loop routine. To satisfy the real time requirement the inequality,

$$t_{total} \geq t_{int} + t_{main} \tag{8.5}$$

has to be true. The sequences sampled with the logic analyser have been analysed, and the execution time for the hardware interrupt routine and main loop routine, have been computed. The time between each execution of the hardware interrupt routine, has also been calculated to verify the spectrum rate F_c. The computations are performed for 469 hardware interrupts, which is approximately 2 seconds.
In Tab. 8.2 the results are shown. It is seen that the hardware interrupt routine is much faster than the main loop routine, which was also expected, due to all the computations performed in the Covariance/Bayes classifier. The table shows that the real time requirement is satisfied, because Eq. 8.5 is satisfied. It is seen that the variance of the main loop execution time is relative larger, compared to the variance of hardware interrupt routine execution time. The variance is caused by the data cache of the NIOS II processor, but is not seen as a problem for the real time execution,

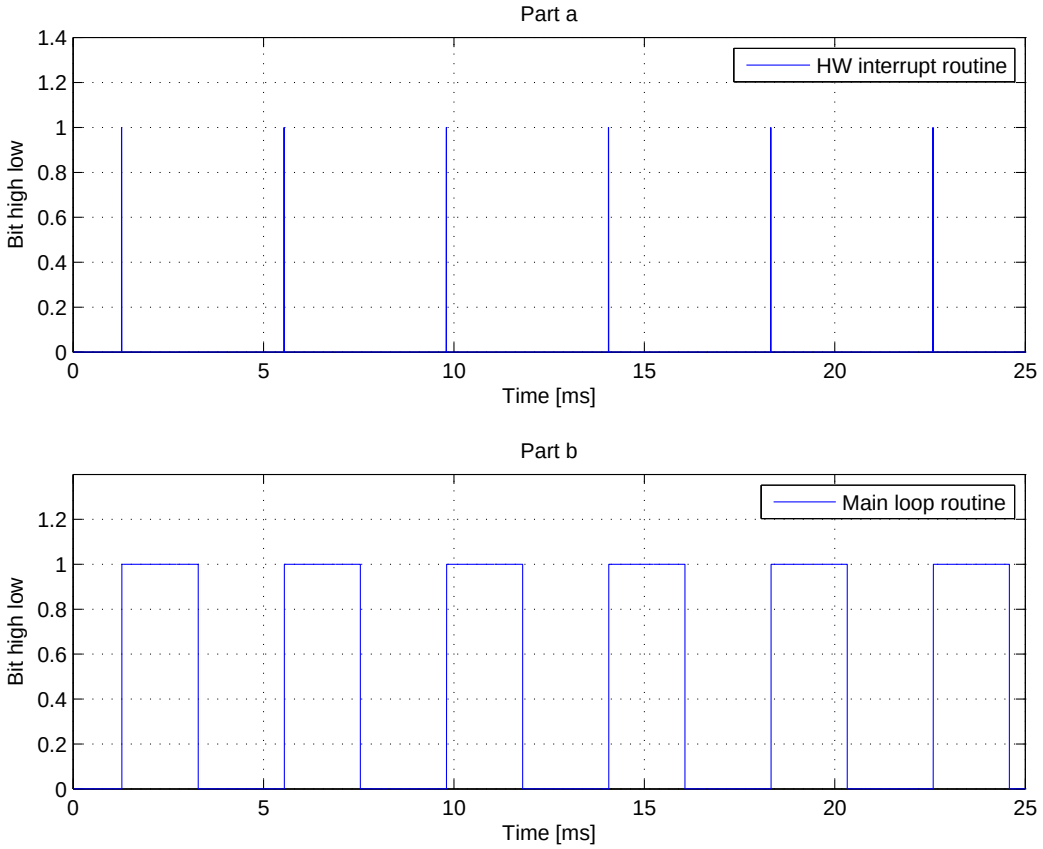

Figure 8.1: Plot of the timing between the hardware interrupt routine and main loop routine

due to the overhead in execution time. The overhead is 2.2418 ms, which is more than the half of the total execution time available. The spectrum sample frequency F_c was chosen be 234.375 Hz . The measured value is 234.747 Hz , which is 0.37 Hz off the correct frequency. The difference is seen as insignificant, because it is less than a sample delay per second. In the next section the classification methods are tested to find out, if it is best to search for the metal, or changes in the product, to detect the metal object.

8.4 Digital Metal Detector Evaluation

The system evaluation concerns the performance of the classification algorithms, where the developed system is used. The classification methods were described in Cha. 3. This section tests the classification methods, to see the performance, limitations and opportunities in the methods. The test is performed with the implemented system described in Cha. 7. The classifiers are trained, as described in Cha. 5. The test is performed for aluminium, AISI 304 and iron spheres in different sizes. The spheres are placed at the conveyer belt, once at a time, at different places. The system is also tested with popcorn, to see if the classifiers classify the popcorn as a metal sphere (False alarm). The reason for choosing a popcorn package, is that the popcorn package contribute to the signal, measured at the receiver coils. The test is performed for each object 20 times. The test is also performed with a popcorn package, and a metal sphere as a mixture, to see the performance in connection with contaminated product. In Tab. 8.4 the results for both the clean spheres test,

Name	Mean [ms]	Variance [ms]
t_{int}	0.0056	$0.23845 \cdot 10^{-6} \approx 0$
t_{main}	2.0125	0.0027
$t_{total_{meas}}$	4.2599	$95.660 \cdot 10^{-9} \approx 0$

Table 8.2: Measured execution time for the hardware interrupt routine and main loop routine. Measured time between the interrupts is also shown,

Classifier	LRAD clean	COV clean	LRAD mix	COV mix
FE 1 mm	0 %	100 %	60 %	0 %
FE 2 mm	100 %	100 %	100 %	100%
FE 5 mm	100 %	100 %	100 %	100%
FE 10 mm	100 %	100 %	100 %	100%
ALU 2 mm	0 %	100 %	25 %	0 %
ALU 3 mm	100 %	100 %	65 %	90%
ALU 5 mm	100 %	100 %	100 %	100 %
ALU 10 mm	100 %	100 %	100 %	100 %
AISI 2 mm	0 %	0 %	20 %	0 %
AISI 3 mm	100 %	100 %	100 %	0 %
AISI 5 mm	100 %	100 %	100 %	85 %
AISI 10 mm	100 %	100 %	100 %	100%
Popcorn	95 %	100 %	X	X

Table 8.3: Results from the system test. The table shows both the results from testing with clean metal spheres, and the mix between a metal sphere and the popcorn package.

and the mixture between sphere and popcorn, are shown. In the following sections the tests are discussed separately.

8.4.1 Clean Metal Spheres

The test with only one sphere present in the detector head, shows that the LRAD classifier cannot detect the 1 mm iron, 2 mm aluminium and 2 mm AISI spheres, and wrongly classifies 5 % of the popcorn packages. The Covariance/Bayes classifier detect all spheres, apart from the 2 mm AISI spheres. It is in general a weakness for both methods to detect the 2 mm AISI spheres. The reason for this, is that the signal from the AISI spheres is week compared to the other spheres. The dependency between the size of the sphere and the signal resulting from the sphere, is seen in Fig. 2.5 in Cha. 2. The reason for the LRAD classifier not detecting the 1 mm iron and 2 mm aluminium spheres, is that the residual error, which is used in the core of the classifier, is less than the detection threshold. The reason for this, is that the signals from these spheres are week as with the 2 mm AISI sphere. This means, that if the signal from the sphere is less than twice the residual error variance of the popcorn package, the method is not able to detect it. The reason for the Covariance/Bayes classifier is detecting these spheres, is that the trajectory of the spheres are still present, and the classifier does not depend of the signal amplitude.

8.4.2 Mix between Metal Sphere and Popcorn

The test, where mixtures of popcorn and metal spheres were moved through the detector, shows that the Covariance/Bayes classifier cannot detect the 1 mm iron, 2 mm aluminium and $2-3$ mm AISI spheres. The reason for this, is that the signal from the popcorn and sphere is mixed in ab additive mixture. If the popcorn is the dominant part in the mixture, the trajectory of the sphere will be distorted by the popcorn trajectory. The distortion degrades the performance of the classifier, because the matched filter chosen to match a metal sphere, does not fit anymore due to the mixture. It is also seen, that for larger sphere sizes, the method is able to detect the spheres, because the spheres are the dominant part of the mixture. It is also seen, that performance degrades at different sizes for the metal types. This is caused by the dependency between material type and signal level.
The LRAD performance is increased, when the popcorn is present together with the metal sphere. This is seen for the 1 mm iron, 2 mm aluminium and 2 mm AISI spheres. The improvement is caused by the mixture of popcorn and metal, because the sum of the week signal from the sphere, is added with the signal from the popcorn, which forces the residual error to be larger than the decision threshold. It shall be noted, that if the complex spectrum of the popcorn is opposite the complex spectrum of the metal sphere, the spectrum measured will have zero amplitude. This can be an explanation for the degrade in performance for the 3 mm aluminium sphere, which was not a problem to detect with the clean spheres test.

In the next section, the points drawn through the system evaluation chapter, are summarised and concluded.

8.5 Conclusion on System Evaluation

In this section the system evaluation is concluded. The chapter started shortly summarising the hardware tests, which showed that the basic building block in the multiplier, the adder, was a factor of 13 times faster than expected. It can be concluded, that the optimisation performed by the complier, improves the speed of the adder. The Hardware part of the DFT preprocessor has also been tested, and the test showed the expected performance. This means that there was no overflow in the accumulator and by applying a orthogonal signal the accumulator zeroed as expected. The HW/SW interface was tested, and it was shown, that the propagation delay from a register to the Avalon Interconnection Fabric, was less than 10 ns, because the HW/SW interface passed the test. If the delay was larger, the interface would fail the test, because the timing of a read cycle was not satisfied. The software platform was tested, and it can be concluded that the scheduling and real time demands for the Covariance/Bayes classifier is satisfied. The test also found, that the spectrum sample rate was 0.37 Hz off frequency. The difference is not seen as a problem, because it equals 0.37 samples per second, which should not degrade the performance of the Covariance/Bayes classifier.

Last in the section the overall system test was performed. The test showed different results. The Covariance/Bayes classifier had the best performance, when the clean spheres were tested. The picture was different for test of mixtures, where the LRAD classifier performed best. It can be concluded, that the choice of classification method, depends of the product, which is present with

the metal sphere. If the product signal is much weaker, than the signals from the small spheres, the Covariance/Bayes classifier is the best choice, but if opposite, the LRAD classifier is the best choice, because the trajectory of the metal sphere is distorted by the product. The performance requirement defined in Cha. 4, was to go below the size of 1% of the detector head opening for iron. The opening of the detector head borrowed from Detectronic A/S is,

$$(W, H) = (370, 170) \text{ mm} \tag{8.6}$$

which converted to radius, which can be compared to radius of the sphere, becomes,

$$R_{opening} = \frac{Circumference}{2 \cdot \pi} = \frac{1080}{2 \cdot \pi} = 171.89 \text{ mm} \tag{8.7}$$

The conversion to a circle is done to compensate for the different between the width and height. The goal is thereby to be able to detect a iron sphere of 1.7189 mm, without food product. Based on the results in Tab. 8.4, it concluded that a 1 mm iron sphere was detected 20 times out of 20 measurements, with the Covariance/Bayes classifier. This means, that the digital metal detector was able to detect 0.58% of the detector head opening. Since the hit rate was 100 %, it is expected that even smaller objects can be detected.

Conclusion

The introduction to this master thesis stated three initial problems. The problems were based on hypotheses, stated by the company Detectronic A/S. This conclusion will clarify the hypotheses, and state answers to the questions. The stated questions are:

1. Do signals from metal objects have a unique phase and amplitude?
2. Is the detection of metal improved, when the frequency is increased?
3. Is it beneficial to move from an analog to a digital implementation?

The thesis started analysing a model of the detector head, by use of Maxwell's equations, and work done by [Yamazaki et al., 2001], [Yamazaki et al., 1998] and [Yamazaki et al., 1996], to seek the answer of the two first questions. It was found through the analysis of the model, that it was not valid in the general case. The limitation in the model was caused by the assumption, that the permeability of an object is independent of the frequency. The model was extended by the use of an exponential decreasing function, describing the dependency between frequency and permeability. The extended model was tested, and the results showed correlation between model and measurements. The amplitude response was verified by [Yamazaki et al., 1996], for the model without extension. It can be concluded that the extended model, derived in Cha. 1, is sufficient to describe the signals from the detector head. It can be argued, whether the amplitude response holds for the extended model, but it is seen as a future perspective to verify this area.

The analysis of the signals, that origin from the detector head, was presented in Cha. 2. The result of the analyses were a signal model, describing the signals from the detector head. It was found that the phase and amplitude of an object depend on size, type and transmitting frequency, which lead to the conclusion, that the amplitude and phase of a metal object are unique, when the size, type and frequency are constant. If the size, type and frequency are changed, the phase and amplitude changes. This answers the first question. The signal model was developed, based on measurements from a metal detector head, and simulations. The measurements and simulations showed an exponential increase in amplitude, when the size of the objects increased. From this it can be concluded, that decreasing the size of the object, makes it harder to detect it, in terms of amplitude. The results also showed, that the amplitude of an AISI 304 sphere is smaller, than both iron and aluminium spheres of same size, which makes it harder to detect in terms of amplitude. Through the development of the signal model, the mixture between two spheres were investigated. The investigation showed, that two objects passed through the detector head are mixed additively. From this it can be concluded, that the objects present in the detector head satisfy the super position principle.

To answer the second question, simulations between product and metal was performed to find out if there was an optimum frequency, where the metal object was the dominant part of the mixture. The simulations showed, that the optimum frequency was strongly dependent on the product. If the product was iron enriched, the optimum frequency was placed around 1 MHz, and if not, the frequency was placed around 2 KHz. The differences lie in the permeability of the different products, which decreases with increasing frequency. It can be concluded, that increasing the frequency, does not necessarily improve the detection of metal objects, but an optimum frequency can be chosen for a specific product, where the SNR_p is maximised.

Through the signal analysis it was observed that a package of unpopped microwave popcorn had a different phase and amplitude trajectory, than the one for the metal objects. This lead to the idea for a classification approach, which tried to search for the metal object through the trajectory. A second method was developed to find changes in the popcorn, and through the changes, detect the contaminated popcorn. The methods were based on opposite hypotheses, which made it interesting to test the performance. The test of the methods showed, that the methods had different performance, which means that the Covariance/Bayes classifier had the best performance, when only the metal spheres where present. When the metal object was present together with the popcorn, the performance of the LRAD classifier increased, and the Covariance/Bayes classifier decreased. The reason for the decrease in performance of the Covariance/Bayes classifier, is due to distortion of the trajectory due to the product. The distortion occur, when the metal object is not the dominant part of the mixture between metal and product. It can be concluded that the choice of classifier depends of the product. If the product is the dominant part of the mixture, the best choice would be the LRAD classifier. If opposite, it would be the Covariance/Bayes classifier.

The classification algorithms were implemented in the Cyclone II FPGA for educational purposes. The focus was put on implementing a flexible hardware and software solution, which can be used for test of different classification algorithms. It was chosen to implement a DFT preprocessor in hardware, and interface the hardware to a NIOS II softcore processor. In the implementation of the DFT preprocessor it was found, that the coefficients could be implemented, such that they were periodic. The periodicity introduced the property, that the fundamental frequency of the coefficients do not change regardless of the quantisation of the coefficients. The cost of quantisation is a reduction of power in the fundamental frequency, which is spread to the harmonic frequencies. In the hardware implementation, the main focus was put on the delay through the Multiplier, which was optimised with a pipeline, because the timing constrains was not satisfied. The Adder, which is the basic building block in the Multiplier, was simulated in Quartus, which showed that the timing was approximately 13 times faster than expected. The improvement origins from optimisations done by the synthesiser of Quartus. It was chosen to implement the Multiplier in VHDL, which from this point of view is time consuming, without gain in performance. It can be concluded, that the optimal solution would be to use a Megafunction Multiplier made by Altera.

To answer the last question, whether a digital solution is advantageous, the results from the system evaluation is needed. The results show that iron spheres of 0.58% of the opening can be detected. The performance measure was to go below 1%, which leads to the conclusion, that the concept of

a digital metal detector is found effective, and makes it a subject for further development.

Project Perspective

In this chapter the project is taken into perspective. The authors have spend time on analysing the three-coil detector head, which have lead to results, that can be used for optimising the detector according to a specific product. The results obtained from analysing the three-coil detector head, are put in the context of future development. The purpose of the chapter is thereby to emphasise interesting aspects of the project, which the authors of the report, have not managed to investigate due to time limitations. The chapter covers nine aspects, which are discussed separately. The aspects cover perspectives for the three-coil detector head, for the classification methods, and for the implemented system. Also ideas for future development are presented.

Generalisation of Three-Coil Detector Model

Solving Maxwell's equations, without assuming the displace current term $\omega\epsilon$ in Eq. 1.11 to be negligible, results in an unrestricted and general model for the three coils. The generalisation extends the model to be valid for any material and any frequency. For analysis of products in the food industri, it is not assumed to be necessary, but if the use of the model is extended, it may be necessary perform the generalisation.

Verification of Permeability Function For Amplitude

The approximated permeability function, developed to compensate for the frequency dependent permeability in Sec. 1.3, has not been verified for the amplitude values. In the report, verification for the phase has been performed, but the amplitude values were assumed to hold, based on [Yamazaki et al., 1996]. Analyses may show, that compensation for change in amplitude is necessary.

Optimal Dimensions For the Three-Coil Detector Head

It should be possible to obtain optimal dimensions for the detector head coils, when a specific food product is given. Usually, the size of the detector head opening is defined according to the size of a specific product. However, the distance between the coils, and the number of turns in the coils, to mention some, are chosen due to practical experience at e.g. Detectronic A/S. The model should be able to reveal an optimal parameter, when the rest are fixed. If e.g. the distance between the three coils is left as a variable L, and the rest are fixed, the problem to be solved could be expressed as,

> Find L, such that the absolute value for the induced signal becomes maximum.

This problem can be solved from derivations based on the model, or simply by iteration. The problem can be stated for any variables in the model, and the one presented here, is only an idea.

The result of optimal coil dimensions, is expected to improve the sensitivity for the object in the detector head, but no arguments are given.

Effects of Transmitting Field Strength

Analysing the current in the transmitting coil may help to choose an optimum current amplitude. The amplitude of the current is reflected in magnetic field strength, and intuitively the largest possible field strength is desired. However, because of saturation in magnetization process for ferromagnetic materials, see Sec. 1.3, there may be an upper limit for the field strength, where no gain is obtained by going beyond.

The Four Maxima Phenomenon

Based on Cha. 2, it was discovered that the food products analysed had four maxima in the amplitude trajectories, and three shifts in the phase. The reason here for, has not been found during the project, but it would be interesting to determine the underlying explanation.

Product Compensation

Compensation for (removal of) the food product in the trajectory is possible, because of the deterministic behaviour. It is known, based on the model, that first order linear regression describes the trajectory, and this knowledge may be utilised. The lower limit for the product removal, is the product variance, which is caused by the product not being in the same position, each time it is passed through the detector head. The product compensation gives the best possible conditions for metal detection, because the product is suppressed as much as possible, in the signals.

System For Product Analysis

In Cha. 2 it was shown that an optimal frequency can be found for the specific food product. It would be essential to develop a system, capable of determining this frequency, to obtain a *product optimal* metal detector. A possibility is to perform a frequency sweep, while the product is in the metal detector head. Doing this, requires another approach, than the one with a capacitor in series/parallel with the coils. This is because the series/parallel LC form a resonance circuit, which limits the sinusoidal signal to the specific frequency. The hardware DFT approach is also focused on the specific frequency, but this can be solved by mixing the sweeped signal into the specific frequency. This can be done, by making the multiplying frequency in the mixer, follow the sweeping frequency, with a frequency offset corresponding to the desired/specific frequency. In this way, the output of the mixeer can be fixed in the specific frequency, regardless of the sweeping frequency. This idea has not been examined by the authors, but reflect ideas obtained in the last part of the project period.
It would be advantageous to combine the design of the product analysis system, with the design of a high frequency system.

High Frequency Metal Detector

In Cha. 2 it was found, that the optimal frequency for the specific dry food simulated, was about 1 MHz. It could be interesting to implement a metal detector utilising this frequency, to see if

performance is gained. The authors have obtained practical experiences, showing that the increase in transmitting frequency, results in a larger amplitude for both the product and the metals, which is also in correlation with the results of App. E.2.
Increasing the frequency, results in harder real time requirements. The high frequency in connection with the DFT method gives no improvements, why the signals can be mixed down, and sampled at a lower rate, than at the high frequency.

Improved Classification Methods

The system implemented in this report, is considered as a flexible hardware and software platform, so other classification methods can be implemented and evaluated.
The DFT preprocessor is, based on practical experiences, more than necessary, to obtain precise amplitude and phase estimates. Due to the practical experiences, the DFT method is observed to represent the noise floor of the system, under which no gain in detection/classification performance is obtained. After the amplitude and phase is obtained, filtering the phase noise will ofcourse improve performance. The flexible platform, makes it possible to develop complex classification methods, where it may be necessary to develop custom instructions for the NIOS II softcore, instead of a stand alone hardware implementation. The next iteration of the metal detector project, is therefore urged to focus on development of methods for classification, based on the flexible platform developed in this report.

This chapter ends the master thesis, concerning the Concept Development for the Next Generation of Metal Detectors.

Bibliography

T. A.-. P. G. 08gr1041, AAU 2008. 'New Generation Metal Detector for Food' by Frank Thornemann Hansen & Lars Vinding, Graduate Master Students, 08gr1041@es.aau.dk.

Altera. *Nios II Software Developer's Handbook*. Altera, 8.0.0 edition, 2008a.

Altera. *Using Nios II Floating-Point Custom Instructions Tutorial*. Altera, 2006.

Altera. *Avalon Interface Specifications*. Altera, version 1.0 edition, 2008b.

A. Devices. *Monolithic A/D Convertes AD9220/AD9221/AD9223*. Analog Devices, rev. e edition, 2003.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., 2nd edition, 2001. ISBN 0-471-05669-3.

H. Ebert. *Quasistatiske elektriske og magnetiske felter*. AAU, 3th edition, 1998.

H. Ebert and P. Raskmark. *Grundlæggende Maxwellsk Feltteori*. AAU, 4th edition, 1998.

S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 4th edition, 2002. ISBN 0-13-090126-1.

A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. K. Laine, and J. Huopaniemi. Frequency-warped signal processing for audio applications. *J. Audio Eng. Soc., Vol. 48, No. 11, 2000 November*, 2000.

J. D. Kraus and D. A. Fleisch. *Electromagnetics With Applications*. Mc-Graw Hill, 5nd edition, 1999. ISBN 0-07-289969-7.

W. MathWorld, November 2007. Spherical Harmonic Addition Theorem at `http://mathworld.wolfram.com/SphericalHarmonicAdditionTheorem.html`.

A. V. Oppenheim, R. W. Schafer, and J. R. Buck. *Discrete-Time Signal Processing*. Prentice-Hall, 2nd edition, 1999. ISBN 0-13-083443-2.

S. M. Ross. *Introduction to Probability and Statistics for Engineers and Scientists*. Elsevier Academic Press, 3rd edition, 2004. ISBN 0-12-598057-4.

R. A. Serway and J. John W. Jewett. *Physics for Scientists and Engineers, with Modern Physics*. Thomson Brooks/Cole, 6th edition, 2004. ISBN 0-534-40949-0.

K. S. Shanmugan and A. Breipohl. *Random Signals Detection, Estimation and Data Analysis*. John Wiley and Sons, Inc., 1st edition, 1988. ISBN 0-471-81555-1.

M. R. Spiegel and J. Liu. *Mathematical Handbook of Formulas and Tables*. McGraw-Hill, 2nd edition, 1999. ISBN 0-07-038203-4.

P. Stoica, H. Li, and J. Li. A new derivation of the apes filter. *IEEE Signal Processing Letters, Vol. 6, NO. 8, August 1999*, 1999.

J. F. Wakerly. *Digital Design - Principles and Practices*. Prentice-Hall, 3rd edition, 2001. ISBN 0-13-090772-3.

I. Wikimedia Foundation, March 2008. URL: http://en.wikipedia.org/wiki/Multiplication_ALU.

S. Yamazaki, H. Nakane, T. Neegishi, and A. Tanaka. Simultaneous measurement of electric and magnetic properties of a spherical sample. *IEEE Transactions on Instrumetation and Measurement*, 1996.

S. Yamazaki, H. Nakane, T. Neegishi, and A. Tanaka. Simultaneous measurement of size and electromagnetic property of multilayered spherical sample. *IEEE Transactions on Instrumetation and Measurement*, 1998.

S. Yamazaki, H. Nakane, and A. Tanaka. Basic analysis of a metal detector. *IEEE Instrumetation and Measurement*, 2001.

J. Zheng, K. W. K. Lui, W. K. Ma, and H. C. So. Two simplified recursive gauss-newton algorithms for direct amplitude and phase tracking of a real sinusoid. *IEEE Signal Processing Letters, Vol. 14, NO. 12, December 2007*, 2007.

Nomenclature

Name	Description
$s_{OoB}(t)$	Out-of-balance component in continous time
$s_{OoB}[n]$	Out-of-balance component in discrete time
$d[n]$	Discrete time signal from transmitting coil
$d(t)$	Continious time signal from transmitting coil
$D[q]$	Complex discrete time containing phase and amplitude of $d[n]$
$u[n]$	Discrete time signal from receiver coils
$u(t)$	Continious time signal from receiver coils
$U[q]$	Complex discrete time signal containing phase and amplitude of $u[n]$
$X[q]$	Complex discrete time signal from phase and amplitude estimator
$t_{pd,name}$	Propagation delay for *name*
A	Amplitude of $X[q]$
ϕ	Phase of $X[q]$
n	Discrete time index in the time domain
t	Continous time
q	Discrete time index for the present instance of $X[q]$

Abbreviations

Abbreviation	Description
A/D	Analog-to-Digital
ADC	Analog-to-Digital Converter
AISI	Stainless steel (aisi 304)
ALU	Aluminium
API	Application Program Interface
CLK	Clock
Covariance/Bayes classifier	Covariance based feature extractor, and classifier based on Bayes decision rule.
CS	Chip Select
DAS	Data Acquisition System
DFT	Discrete Fourier Transform
FAPES	Forward Amplitude and Phase Estimation
FE	Iron
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input and Output
HAL	Hardware Abstraction Layer
HW	Hardware
HW/SW	Hardware/Software
IRQ	Interrupt Request
IP	Independent Platform
ISR	Interrupt Service Routine
MAC	Multiply-and-Accumulate
LMMSE	Linear Minimum Mean Square Error
LRAD	Linear Regression with Amplitude Detection
LSDFT	Least-Squares Discrete Fourier Transform
NIOS2-IDE	NIOS II Integrated Development Environment
OoB	Out-of-Balance
RGN-GCMA	Recursive Gauss Newton - Gradient Correlation Matrix Approximation
RTL	Register Transfor Level

Abbreviation	Description
SNR	Signal-to-Noise Ratio
SoC	System-on-Chip
SOPC	System-On-Programmable-Chip
STFT	Short Time Fourier Transform
SW	Software
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
XCORR	Cross correlation

Part III

Metal Detection System - Appendix

Appendix A

Simulation Journals

This appendix contains various simulation journals, which support statements and decisions made in the report. The specific references to the simulation journals are found in the report.

A.1 Phase and Amplitude Estimation Methods Performance Simulation

A.1.1 Purpose

The purpose of the simulation is to find the phase and amplitude estimation method, which performs best under noisy conditions.

A.1.2 Simulation Setup

In Fig. A.1 the execution of the simulation is shown. The simulations starts with generating two signals of 1000 sample, with a sample frequency of 1 MHz, and a frequency of 100 KHz. The blocks of 1000 samples are phase shifted $\frac{\pi}{5}$, between them and have a gain difference of two. Both signals are applied with a normal distributed random noise, with a specific SNR. The signal blocks are shipped to the methods, which produce an amplitude and phase estimate. The estimation is perform a 1000 times, with new generated signal and noise components. The phase shift, gain and SNR are unchanged for all 1000 estimations.
When 1000 estimations are performed, the mean and standard deviation of the gain and phase estimates are calculated for all methods. Afterwards the SNR is changed, and the simulations are performed a 1000 times again, with the new SNR and the unchanged gain and phase shift.

The simulations are performed for a set of SNR's, and all means and standard deviations are saved for all methods, and for all SNR's. All simulation parameters are listed below.

$$\text{Number of estimations for each SNR} = 1000 \tag{A.1}$$

$$\text{Estimation block length} = 1000 \tag{A.2}$$

$$\text{Phase shift} = \frac{\pi}{5} \tag{A.3}$$

$$\text{Gain} = 2 \tag{A.4}$$

$$\text{Sample frequency} = 1000000 \tag{A.5}$$

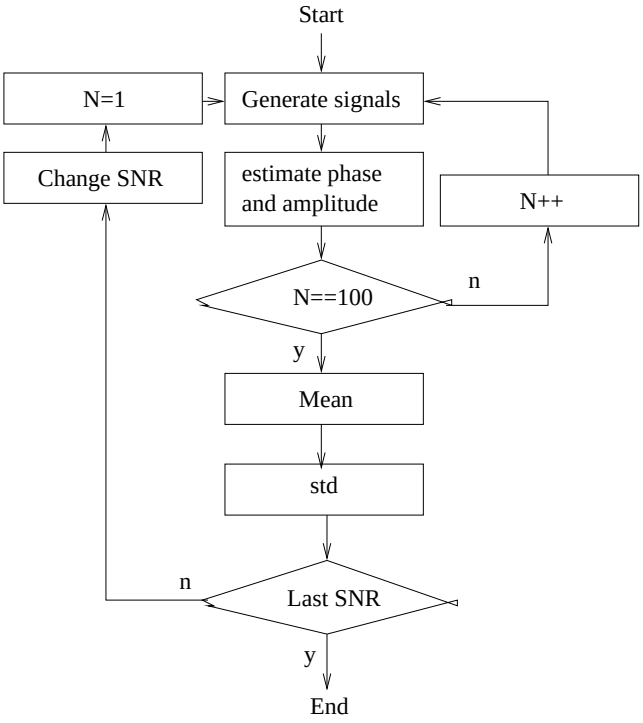

Figure A.1: Block diagram of the execution of the phase and amplitude estimation simulation

$$\text{Signal frequency} = 100000 \tag{A.6}$$

$$\text{SNR interval} = [-20 : 3 : 30] \tag{A.7}$$

A.1.3 Result Processing

The calculated means are converted to a mean error, which is expressed as,

$$e_{g,m,s} = \mu_{g,m,s} - G \tag{A.8}$$

$$\text{and} \tag{A.9}$$

$$e_{p,m,s} = \mu_{p,m,s} - \phi \tag{A.10}$$

where e denotes the error, g the gain mean, p the phase mean, m the method, s the SNR, ϕ the phase shift, and G the gain. The figures holding the results, are shown in Cha. 3.1.

A.2 Influence of Over Sampling Rate in Phase and Amplitude Estimation

A.2.1 Purpose

The purpose of the Over Sampling Rate (OSR) simulation is to find out, which influence over sampling has on the DFT phase and amplitude estimation.

A.2.2 Simulation Setup

The simulation setup is shown Fig. A.2.

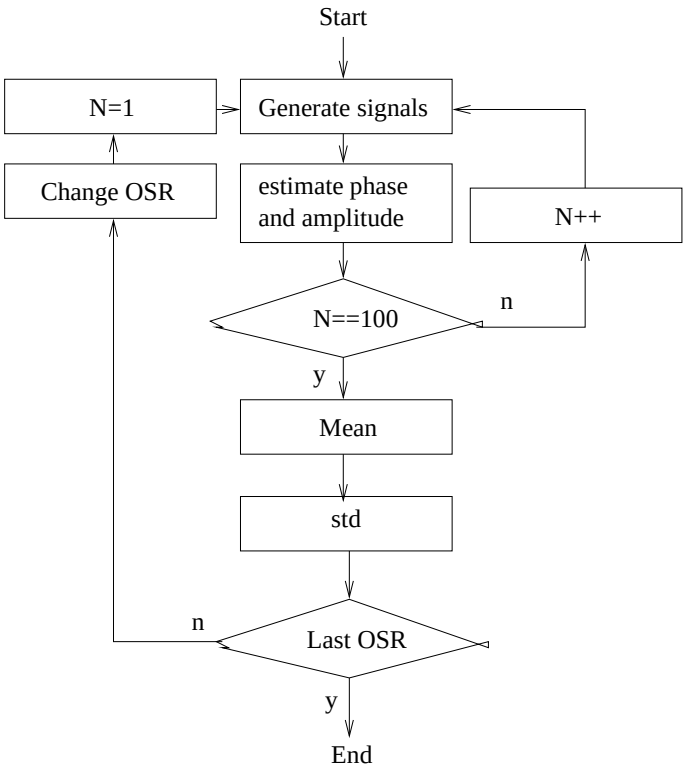

Figure A.2: Block diagram of the execution of the oversampling simulation.

The phase and amplitude are estimated 100 times for every over sample rate. The phase and amplitude of the signal, are estimated from the two sinusoidals, which has a gain of 2 between them, and a phase shift of $\frac{\pi}{5}$. The signal length is calculated to contain five periods of the sampled signal. The signals are added with a additive noise component of 6 dB SNR on both signals. The reason for adding noise to the signals is, that the over sampling rate only has an influence, when noise is present at the signals. When 100 simulations are performed, with the same over sampling rate and SNR, the mean and standard deviation of the estimates are calculated. After the mean and standard deviation calculation, the over sampling rate is changed, and the estimation step is executed again. The sample space of the simulation is $[1:1:1000]$.

The over sampling rate is defined as,

$$k = \frac{F_s}{2 \cdot B} \tag{A.11}$$

where k are the over sampling rate, F_s is the sampling frequency, and B is the highest frequency component present in the signal. The signal length for the estimations can be calculated as,

$$S_l = 5 \cdot k \cdot 2 \cdot B \tag{A.12}$$

with the condition, that 5 periods of the B frequency, should be present in the signal.

The effects of increasing the sample frequency, when the length of the observation window is fixed, results in an increase in the frequency resolution. The frequency resolution is the distance between

the frequency bins in the frequency spectrum. Focusing on a single bin, the effect of increasing the sample frequency, is that the specific bin is split into more bins. In the case of a sinusoid represented in exactly one bin, the effect of increasing the sample frequency, is that the noise power present in the bin, is spread out into the surrounding bins. This is valid, when the sinusoid is represented in the specific bin only.

A.2.3 Results

Fig. A.3 and Fig. A.4 show the phase and amplitude estimation results respectively. Fig. A.3 shows, that the standard deviation and mean estimation error of the phase estimation decreases, as the over sampling rate is increased. Fig. A.4 shows, that the amplitude estimation and standard deviation decreases as the over sampling rate is increased.

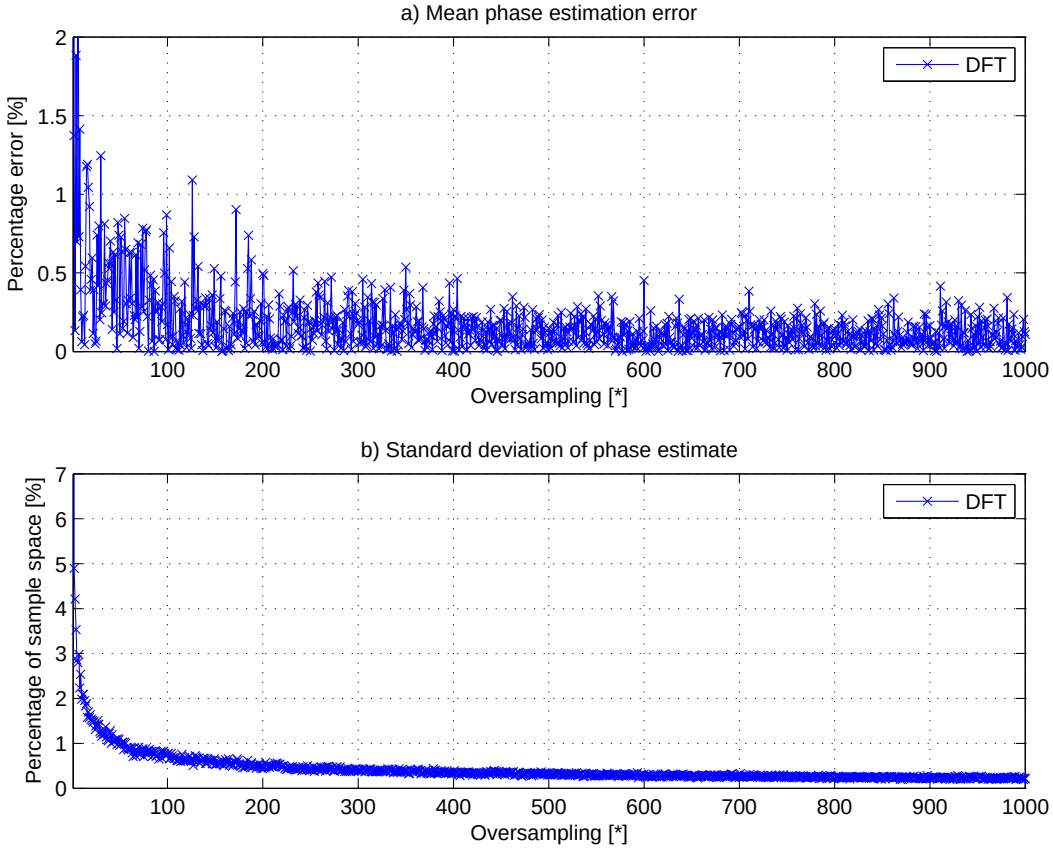

Figure A.3: Plot of the mean error and standard deviation of the phase estimation as function of over sampling rate.

A.2.4 Conclusion

The standard deviation is a way of measuring the precision of an estimate, because the deviation expresses the spread of the estimates around the mean value. When the spread is low, the estimation precision is improved, under the condition, that the mean error is zero. The simulation shows, that the mean error decreases, when the over sampling rate is increased. The standard deviation also

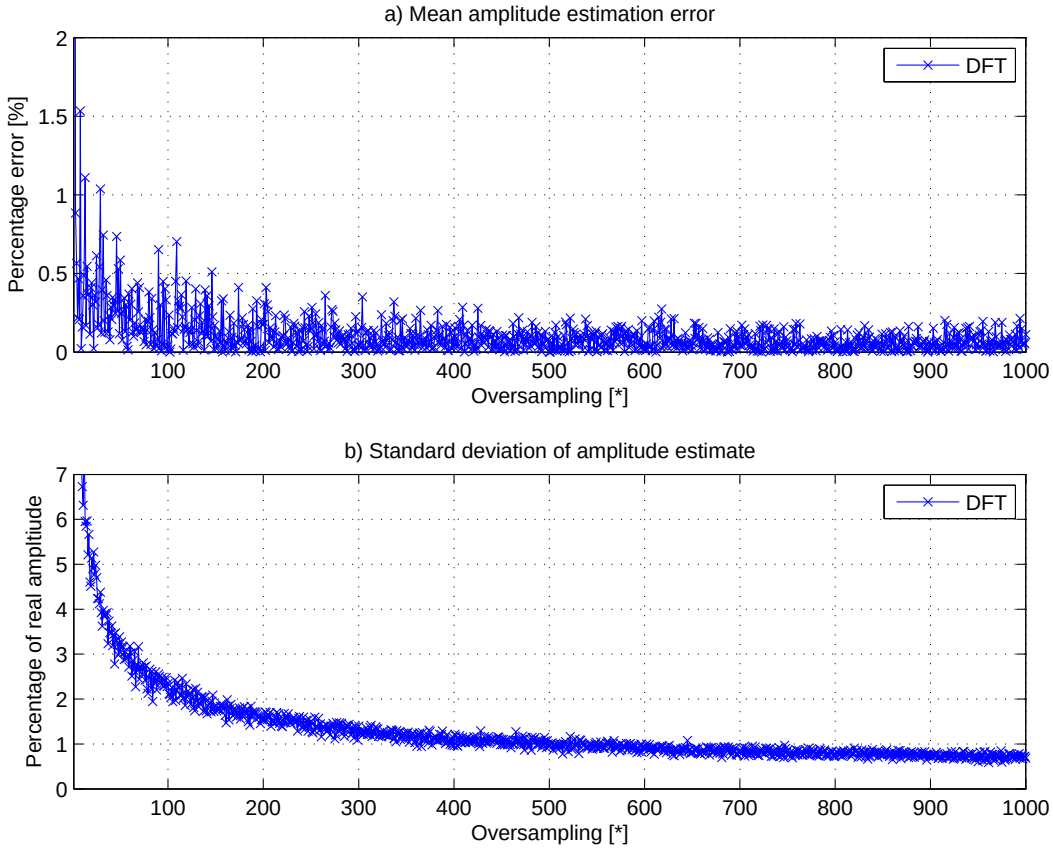

Figure A.4: Plot of the mean error and standard deviation of the gain as function of the over sampling rate.

decreases in phase and gain estimate, when the over sampling rate is decreased. It can be concluded, that the precision of the estimation is increased, when the over sampling rate is increased.

A.3 Determination of Minimum Filter Length

The minimum filter length for the covariance based feature extractor is determined in this appendix. The appendix is being referenced from the System Specification chapter in the design report, see Cha. 4.

A.3.1 Purpose

The simulation has the purpose to find the minimum filter length needed in the covariance calculation.

A.3.2 Simulation Execution

The filter used in the covariance, for seeking the metal, is obtained from the real and imaginary parts of the complex spectrum $X[q]$. The spectrum $X[q]$ represents the amplitude ratio and phase difference, between the reference signal in the transmitting coil and the observation signal from the receiving coils taken differentially, when an object is present in the metal detector head.

To evaluate the minimum length of the covariance filter, corresponding to the number of complex spectra needed to represent the trajectory, when an object is moved through the metal detector head, the maximum value of the covariance is analysed. The maximum covariance value is obtained, when the covariance filter and trajectory of the moving object in $X[q]$ is aligned in time.
The metal detector head model, see Cha. 1, is used for generating both the covariance filter and trajectory. To the trajectory random noise (white) is added in various signal-to-noise ratios (SNR)[1]. The calculation of covariance is performed 500 times for each SNR, and filter length with different noise sequences added.
The covariance filter is obtained for an object of size 10 % and the trajectory for an object of size 2.5 %.

A.3.3 Results and Discussion

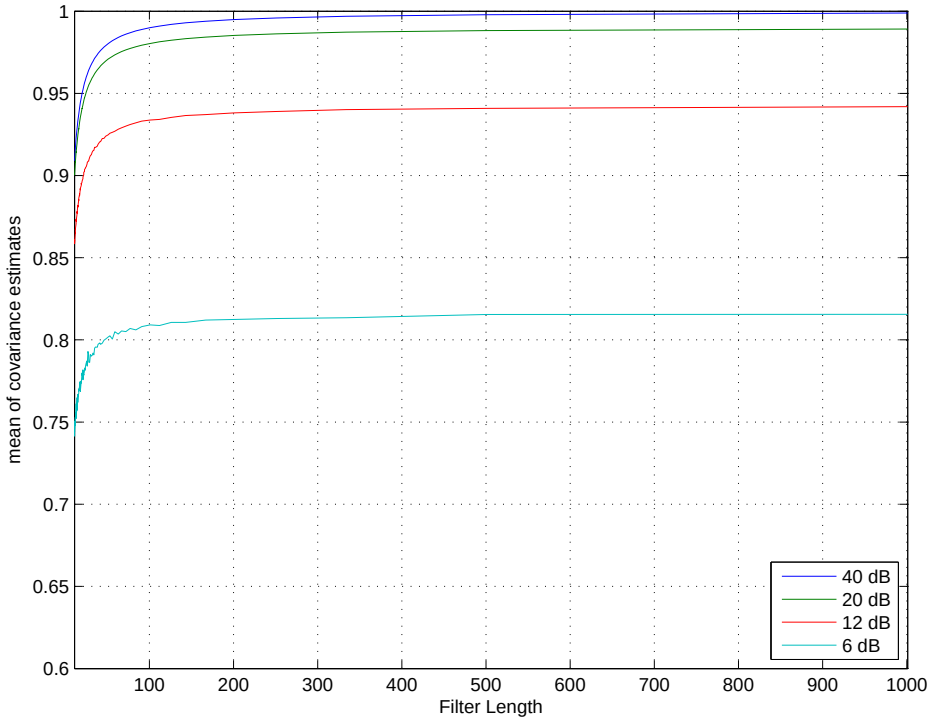

Figure A.5: Illustration the mean of covariance estimates for filter lengths from 7 to 1000 samples.

Fig. A.5 show that the estimate of the covariance is dependent of the filter length, because the covariance in mean converges as the filter length is increased. The convergence point is different for the different SNR's, which are caused by the SNR ratio. The point of interest in the figure, is the filter length, for which the estimate in mean is nearly converged. It is seen, that as the filter length is increased, the convergence speed, becomes slower. This leads to the conclusion that an increases from 20 to 50 taps in the filter, enhances the estimate more than an increase from 50 to 80. In the convergence discussion, the standard deviations of the estimates have to be taken into account, because the standard deviation describes the spread around the mean estimates. It is wanted to have a spread close to zero, because it makes the estimates independent of the noise.

[1]Defined as the ratio between the signal variance and noise variance.

The standard deviation of the estimates are shown Fig. A.6.

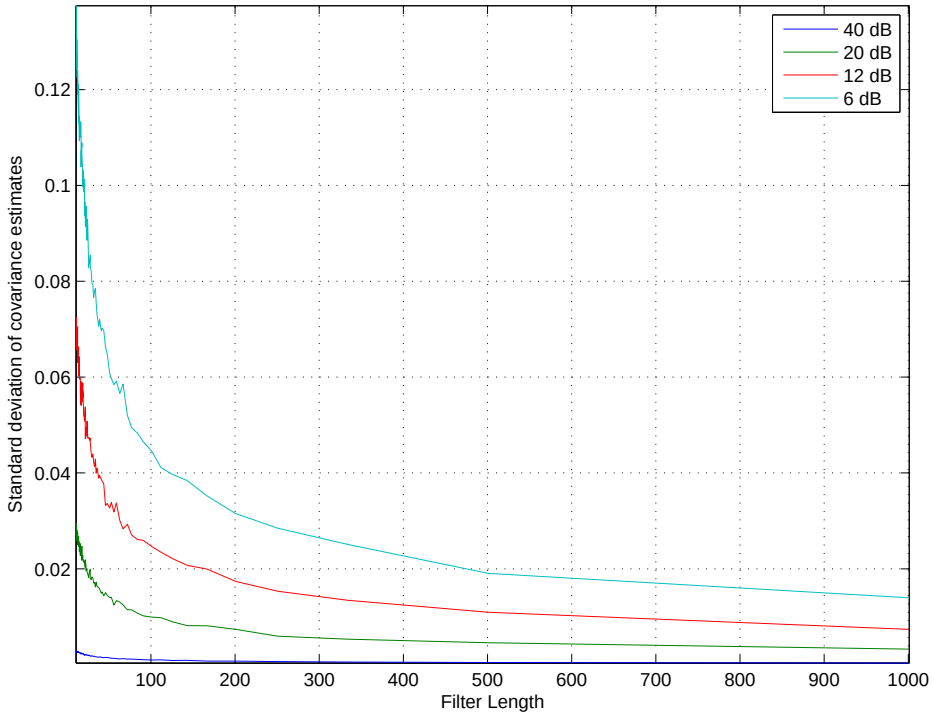

Figure A.6: Illustration the variance of the covariance estimates for filter lengths from 7 to 1000 samples.

The figure shows, that the standard deviation is dependent of SNR, because the standard deviation for the 6 dB SNR simulation is larger than the standard deviation for the 20 dB SNR simulation. The figure also shows, that an increase in filter length, decreases the standard deviation of the estimates.

A.3.4 Conclusion on Filter Length

Based on the discussion it can be concluded, that the minimum filter length depends on the SNR in the system, and the allowable mean estimation error and standard deviation. It is concluded that filter length has to be at least 100 samples, to get a mean estimation error close to zero, and a standard deviation below 0.05 at 6 dB SNR. A standard deviation of 0.05 equals a deviation of 5 % compared to sample space between 0 and 1. It shall be noted, that an increase in filter length enhances the estimates in terms of deviation and mean error.

Appendix B

Measurement on Detector Head Model

This chapter documents the measurement on the metal detector head model made in connection with verification of the mathematical model, see Cha. 1. The purpose of the measurement is to obtain the phase between the current in the transmitting coil, I_x, and the induced voltage in the receiving coils, V_s. The induced voltage is measured differentially across the terminals of the receiving coils.

B.1 Theory

The detector head model is assumed to be balanced, such that the differential voltage across the receiving coils, without an object, becomes zero. Thereby introducing an object in the proximity of the transmitting coil, should produce a phase shifted version of the sinusoid in the transmitting coil in the receiving coil. The phase between I_x and V_s is determined by the physical properties of the material, which furthermore depend on the frequency of the sinusoid. Thereby it is necessary to obtain the phase shift for different frequencies depending on the material

Diamagnetic and Paramagnetic materials, such as brass and aluminium respectively, loose their magnetic properties quickly, i.e. the permeability tends to one, when the frequency is increased. These materials should be examined intensively in the low frequencies up to about 20 KHz.

Ferromagnetic materials, such as iron and ferrite, maintain their magnetic properties higher in frequency. The lower frequencies should be examined less intensive than the higher frequencies, such that frequencies above 10 KHz is examined intensively.

B.2 Environment of Measure

The time and place of measurement is given in Tab. B.1.

Time	Friday the 26th of October 2007
Place	Laboratory at Fredrik Bajersvej 7B
Temperature	$\approx 20^\circ$ celcius

Table B.1: Information about environment of measurement on detector head model.

B.3 Equipment and Setup

The equipment used in the measurement is given in Tab. B.2 all placed in the location given above.

Equipment	**Reference**	**Name**	**AAU-Nr**
Oscilloscope	SCO	Agilent 54621A	33850
Function Generator	FCG	Philips 08559	08559
Multimeter	MMT	Fluke 189	52832
Power Supply	PSU	Hamag HM7042	33888
Power Amplifier	PA	-	-
1 Ω Resistor (3 Watts)	RIV	-	-

Table B.2: Equipment and other components used in the measurement.

The impedance of a coil varies with frequency and becomes smaller as the frequency approaches zero. Thereby loading the FCG with a low impedance requires a large current. Therefore a power amplifier is built, consisting of inverting buffer followed by a push-pull stage of BJT transistors. More details are found in Fig. C.2 in App. C, but the complete documentation is found in the documentation of 08gr1041 [AAU 2008].

The materials (objects) used in the measurement are given in Tab. B.3.

Material	**Permeability μ_r**	**Conductivity σ**	**Permittivity ϵ**
Brass	0.9999	$1.6 \cdot 10^7$	0
Aluminium	1.000023	$3.5 \cdot 10^7$	0
Iron	4000	$1.0 \cdot 10^7$	0
Ferrite	8000	$1.0 \cdot 10^{-8}$	14

Table B.3: Materials (objects) used in the measurement. The values are for low frequencies and approximated, as they vary with the type of material.

B.4 Obtaining Data Set

The object of interest is placed in the detector head model, such that the induced voltage in the receiving coils is greatest - this should be in the center of *one* of the receiving coils. This is done by monitoring the output voltage swing in channel 2 of SCO. The frequency of the sinusoidal waveform generated in FCG is fixed to a specified value and the voltage on the transmitting side is adjusted such that the current is approximately 0.5 A. The current is measured by MMT. Then SCO is adjusted in time and amplitude, such that both channels are represented by at least one period at

the specified frequency. Now the difference in time (μs) between the two channels can be read by placing the oscilloscope cursors at the maximum of each sinusoid. Channel two is always measured to the left of channel one, meaning that channel two behind in time. This is repeated for each frequency and object of interest.

B.5 Obtained Data Set

The data set obtained for each material is given in

- Tab. B.4 for the frequencies between 0.1 and 4.5 KHz
- Tab. B.5 for the frequencies between 5 and 45 KHz
- Tab. B.6 for the frequencies between 50 and 500 KHz

Frequency [KHz]	**Time Delay** [μs]			
	Aluminium	**Brass**	**Iron**	**Ferrite**
0.1	-	1250	7000	7360
0.5	720	780	-	-
0.7	480	-	-	-
1	320	324	664	748
1.2	260	-	-	-
1.4	224	-	-	-
1.6	196	190	-	-
1.8	170	-	-	-
2	148	144	324	372
2.2	132	-	-	-
2.4	120	-	-	-
2.6	112	110	-	-
2.8	98	-	-	-
3	92	92	210	248
3.5	79	-	-	-
4	69	-	156	-
4.5	61	-	-	-

Table B.4: Data set for measurement on detector head model at frequencies in between 100 Hz and 4.5 KHz.

B.6 Discussion

The data sets are now obtained for aluminium, brass, iron and ferrite respectively. The data presents the phase between the current in the tranmitting coil and the induced voltage in the receiving coils measured differentially. The values are read manually on the screen of the oscilloscope, which is expected to be the largest contributor of errors. The reading of the screen is difficult because of fx. radiation of noise into the coils. This is especially the case for low frequencies, where the voltage induced in the receiving coils is only a few millivolts.

Frequency [KHz]	**Time Delay** [μs]			
	Aluminium	**Brass**	**Iron**	**Ferrite**
5	55	53	124	149
5.5	49	-	-	-
6	45	-	100	-
6.5	41	-	-	-
7	38	-	85	-
7.5	36	36	-	-
8	34	-	73	-
8.5	31	-	-	-
9	29	-	65	-
9.5	27	-	-	-
10	26	27	57	75
12	22	-	46	-
14	19	-	39	-
16	16	-	34	-
18	14	-	29	-
20	13	12.8	26	-
25	-	-	20	-
30	9	-	16	-
35	-	-	13	-
40	7	-	11	-
45	-	-	9.5	-

Table B.5: Data set for measurement on detector head model at frequencies in between 5 and 45 KHz.

Frequency [KHz]	**Time Delay** [μs]			
	Aluminium	**Brass**	**Iron**	**Ferrite**
50	5.5	5.4	8.5	14.5
60	-	-	6.5	-
70	-	-	5.5	-
80	-	-	4.5	-
90	-	-	4	-
100	3	2.5	3.5	7.5
125	-	-	3	-
150	-	-	2	-
175	-	-	1.7	-
200	1.3	1.2	1.5	-
250	-	-	1	2.5
300	-	-	0.8	-
350	-	-	0.7	-
400	-	-	0.64	-
450	-	-	0.55	-
500	0.5	0.48	0.50	1.4

Table B.6: Data set for measurement on detector head model at frequencies in between 50 and 500 KHz.

Appendix C

Measurement on Metal Detector

The observation signal $u(t)$ defined as the difference between the induced voltages in the receiving coils, and the reference signal is the current in the transmitting coil $d(t)$.

C.1 Metal Detector Head and Conveyor Belt

A real metal detector including a conveyor belt has been borrowed from Detectronic A/S. The dimensions of the opening of the metal detector head is 360x170x170 mm (HxWxL), as depicted in Fig. C.1.

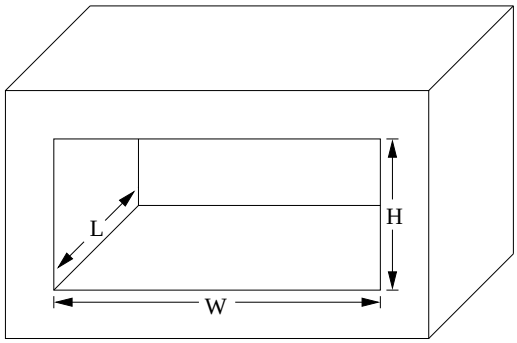

Figure C.1: Illustration of detector head and the dimensions of the borrowed metal detector head. The dimensions are 370x170x170 mm (HxWxL).

A conveyor belt passed the objects through the detector head opening. The speed of the conveyor belt is approximately 25 meters per minute. In the following section the Data Acquisition System developed for measurements is described.

C.2 Data Acquisition System (DAS)

To be able to analyse the signals from the metal detector head a Data Aquisition System (DAS) is developed. The DAS is based on the Measurement Computing PCI-DAS4020/12 Analog & Digital I/O Board, which is installed in PC with a Debian Linux distribution. The sample board is borrowed from Detectronic A/S. No drivers for the sample board have been supplied, so Comedi

(Linux Control and Measurement Device Interface) version comedilib-0.7.11 is used for development of DAS software. The Linux distribution is a Debian 2.6.22-2-686.
Comedi is developed as open-source drivers, tools and libraries for data acquisition. Information and documentation about Comedi is found at *http://stm.lbl.gov/comedi/*.
The following specifications are based on PCI-DAS4020/12:

- Four 12 bit analog inputs (two is needed).
- Input voltage range ±1 V and ±5 V, which is software configurable.
- The sample rate is up to 20 MHz in two channels and the lower bound is 2 KHz, which is also software configurable.

The PC used, has the following specifications:

- Core 2 Duo, 2 GHz CPU.
- 3 GB PC6400 DDR2 RAM.
- 160 GB SATA Harddrive.
- A free PCI-32 slot for the PCI-DAS4020/12 sample board.

The sample frequency f_s is chosen to 20 MHz and two channels is needed for the reference signal $d(t)$ and the observation signal $u(t)$. The precision B of the sample board is 12 bits in unsigned integer (UINT), but according to the user's guide 5.5 times the LSB is lost in noise from the board it self. Therefore approximately 10 bits are assumed usefull.
The data rate DR of the data acquisition system becomes,

$$DR = \#CH \cdot \frac{f_s \cdot B}{10^6} \text{ [megabit/sec.]} \quad \text{(C.1)}$$

$$= 480 \text{ [megabit/sec.]} \quad \text{(C.2)}$$

corresponding to 60 megabytes per second. Under the data acquisition the samples should be transferred into the PC RAM and then stored on the harddrive when acquisition is done. Using the 20 MHz sample frequency in two channels, data acquisition can be performed continuously. The purpose of the DAS is however to obtain shorter sequences of the signals, when an object (e.g. metal or food) is passed through the metal detector. Based on the speed of the conveyor belt, the total acquisition time per measurement is defined to be two seconds. The inputs to the digital part of the DAS is $u(t)$ and $d(t)$.

C.3 Analog Hardware

The authors of the report have cooperated with ASPI-10 Group 08gr1041, in the development of a power amplifier for supplying current to the transmitting coil. Small signal voltage amplifiers, and other analog hardware components, are credited the work of ASPI-10 Group 08gr1041. The documentation of the power amplifier, together the rest of the analog hardware, is found in their report, see the documentation of 08gr1041 [AAU 2008]. Because of this, the analog hardware is only described on block level in this report. The block diagram for hardware is presented in Fig. C.2.
In the following the blocks of the figure are described.

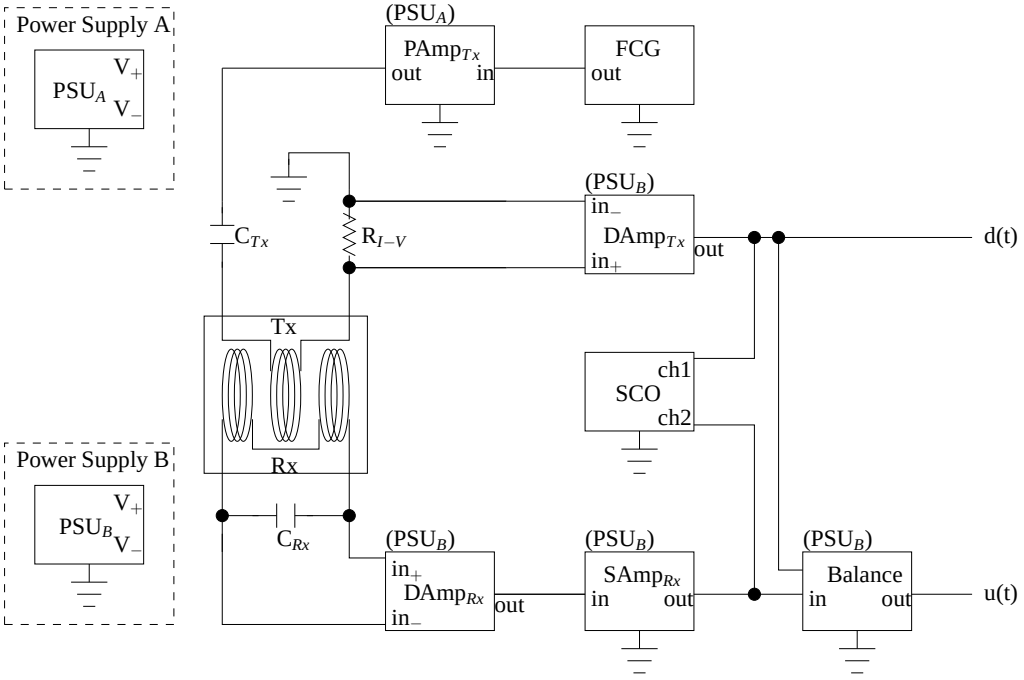

Figure C.2: Block diagram of analog hardware connected to the receiving coils (Rx) and the transmitting coil (Tx) of the metal detector head.

C.3.1 I-V Converter (R_{I-V})

The obtain the sinusoidal waveform in the transmitting coil, i.e. the current, a resistor, R_{I-V}, is placed in series with the transmitting coil and capacitor C_{Tx}. The resistor acts as a current-to-voltage converter. The I-V conversion has to be performed, to be able to connect and measure with the SCO. The sinusoidal waveform is the reference for obtaining the phase shift, made by the object placed in the detector head model.

C.3.2 Tx Power Amplifier ($PAmp_{Tx}$)

The push-pull amplifier stage (Class AB), consists of the NPN transistor BDX33B and PNP counterpart BDX34B, which are both Darlington transistors. The current gain is approximately 100 times at 1 MHz, when the collector current is 2 A, which should be sufficiently to drive the transmitting coil in detector head. Biasing of the stage is performed by a V_{BE}-multiplier, and is done to avoid crossover distortion in the output.
The series capacitor C_{Tx} is present due to minimize the impedance of load of the amplifier. Minimizing the load impedance, results in supplying lower voltage supplied from the power amplifier. The RLC circuit has the effect of attenuating noise in frequencies outside the transmitting frequency. For more details see App. D.2.

C.3.3 Differential Amplifiers ($DAmp_{Tx}$ and $DAmp_{Rx}$)

Both the refence signal $d(t)$ and the observation signal $u(t)$, are obtained from a differential amplifier. The reference signal is taken differentially, bacause of the voltage-to-current converter R_{I-V}.

The observation signal $u(t)$ is taken differentially across parallel between the receiving coils and a capacitor C_{Rx}.

C.3.4 Small Signal Amplifier ($SAmp_{Rx}$)

The small signal amplifier(s) is due to the amplification of differential signal, before the ADC.

C.3.5 Balance

The balance circuit consists of three allpass filters, followed by a differential amplifier, for analog balancing of the receiving coils.

The measurement setup is described in the following.

C.4 Measurement Setup

The output terminals of the analog hardware ($u(t)$ and $d(t)$), are connected to the input channels 2 and 3 of the PCI-DAS4020/12 sample board in the DAS.

Equipment	**Reference**	**Name**	**AAU-Nr**
Oscilloscope	SCO	Agilent 54621A	33850
Function Generator	FCG	Philips 08559	08559
Power Supply	PSU	Hamag HM7042	33888
Power Supply	PSU	Hamag HM7042	33891
PC	PC	Fujitsu-Siemens	-
Rx Differential Amplifier	DAmp_{Rx}	-	-
Rx Small Signal Amplifier	SAmp_{Rx}	-	-
Tx Differential Amplifier	DAmp_{Tx}	-	-
Tx Power Amplifier	PAmp_{Tx}	-	-
1 Ω Resistor (3 Watts)	RIV	-	-

Table C.1: Equipment and other components used for measuring the signals from the metal detector.

Appendix D

Model of Detector Head Coils

This appendix is related to Cha. 1, concerning the model of the three-coil detector head. The appendix covers the mathematics behind the derivation of the model, followed by an analysis of the effects on the transmitting coil, when an object is passed through it.

D.1 Mathematical Derivation of Detector Head Model

This appendix section covers the details about the mathematical derivation of the metal detector head model.

D.1.1 Vector Potential in General

The vector potential A is a way of defining the magnetic field $\mathbf{B}$ in some point in space. The reason for choosing A instead of $\mathbf{B}$is due to the definition,

$$\nabla \times A = \mathbf{B} \tag{D.1}$$

which shows that the vector potential A, at a given point in space, can be converted to $\mathbf{B}$ in that point. When introducing spherical coordinates, the vector potential has the advantage of making the derivations less complex, as mentioned before, if the orientation in the coordinate system is chosen wisely. It can be argued by the fact that the Eddy currents are assumed to flow in circles, whereas the vector potential will follow these. The vector potential is defined as [*kilde*],

$$A = \frac{\mu_0 I}{4\pi} \cdot \oint_{c'} \frac{dl}{R} \quad \left[\frac{Wb}{m}\right] \tag{D.2}$$

for a thin wire carrying a current I. The curve integral is evaluated over the current path c' and R is the distance from the path c' to the point in space where the vector potential is observed.
In Fig. D.1 the model of the system is shown. The loop is equal to a one turn coil carrying a current I. The circle in the centre of the coordinate system is a massive spherical object. Due to the position of the loop, it is chosen to use spherical coordinates. The vector potential A at point P in spherical coordinates is expressed as,

$$A = \frac{\mu_0 I}{4\pi} C \sin\alpha \cdot \oint_0^{2\pi} \frac{\cos(\phi - \phi')}{R} d\phi \tag{D.3}$$

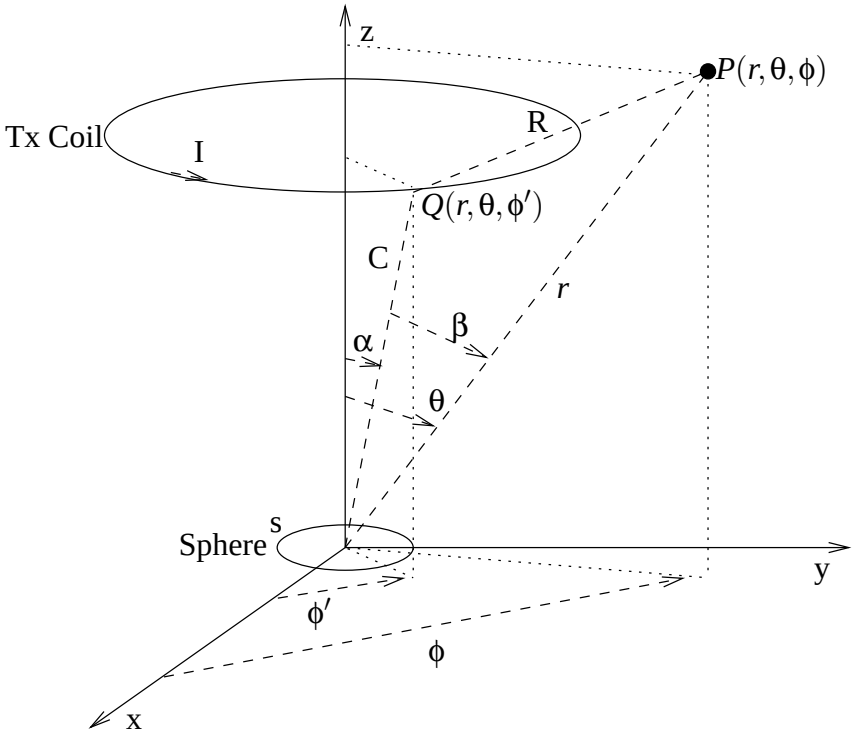

Figure D.1: Model of the system

where $C \sin\alpha \cdot \cos(\phi - \phi')$ is the path of integration. The vector potential A is only represented in the ϕ direction, due to the fact that the current flow in the current carrying loop only exists in ϕ, which is also seen from the path of integration. The distance R is found from the cosine relation as,

$$\cos(\beta) = \frac{r^2 + C^2 - R^2}{2rC} \tag{D.4}$$

$$\Leftrightarrow \frac{1}{R} = \frac{1}{\sqrt{r^2 + C^2 - 2rC\cos\beta}} \tag{D.5}$$

The expression $\frac{1}{R}$ can be rewritten to form the generating function for Legendre polynomials. The generating function for the Legendre polynomials is expressed as [Spiegel and Liu, 1999],

$$\sum_{n=0}^{\infty} P_n(x) \cdot t^n = \sqrt{\frac{1}{1 + t^2 - 2tx}} \tag{D.6}$$

The expression for $\frac{1}{R}$ is rewritten,

$$\frac{1}{R} = \sqrt{\frac{1}{r^2 + C^2 - 2rC\cos\beta}} \tag{D.7}$$

$$= \sqrt{\frac{\frac{1}{C^2}}{1 + \frac{r^2}{C^2} - 2\frac{r}{C}\cos\beta}} \tag{D.8}$$

$$= \frac{1}{C}\sqrt{\frac{1}{1 + \left(\frac{r}{C}\right)^2 - 2\frac{r}{C}\cos\beta}} \tag{D.9}$$

from which it is seen that,

$$\frac{1}{R} = \frac{1}{C}\sum_{n=0}^{\infty} P_n(\cos\beta)\left(\frac{r}{C}\right)^n \tag{D.10}$$

$$= \sum_{n=0}^{\infty} \frac{r^n}{C^{n+1}} P_n(\cos\beta) \tag{D.11}$$

It should be noted that the points P and Q cannot be equal, as this implies $\frac{1}{R} \to \infty$. The term $\cos\beta$ is defined as,

$$\cos\beta = \cos\theta\cos\alpha + \sin\theta\sin\alpha\cos(\phi - \phi') \tag{D.12}$$

which is defined by use of the addition formula. However the extra term $\cos\phi - \phi'$ is needed for projection of the plane with C as diagonal onto the plane with r as diagonal to obtain the correct orientation. When the cosine addition theorem is applied to the Legendre polynomium in Eq. D.11, it is expressed as,

$$P_n(\cos\beta) \tag{D.13}$$

$$= P_n(\cos\theta)P_n(\cos\alpha) + 2\sum_{m=1}^{n} \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta)P_n^m(\cos\alpha)\cos(m(\phi - \phi')) \tag{D.14}$$

$$= k\sum_{m=0}^{n} \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta)P_n^m(\cos\alpha)\cos(m(\phi - \phi')) \tag{D.15}$$

due to the Spherical Harmonic Addition theorem [MathWorld, 2007]. The constant k equals 1 for $m = 0$ and 2 for $m > 0$.

To find the vector potential A at a point P, Eq. D.11 and Eq. D.15 should be substituted into Eq. D.3. First Eq. D.15 is substituted into Eq. D.11 and this is substituted into Eq. D.3. The integration in Eq. D.3 is moved inside the summation in Eq. D.15 and only the cosines are kept to illustrate the point,

$$\sum_{m=0}^{n} \int_0^{2\pi} \cos(\phi - \phi')\cos(m(\phi - \phi'))d\phi \tag{D.16}$$

Because of the sum of integrated products over an integer multiplum of entire periods (0 to 2π), the sum of integrations will equal zero when $m \neq 1$ due the orthogonality of sinusoids. The integration is solved as,

$$\int_0^{2\pi} \cos(\phi - \phi')\cos(1(\phi - \phi'))d\phi = \int_0^{2\pi} \frac{1}{2}(1 - \cos(2(\phi - \phi')))d\phi \tag{D.17}$$

$$= \left[\frac{1}{2}(\phi - \frac{\sin(2(\phi - \phi'))}{2})\right]_0^{2\pi} \tag{D.18}$$

$$= \pi \tag{D.19}$$

Before substituting Eq. D.15 into Eq. D.11 it is reduced to,

$$k\sum_{m=0}^{n} \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta)P_n^m(\cos\alpha)\cos(m(\phi - \phi')) \tag{D.20}$$

$$= 2\frac{(n-1)!}{(n+1)!} P_n^1(\cos\theta)P_n^1(\cos\alpha)\pi \tag{D.21}$$

$$= \frac{2}{n(n+1)} P_n^1(\cos\theta)P_n^1(\cos\alpha)\pi \tag{D.22}$$

by use of the integration results in Eq. D.19. When substitution of Eq. D.11 into Eq. D.3 is performed, A is expressed as,

$$A = \frac{\mu_0 I}{4\pi} C \sin\alpha \cdot \sum_{n=1}^{\infty} \frac{r^n}{C^{n+1}} \frac{2}{n(n+1)} P_n^1(\cos\theta) P_n^1(\cos\alpha)\pi \tag{D.23}$$

$$= \sum_{n=1}^{\infty} \frac{\mu_0 I \sin\alpha}{2n(n+1)} C^{-n} P_n^1(\cos\alpha) P_n^1(\cos\theta) r^n \tag{D.24}$$

$$= \sum_{n=1}^{\infty} A_n P_n^1(\cos\theta) r^n \tag{D.25}$$

$$\text{where } A_n = \frac{\mu_0 I \sin\alpha}{2n(n+1)} C^{-n} P_n^1(\cos\alpha) \tag{D.26}$$

Eq. D.25 expresses the vector potential at any point P due to a current carrying loop. In the next section the general form of A_n is derived. The general form is used to describe the vector potential inside the sphere in Sec. D.1.3 and outside in Sec. D.1.4.

D.1.2 Vector Potential in Differential Form

Maxwell's first equation, see Eq. 1.1, states the relation between the electric field $\mathbf{E}$ and the magnetic field $\mathbf{B}$. The magnetic field can be expressed by the magnetic vector potential A, which gives,

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} = -\frac{\partial \nabla \times A}{\partial t} = -\nabla \times \frac{\partial A}{\partial t} \tag{D.27}$$

This is rewritten as follows,

$$\nabla \times \mathbf{E} + \nabla \times \frac{\partial A}{\partial t} = 0 \tag{D.28}$$

$$\Leftrightarrow \quad \nabla \times \left\{ \mathbf{E} + \frac{\partial A}{\partial t} \right\} = 0 \tag{D.29}$$

Now for Eq. D.27 to be fulfilled, the inner expression of Eq. D.29 has to equal zero and thereby,

$$\mathbf{E} = -\frac{\partial A}{\partial t} \tag{D.30}$$

Eq. D.30 states that the electric field equals the rate of change of the vector potential, which is an important point in the following calculations.
Maxwell's second equation, see Eq. 1.2, describes the current density by means of the magnetic field strength $\mathbf{H}$. This fact can be used to describe the secondary field generated in the object.
Eq. 1.2 is rewritten as follows

$$\nabla \times \mathbf{H} = \nabla \times \frac{1}{\mu}\mathbf{B} = \nabla \times \frac{1}{\mu} \nabla \times A \tag{D.31}$$

where $\mu = \mu_0 \mu_r$. The current density in the object, because of the presence of a magnetic field, can now be expressed as,

$$\nabla \times \nabla \times A = \mu \nabla \times \mathbf{H} \tag{D.32}$$

$$= \mu \left[\mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \right] \tag{D.33}$$

$$= \mu \left[\sigma \mathbf{E} + \frac{\partial \epsilon \mathbf{E}}{\partial t} \right] \tag{D.34}$$

Now the expression for **E** can be inserted, which yields,

$$\nabla \times \nabla \times A = \mu \left[-\sigma \frac{\partial A}{\partial t} + \frac{\partial}{\partial t} \left\{ -\epsilon \frac{\partial A}{\partial t} \right\} \right] \tag{D.35}$$

Now assuming all fields to be time-harmonic in the system, the derivative $\frac{\partial}{\partial t}$ can be replaced the complex operator $j\omega$, giving,

$$\begin{aligned} \nabla \times \nabla \times A &= \mu \left[-j\omega\sigma A + \omega^2 \epsilon A \right] && \text{(D.36)} \\ &= -j\omega\mu(\sigma + j\omega\epsilon)A && \text{(D.37)} \end{aligned}$$

Now from Eq. D.37 the current density in the object is defined as,

$$\nabla \times \nabla \times A = -jPA \tag{D.38}$$

with $P = \omega\mu(\sigma + j\omega\epsilon)$. Because $\omega\epsilon << \sigma$ at low frequencies, it is assumed that the term $j\omega\epsilon$ can be removed, which leads to $P = \omega\mu\sigma$.
In the following, Eq. D.37 is written as differentials to find one general solution. Since the vector potential is only pointing in the ϕ-direction, the contribution from the curl in the other directions is zero. Thus,

$$[\nabla \times \nabla \times A_n]_\phi = \frac{1}{r} \left[\frac{\partial}{\partial r} \left\{ -\frac{\partial}{\partial r} A_n r \right\} \right] - \frac{1}{r} \left[\frac{\partial}{\partial \theta} \left\{ \frac{1}{r \sin\theta} \left(\frac{\partial}{\partial \theta} A_n \sin\theta \right) \right\} \right] \tag{D.39}$$

The two parts concerning r and θ is treated each on its own. First the part concerning r is treated,

$$\begin{aligned} \frac{1}{r} \left[\frac{\partial}{\partial r} \left\{ -\frac{\partial}{\partial r} A_n r \right\} \right] &= -\frac{1}{r} \left[\frac{\partial}{\partial r} \left\{ A_n \frac{\partial r}{\partial r} + r \frac{\partial A_n}{\partial r} \right\} \right] && \text{(D.40)} \\ &= -\frac{1}{r} \left[A_n \frac{\partial^2 r}{\partial r^2} + 2 \frac{\partial r}{\partial r} \frac{\partial A_n}{\partial r} + r \frac{\partial^2 A_n}{\partial r^2} \right] && \text{(D.41)} \\ &= -\frac{1}{r^2} \left[r A_n \frac{\partial^2 r}{\partial r^2} + 2r \frac{\partial r}{\partial r} \frac{\partial A_n}{\partial r} + r^2 \frac{\partial^2 A_n}{\partial r^2} \right] && \text{(D.42)} \end{aligned}$$

Now the right most term can be written as $r^2 \frac{\partial^2 A_n}{\partial r^2} = \frac{\partial}{\partial r} \left\{ r^2 \frac{\partial^2 A_n}{\partial r^2} \right\} + expr$, from which the unknown term $expr$ should be found,

$$\begin{aligned} expr &= r^2 \frac{\partial^2 A_n}{\partial r^2} - \frac{\partial}{\partial r} \left\{ r^2 \frac{\partial A_n}{\partial r} \right\} && \text{(D.43)} \\ &= r^2 \frac{\partial^2 A_n}{\partial r^2} - \left(2r \frac{\partial A_n}{\partial r} + r^2 \frac{\partial^2 A_n}{\partial r^2} \right) && \text{(D.44)} \\ &= -2r \frac{\partial A_n}{\partial r} && \text{(D.45)} \end{aligned}$$

which inserted into Eq. D.42 yields,

$$\begin{aligned} & -\frac{1}{r^2} \left[r A_n \frac{\partial^2 r}{\partial r^2} + 2r \frac{\partial r}{\partial r} \frac{\partial A_n}{\partial r} + r^2 \frac{\partial^2 A_n}{\partial r^2} \right] && \text{(D.46)} \\ = & -\frac{1}{r^2} \left[2r \frac{\partial A_n}{\partial r} + \frac{\partial}{\partial r} \left\{ r^2 \frac{\partial A_n}{\partial r} \right\} - 2r \frac{\partial A_n}{\partial r} \right] && \text{(D.47)} \\ = & -\frac{1}{r^2} \left[\frac{\partial}{\partial r} \left\{ r^2 \frac{\partial A_n}{\partial r} \right\} \right] && \text{(D.48)} \end{aligned}$$

because the second order derivative of r equals zero. Next the term related to θ in Eq. D.39 is treated,

$$-\frac{1}{r}\frac{\partial}{\partial\theta}\left\{\frac{1}{r\sin\theta}\left[\frac{\partial}{\partial\theta}A_n\sin\theta\right]\right\} \tag{D.49}$$

$$= -\frac{1}{r}\frac{\partial}{\partial\theta}\left\{\frac{1}{r\sqrt{1-\cos^2\theta}}\left[\frac{\partial}{\partial\theta}\left\{A_n\sqrt{1-\cos^2\theta}\right\}\right]\right\} \tag{D.50}$$

because $\sin\theta = \sqrt{1-\cos^2\theta}$. Next substitution of variables is performed with $u = \cos\theta$ and the differential is changed to $\partial\theta = \frac{\partial u}{-\sin\theta}$, which yields,

$$-\frac{1}{r}(-\sin\theta)\frac{\partial}{\partial u}\left\{\frac{1}{r\sqrt{1-u^2}}\left[(-\sin\theta)\frac{\partial}{\partial u}\left\{A_n\sqrt{1-u^2}\right\}\right]\right\} \tag{D.51}$$

$$= \frac{\sin\theta}{r}\frac{\partial}{\partial u}\left\{-\frac{\sin\theta}{r\sqrt{1-u^2}}\left[\sqrt{1-u^2}\frac{\partial A_n}{\partial u}+A_n\frac{\partial\sqrt{1-u^2}}{\partial u}\right]\right\} \tag{D.52}$$

$$= -\frac{\sin\theta}{r^2}\frac{\partial}{\partial u}\left\{\sqrt{1-u^2}\frac{\partial A_n}{\partial u}+A_n\frac{\partial\sqrt{1-u^2}}{\partial u}\right\} \tag{D.53}$$

because $\sin\theta = \sqrt{1-u^2}$ and using the fact that r is a constant in connection with θ. Now the second derivative in Eq. D.53 can be performed, which yields,

$$-\frac{\sin\theta}{r^2}\left[\sqrt{1-u^2}\frac{\partial^2 A_n}{\partial u^2}+\frac{\partial A_n}{\partial u}\frac{\partial\sqrt{1-u^2}}{\partial u}+\left(\frac{\partial A_n}{\partial u}\frac{\partial\sqrt{1-u^2}}{\partial u}+A_n\frac{\partial^2\sqrt{1-u^2}}{\partial u^2}\right)\right] \tag{D.54}$$

$$= -\frac{\sin\theta}{r^2}\left[\sqrt{1-u^2}\frac{\partial^2 A_n}{\partial u^2}+2\frac{\partial A_n}{\partial u}\frac{\partial\sqrt{1-u^2}}{\partial u}+A_n\frac{\partial^2\sqrt{1-u^2}}{\partial u^2}\right] \tag{D.55}$$

$$= -\frac{\sin\theta}{r^2}\frac{\partial^2}{\partial u^2}\left\{A_n\sqrt{1-u^2}\right\} \tag{D.56}$$

The terms related to r and θ, Eq. D.42 and Eq. D.56 respectively, are now combined in Eq. D.39 giving Eq. D.58,

$$[\nabla\times\nabla\times A_n]_\phi \tag{D.57}$$

$$= \frac{1}{r^2}\frac{\partial}{\partial r}\left\{r^2\frac{\partial A_n}{\partial r}\right\}+\frac{\sqrt{1-u^2}}{r^2}\frac{\partial^2}{\partial u^2}\left\{\sqrt{1-u^2}A_n\right\} = jPA_n \tag{D.58}$$

The general solution given in Eq. D.58 is now used in the two following subsections, where the vector potential is determined inside and outside the sphere.

D.1.3 Vector Potential Inside Sphere

Eq. D.58 consists of two second order differential equations relating to r and u. To solve the equation, it is separated into two equations, where the first term relates to r and the second to u. The vector potential A_n in Eq. D.58 is in this section written as $A_{s,n}$, because it is the general solution to the vector potential inside the sphere. Because $A_{s,n}$ dependends on the two variables r and u, it is expressed as,

$$A_{s,n} = \frac{R_n\cdot U_n}{\sqrt{r}} \tag{D.59}$$

where R_n relates to r and U_n to u, both in A_s. When Eq. D.59 is substituted into Eq. D.58 the equation can be parted into the two terms relating to r and u respectively,

$$\frac{U_n}{r^2 \cdot \sqrt{r}} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) + \frac{R_n \sqrt{1-u^2}}{\sqrt{r} \cdot r^2} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} = \frac{jPR_nU_n}{\sqrt{r}} \tag{D.60}$$

$$\Leftrightarrow \frac{1}{R_n} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) + \frac{\sqrt{1-u^2}}{U_n} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} = jPr^2 \tag{D.61}$$

$$\Leftrightarrow \frac{1}{R_n} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) - jPr^2 + \frac{\sqrt{1-u^2}}{U_n} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} = 0 \tag{D.62}$$

Due to the separation of the variables in Eq. D.62, it can be written as,

$$f(r) + g(u) = 0 \tag{D.63}$$

which gives rise to,

$$f(r) = -g(u) = k \tag{D.64}$$

where k is the separation constant. This means that Eq. D.62 can be parted into to seperate equations written as,

$$f(r) = \frac{1}{R_n} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) - jPr^2 = k \tag{D.65}$$

$$g(u) = \frac{\sqrt{1-u^2}}{U_n} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} = -k \tag{D.66}$$

The separation constant k is chosen to $n(n+1)$ due to the guess of the solution to Eq. D.65 and Eq. D.66 [*It is not obvious why k should be n(n+1).*]. With $k = n(n+1)$, Eq. D.65 can be expressed as,

$$\frac{1}{R_n} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) - jPr^2 = n(n+1) \tag{D.67}$$

$$\Leftrightarrow \quad \frac{1}{R_n} \left(r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - \frac{1}{4} R_n \right) - (jPr^2 + n(n+1)) = 0 \tag{D.68}$$

$$\Leftrightarrow \quad r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - (jPr^2 + n(n+1) + \frac{1}{4}) R_n = 0 \tag{D.69}$$

$$\Leftrightarrow \quad r^2 \frac{\partial^2 R_n}{\partial r^2} + r \frac{\partial R_n}{\partial r} - ((\sqrt{jP} r)^2 + (n + \frac{1}{2})^2) R_n = 0 \tag{D.70}$$

and Eq. D.66 as,

$$\frac{\sqrt{1-u^2}}{U_n} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} = -n(n+1) \tag{D.71}$$

$$\Leftrightarrow \quad \sqrt{1-u^2} \frac{\partial^2}{\partial u^2} \left\{ \sqrt{1-u^2} U_n \right\} + n(n+1) U_n = 0 \tag{D.72}$$

$$\Leftrightarrow \quad (1-u^2)\frac{\partial^2 U_n}{\partial u^2} - 2u\frac{\partial U_n}{\partial u} - \left(\frac{1}{1-u^2}\right)U_n + n(n+1)U_n = 0 \tag{D.73}$$

$$\Leftrightarrow \quad (1-u^2)\frac{\partial^2 U_n}{\partial u^2} - 2u\frac{\partial U_n}{\partial u} + \left(n(n+1) - \frac{1}{1-u^2}\right)U_n = 0 \tag{D.74}$$

Eq. D.70 and Eq. D.74 have the form of a modified Bessel function and an associated Legendre function respectively, which give the general solutions,

$$R_n = B_n \cdot \sqrt{r} I_{n+\frac{1}{2}}(\sqrt{jP}r) + C_n \cdot \sqrt{r} K_{n+\frac{1}{2}}(\sqrt{jP}r) \tag{D.75}$$

$$U_n = D_n \cdot P_n^1(u) + E_n \cdot Q_n^1(u) \tag{D.76}$$

where B_n, C_n, D_n and E_n are constants. The higher order terms $K_{n+\frac{1}{2}}(r)$ and $Q_n^1(u)$ are removed, because they do not satisfy the physical model. For further explanation see [nake et al.] [*We should explain this instead.*]. The solution to the vector potential inside the sphere, $A_{s,n}$, can then be expressed as,

$$A_{s,n} = B_n D_n \cdot \sqrt{r} I_{n+\frac{1}{2}}(\sqrt{jP}r) \cdot P_n^1(u) \tag{D.77}$$

$$= F_n \cdot \sqrt{r} I_{n+\frac{1}{2}}(\sqrt{jP}r) \cdot P_n^1(u) \tag{D.78}$$

The vector potential outside the sphere is found in the following subsection.

D.1.4 Vector Potential Outside Sphere

The vector potential outside the spherical conductor, $A_{o,n}$, can be expressed by use of the general solution in Eq. D.58. Due to no current flowing outside the conductor ($\sigma = 0$), the equation is expressed as,

$$[\nabla \times \nabla \times A_{o,n}] = \frac{1}{r^2}\frac{\partial}{\partial r}\left\{r^2\frac{\partial A_{o,n}}{\partial r}\right\} + \frac{\sqrt{1-u^2}}{r^2}\frac{\partial^2}{\partial u^2}\left\{\sqrt{1-u^2}A_{o,n}\right\} = 0 \tag{D.79}$$

Separation of variables is performed again, where $A_{o,n}$ is expressed as,

$$A_{o,n} = R_n \cdot U_n \tag{D.80}$$

The separation constant is again chosen to $k = n(n+1)$. The separated equations for r and u are then expressed as,

$$\frac{1}{R_n}\frac{\partial}{\partial r}\left\{r^2\frac{\partial R_n}{\partial r}\right\} - n(n+1) = 0 \tag{D.81}$$

$$\frac{\partial}{\partial r}\left\{r^2\frac{\partial R_n}{\partial r}\right\} - n(n+1)R_n = 0 \tag{D.82}$$

$$r^2\frac{\partial^2 R_n}{\partial r^2} + 2r\frac{\partial R_n}{\partial r} - n(n+1)R_n = 0 \tag{D.83}$$

and

$$\frac{\sqrt{1-u^2}}{U_n}\frac{\partial^2}{\partial u^2}\left\{\sqrt{1-u^2}U_n\right\} = -n(n+1) \tag{D.84}$$

$$\sqrt{1-u^2}\frac{\partial^2}{\partial u^2}\left\{\sqrt{1-u^2}U_n\right\} + n(n+1)U_n = 0 \tag{D.85}$$

$$(1-u^2)\frac{\partial^2 U_n}{\partial u^2} - 2u\frac{\partial U_n}{\partial u} + \left(n(n+1) - \frac{1}{1-u^2}\right)U_n = 0 \tag{D.86}$$

The solution to R_n in Eq. D.83 is expressed as,

$$R_n = G_n r^n + H_n r^{-n-1} \tag{D.87}$$

where G_n and H_n are constant. The solution to U_n is the same as the solution to Eq. D.74 due to the identical structure of the differential equation. The solution is given as,

$$\begin{aligned} U_n &= L_n \cdot P_n^1(u) + M_n \cdot Q_n^1(u) && \text{(D.88)} \\ U_n &= L_n \cdot P_n^1(u) && \text{(D.89)} \end{aligned}$$

The solution to the vector potential outside the sphere, $A_{o,n}$, is then written as,

$$\begin{aligned} A_{o,n} &= G_n L_n P_n^1(u) r^n + H_n L_n P_n^1(u) r^{-n-1} && \text{(D.90)} \\ A_{o,n} &= S_n P_n^1(u) r^n + T_n P_n^1(u) r^{-n-1} && \text{(D.91)} \end{aligned}$$

The next subsection concerns finding a specific solution to the differential equations, which corresponds to the change in total vector potential, when the spherical conductor is present in the detector head.

D.1.5 Vector Potential Due to Sphere

The first term in $A_{o,n}$ in Eq. D.91 is seen to be the vector potential due to the current carrying loop, which is given in Eq. D.26. Thereby,

$$S_n P_n^1(u) r^n = A_n P_n^1(\cos\theta) r^n \tag{D.92}$$

The second term in Eq. D.91 can then be seen to be produced due to the presence of the sphere. If Eq. D.91 is rewritten as,

$$A_{o,n} = S_n P_n^1(u)(r^n + \frac{T_n}{S_n} r^{-n-1}) \tag{D.93}$$

it can be seen that the change in $A_{o,n}$ due to the spherical conductor is,

$$\Delta A_{o,n} = S_n P_n^1(u) \frac{T_n}{S_n} r^{-n-1} \tag{D.94}$$

To solve Eq. D.94, the constants S_n and T_n have to be known. S_n is known because it equals A_n due to the fact that S_n is the constant introduced by the current carrying loop. The only unknown in Eq. D.94 is T_n. Instead of finding T_n the ratio $\frac{T_n}{S_n}$ is found, because there are more unknowns that boundary conditions. To find the ratio, two specific solutions are setup, both concerning the boundary region between sphere and surroundings, where r equals the radius of the sphere a. The first condition is, that the total vector potentials A_s and A_o in the boundary region are equal,

$$\begin{aligned} A_s &= A_o && \text{(D.95)} \\ &\Downarrow && \text{(D.96)} \\ A_{s,n} &= A_{o,n} && \text{(D.97)} \end{aligned}$$

The second condition is that the field strengths H_s and H_o in the θ direction, which encapsulates the sphere, are equal in the boundary region,

$$\begin{aligned} H_s &= H_o && \text{(D.98)} \\ \frac{1}{\mu_r\mu_0}\nabla \times A_s &= \frac{1}{\mu_0}\nabla \times A_o && \text{(D.99)} \\ -\frac{1}{\mu_r\mu_0}\cdot\frac{\sin(\theta)}{a\sin(\theta)}\frac{\partial(aA_s)}{\partial a} &= -\frac{1}{\mu_0}\cdot\frac{\sin(\theta)}{a\sin(\theta)}\frac{\partial(aA_o)}{\partial a} && \text{(D.100)} \\ \mu_0\frac{\partial(aA_s)}{\partial a} &= \mu_0\mu_r\frac{\partial(aA_o)}{\partial a} && \text{(D.101)} \\ \Rightarrow \mu_0\frac{\partial(aA_{s,n})}{\partial a} &= \mu_0\mu_r\frac{\partial(aA_{o,n})}{\partial a} && \text{(D.102)} \end{aligned}$$

The ratio $\frac{T_n}{S_n}$ is found by substituting Eq. D.78 and Eq. D.91 into Eq. D.97 and Eq. D.102. The ratio is expressed as,

$$\frac{T_n}{S_n} = a^{2n+1}\left(1 - \frac{(2n+1)\cdot\mu_r\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}{\sqrt{jP}\cdot a\cdot I_{n-\frac{1}{2}}(\sqrt{jP}a) + n(\mu_r-1)\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}\right) = V_n \qquad \text{(D.103)}$$

The change in vector potential at a point P in space due to the existence of a spherical object is obtained by substituting Eq. D.26 and Eq. D.102 into Eq. D.94, which gives,

$$\begin{aligned} \Delta A_o &= A_n P_n^1(\cos\theta)V_n r^{-n-1} && \text{(D.104)} \\ &= \sum_{n=1}^{\infty}\frac{\mu_0 I\sin\alpha C^{-n}}{2n(n+1)}P_n^1(\cos\alpha)P_n^1(\cos\theta) && \text{(D.105)} \\ \cdot \quad & a^{2n+1}\left(1 - \frac{(2n+1)\cdot\mu_r\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}{\sqrt{jP}\cdot a\cdot I_{n-\frac{1}{2}}(\sqrt{jP}a) + n(\mu_r-1)\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}\right)r^{-n-1} && \text{(D.106)} \end{aligned}$$

This equation ends the derivation in the appendix, and the rest is found in part 1 of the report in Cha. 1.

In the following appendix section, the effects of moving an object through the three-coil detector head is analysed.

D.2 Change in Transmitting Coil Due to Sphere

The inductance of the transmitting coil can be separated in two inductances. A self inductance and a mutual inductance. The self inductance describes the inductance of the transmitting coil without other current loops present. The mutual inductance is the inductance due to other field producing loops or objects present in the system. In Fig. D.2 a model of the setup is shown.
The voltage drop across the coil due to the total inductance is given as,

$$\begin{aligned} \varepsilon_t &= -(L_s\frac{\partial I}{\partial t} + L_m\frac{\partial I}{\partial t}) && \text{(D.107)} \\ &= -(L_s + L_m)\frac{\partial I}{\partial t} && \text{(D.108)} \\ &= -L\frac{\partial I}{\partial t} && \text{(D.109)} \end{aligned}$$

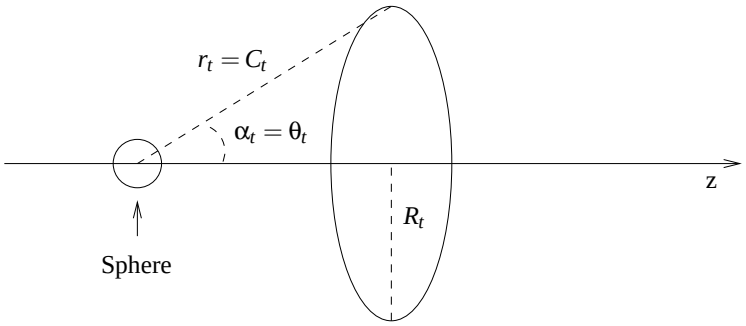

Figure D.2: Model of sphere moving through the transmitting coil.

where ε_t is the electromotive force applied to the transmitting coil, L_s is the self inductance of the coil, and L_m is the mutual induction between the coil and the sphere. From Faraday's law the voltage drop can be expressed as,

$$\begin{aligned}
\varepsilon_t &= -\frac{\partial}{\partial t}\oint_C A d\mathbf{l} - \frac{\partial}{\partial t}\oint_C \Delta A_o d\mathbf{l} && \text{(D.110)}\\
&= -\oint_C A' d\mathbf{l}\frac{\partial I}{\partial t} - \oint_C \Delta A'_o d\mathbf{l}\frac{\partial I}{\partial t} && \text{(D.111)}\\
&= -(\oint_C A' d\mathbf{l} + \oint_C \Delta A'_o d\mathbf{l})\frac{\partial I}{\partial t} && \text{(D.112)}\\
&= -(L_s + L_m)\frac{\partial I}{\partial t} && \text{(D.113)}
\end{aligned}$$

which is used to derive the formula for the change in inductance in the transmitting coil. Eq. D.113 shows that the inductance in the transmitting coil, consists of a summation of the self inductance and mutual inductance, which is expressed through curve integrals. When Eq. D.113 is compared with Eq. D.108, it is seen that the second curve integral is the mutual inductance between the sphere and the coil. This means, that the change in inductance can be expressed as,

$$\begin{aligned}
L &= L_s + L_m && \text{(D.114)}\\
&= L_s + \oint_C \Delta A'_0 d\mathbf{l} && \text{(D.115)}
\end{aligned}$$

The path of integration is the circumference of the transmitting coil, due to the transformation from a surface integral to a curve integral with use of Stoke's theorem. The term $\Delta A'_o$ is the change in vector potential due to the sphere described in Eq. 1.16. The $'$ notation means, that the derivative of the current term is removed from the vector potential, as it is moved outside the expression, as in Eq. D.111. When substituting Eq. 1.16 into Eq. D.115 the inductance of the transmitting coil is expressed as,

$$\begin{aligned}
L &= L_s + \oint_C \Delta A'_0 d\mathbf{l} && \text{(D.116)}\\
&= L_s + 2\cdot\pi\cdot R_t\cdot\Delta A'_0 && \text{(D.117)}\\
&= L_s + 2\cdot\pi\cdot R_t\cdot N_t(\sum_{n=1}^{\infty}\frac{\mu_0 N_t\sin\alpha_t}{2n(n+1)}C_t^{-n}P_n^1(\cos\alpha_t)P_n^1(\cos\theta_t\\
&\quad\cdot\ a^{2n+1}\left(1-\frac{(2n+1)\cdot\mu_r\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}{\sqrt{jP}\cdot a\cdot I_{n-\frac{1}{2}}(\sqrt{jP}a)+n(\mu_r-1)\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}\right)r_t^{-n-1}) && \text{(D.118)}
\end{aligned}$$

The path of integration is along the coil, which implies that $C_t = r_t$ and $\alpha_t = \theta_t$. This shortens Eq. D.118 to,

$$\begin{aligned} L &= L_s + \pi \cdot R_t \cdot N_t^2 \mu_0 \sin\alpha_t \left(\sum_{n=1}^{\infty} \frac{r_t^{-2n-1}}{n(n+1)} P_n^1(\cos\alpha_t)^2 \right. \\ &\cdot \left. a^{2n+1} \left(1 - \frac{(2n+1)\cdot\mu_r \cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}{\sqrt{jP}\cdot a \cdot I_{n-\frac{1}{2}}(\sqrt{jP}a) + n(\mu_r - 1)\cdot I_{n+\frac{1}{2}}(\sqrt{jP}a)}\right)\right) \end{aligned} \tag{D.119}$$

Eq. D.119 describes the inductance, when a spherical object is moved through the center of the transmitting coil. The term r_t is the distance between the center of the object to the circumference of the coil, α_t is the angle between the z-axis and r_t, and R_t is the radius of the transmitting coil.

In Fig. D.3 the change in inductance is illustrated, when moving an iron object through the detector head. The radius of the transmitting coil is 42 mm, and the distance is varied from -50 to 50 mm. The self inductance is $L_s = 17.6 \cdot^{-6}$ H, and size of the object is varied from 10 % to 90 % of the detector head opening.

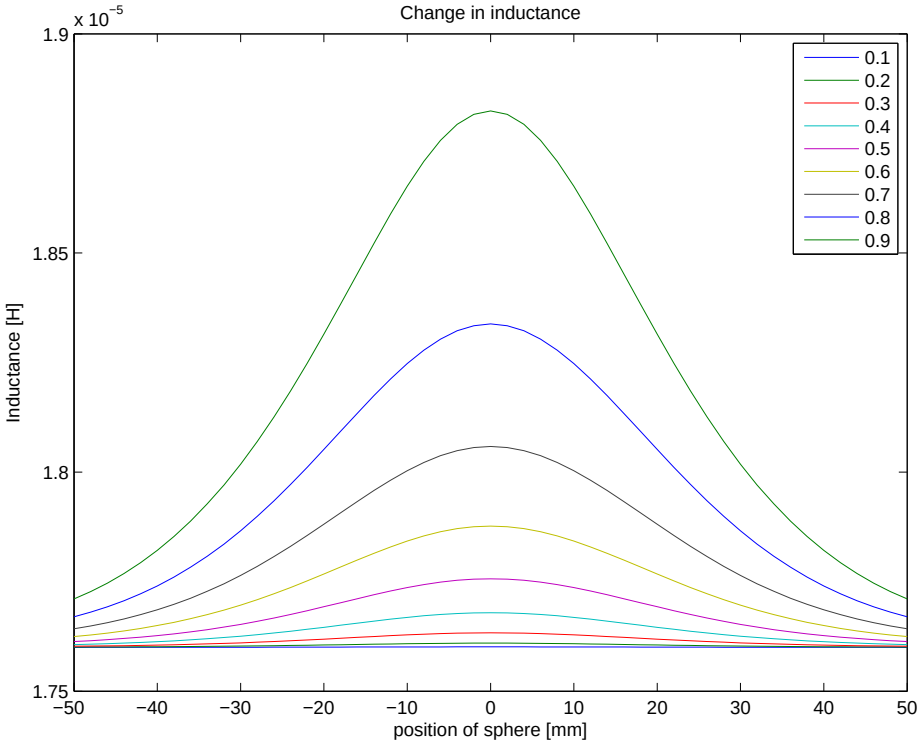

Figure D.3: Change in inductance due to a sphere moving through the transmitting coil. The transmitting coil is placed in zero. Each line represent a ratio between R_t and the radius of the sphere.

From Fig. D.3 the change in total inductance is seen to vary approximately 6.5 %, when the object is 90 % of the opening. When the object is smaller the variation also becomes smaller, which signifies that the importance of change in inductance is present for the magnetic food articles and for very large foreign metal objects to be detected.

Tx Coil in Series RLC

As described preceding to this, highly magnetic food articles and large foreign metal objects will change the inductance of the transmitting coil. However, to force a large current through the transmitting coil a series RLC circuit should be utilised. I such a circuit a variation in inductance will give a variation in resonance frequency, $\Delta\omega$. The impedance in the resonance frequency of a series RLC circuit is given by the value of the resistor, because the impedance of the capacitor and the coil cancel out. Across the series RLC a sinusoidal voltage of frequency corresponding to the RLC resonance frequency is applied, which consequently result in a current determined by the resistor through the circuit. The resistor is the component, that limits the current, such that a short circuit of infinite current is impossible. The resistor can be the internal ohmic resistance of the coil or an added component, which in both cases determines the 3 dB (half power) bandwidth around the resonance frequency - a larger resistor gives a larger bandwidth.
As a consequence of the change in inductance the resonance frequency, ω_0, will also change because of the following relation [Serway and John W. Jewett, 2004],

$$\omega_0 = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2} \tag{D.120}$$

$$= \frac{1}{\sqrt{LC}} \text{ for } R << \frac{4L}{C} \tag{D.121}$$

where L, C and R are the inductance, capacitance and resistance of the series RLC circuit respectively. As the applied voltage across the RLC circuit has a well defined frequency corresponding to ω_0, the total impedance increases with changing inductance and thereby a smaller current is flowing through the transmitting coil. The rate of change in impedance is determined by the size of the magnetic object affecting the transmitting coil, but can also be seen to depend on the half power bandwidth. If the RLC circuit has a narrow half power bandwidth, a small change in inductance will give rise to a large change in impedance and vice versa for a wide bandwidth. The half power bandwidth, BW_ω, is given as [Serway and John W. Jewett, 2004],

$$BW_\omega = \frac{R}{L} \tag{D.122}$$

The bandwidth BW_ω is interesting to observe together with the change in resonance frequency $\Delta\omega_0$. If $\Delta\omega_0$ is sufficiently large to exceed BW_ω, the increase in impedance will consequently decrease the current. An expression for $\Delta\omega_0$ can be obtained through the following,

$$\omega_0 + \Delta\omega_0 = \frac{1}{\sqrt{(L+\Delta L)C}} \tag{D.123}$$

$$\Delta\omega_0 = \frac{1}{\sqrt{(L+\Delta L)C}} - \omega_0 \tag{D.124}$$

where ΔL is the change in inductance causing the change in resonance frequency. By defining a new term expressing the ratio between BW_ω and $\Delta\omega_0$, the importance of change in inductance can be analyzed. Using Eq. D.122 and Eq. D.124 the ratio is obtained,

$$\frac{BW_\omega}{2\Delta\omega_0} = \frac{\frac{R}{L}}{2\left(\sqrt{\frac{1}{(L+\Delta L)C}} - \omega_0\right)} \tag{D.125}$$

$$= \frac{\frac{R}{2L}}{\frac{1}{\sqrt{(L+\Delta L)C}} - \frac{1}{LC}} \tag{D.126}$$

where ω_0 is expressed as $\frac{1}{\sqrt{LC}}$ and $\Delta\omega_0$ is multiplied by 2 to compensate for the bandwidth represented on both sides of ω_0. When $\frac{BW_\omega}{2\Delta\omega_0}$ becomes 1, the total power in the circuit is halved. Below 1 the power desipated becomes smaller and otherwise larger. To force most current through the circuit, and thereby through the transmitting coil, the inductance must be constant at the expected value - corresponding to $\frac{BW_\omega}{2\Delta\omega_0}$ being larger than 1 or approaching infinity. Using the values $C = 4.7$ nF and $R = 0.5\ \Omega$, the ratio $\frac{BW_\omega}{2\Delta\omega_0}$ is given with the change in inductance, ΔL, of ± 6.5 % in part a) of Fig. D.4, together with the change in frequency in part b).
In part a) of Fig. D.4 $\frac{BW_\omega}{2\Delta f_0}$ is given, which is the representation in frequency [Hz]. Here the

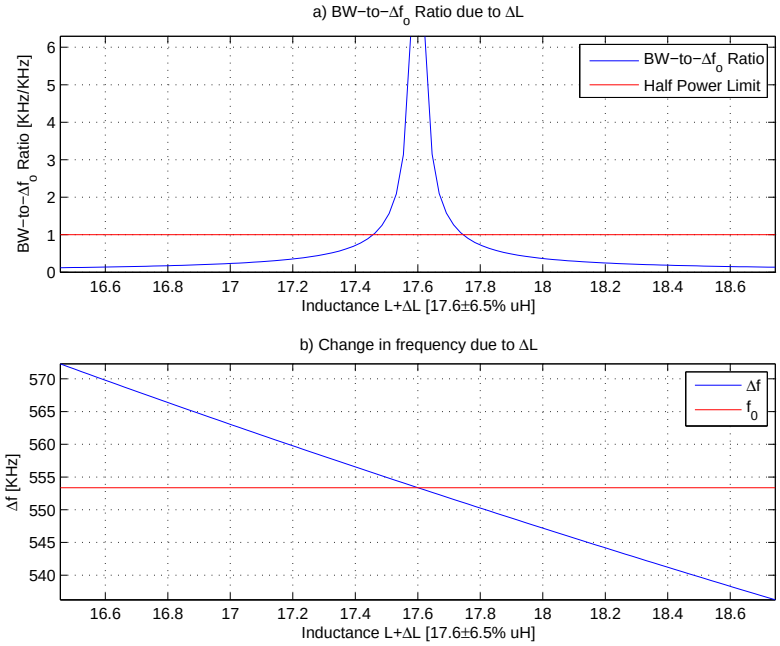

Figure D.4: In part a) an illustration of $\frac{BW_\omega}{2\Delta f_0}$ when the value of the inductance is changing because of e.g. magnetic food articles, using the size of 90 % of the detector head opening. In the center Δf_0 is zero. Increasing the inductance result in negative Δf_0 and decreasing inductance result in positive Δf_0. The sign cannot be seen in the plot. In part b) the change in resonance frequency is given. In both cases $C = 4.7$ nF and $R = 0.5\ \Omega$, and radian frequency [Rad/s] is changed to frequency [Hz].

magnetic object is placed in the center of transmitting coil and the size is 90 % of the detector head opening. At the correct inductance value of $L = 17.6 \cdot 10^{-6}$, corresponding to ΔL being zero, the ratio approaches infinity, because $\Delta\omega_0$ is zero. Introducing the change in inductance ΔL results in a change in ω_0 of the RLC circuit. When ΔL is large enough, the change in resonance frequency will exceed BW_ω of the circuit. In the limit, where $\Delta\omega_0$ is equal to BW_ω the power through the circuit is halved - this is illustrated as the Half Power Limit. This is true because the voltage applied to the circuit is fixed at ω_0. Exceeding this limit causes the power, and thereby the current through the transmitting coil, to decrease. The lower crossing is read as 17.456 μH and the upper crossing as 17.745 μH, corresponding to approximately 1 % change of inductance. As shown before, a total change in inductance of 6.5 % is obtained, when moving a large magnetic object through the transmitting coil. The result of this change reduces the current through the transmitting coil significantly, when the frequency of the applied voltage is constant at ω_0.

In part b) of Fig. D.4 the change in frequency, because of change in inductance, is given. When no magnetic object is in the transmitting coil, corresponding to $L = 17.6 \cdot 10^{-6}$, the resonance frequency f_0 is 553.37 KHz. When the inductance is changed by -6.5 % the frequency becomes 572 KHz and when the change is 6.5 % the new resonance frequency becomes 544 KHz, giving,

$$\Delta\omega_{0,-6.5\%} = 553.37 - 544 = 9.37 \text{ KHz} \tag{D.127}$$

$$\Delta\omega_{0,+6.5\%} = 572 - 553.37 = 18.63 \text{ KHz} \tag{D.128}$$

Relating the change in resonance frequency to the electromagnetic field theory, the magnetic field generated by the transmitting coil is reduced with the decrease in current. This corresponds to the magnetic field being weaker, which again weakens the effect of the foreign metal object to be detected. The observation field, $\mathbf{B}_x$, can be described as,

$$\mathbf{B}_x = \mathbf{B}_T + (\mathbf{B}_f + \mathbf{B}_s) \tag{D.129}$$

where $\mathbf{B}_T$ is the field from the transmitting coil, $\mathbf{B}_f$ is the field from the magnetic food article and $\mathbf{B}_s$ is the field from an eventual foreign metal object to be detected. The two last terms depend on $\mathbf{B}_T$, as this is the basis of a functioning metal detector. In the equation the field from the food article is seen as the noise term. This observation field equation is valid when no additive noise from outside the system[1] is present or can be neglected. The more or less ideal scenario is valid if enough current is forced through the transmitting coil, such that the extra noise term can be neglected. If, however, the transmitting field $\mathbf{B}_T$ is too weak, the observation equation is extended to,

$$\mathbf{B}_{x,N} = \mathbf{B}_T + (\mathbf{B}_f + \mathbf{B}_s) + \mathbf{B}_N \tag{D.130}$$

where $\mathbf{B}_N$ is the noise term from outside the system and it is by definition independent of the system. This can be related to a magnetic signal-to-noise ratio (MSNR) expressed for the observation equation,

$$\text{MSNR}_{obs} = \frac{\mathbf{B}_x(I_T)}{\mathbf{B}_N} \tag{D.131}$$

where I_T is the current in the transmitting coil, which determines the strength of $\mathbf{B}_T$, $\mathbf{B}_f$ and $\mathbf{B}_s$. If I_T is decreased because of the vicinity of magnetic food articles, the MSNR_{obs} is decreased rapidly, as seen from $\frac{BW_\omega}{2\Delta\omega_0}$.
As seen from Fig. D.4 a 1 % change in inductance can cause the power in the circuit to be halved, which corresponds to decreasing the current by a factor of $\sqrt{2}$. The consequence of this decrease of current is a decrease in $\mathbf{B}_T$, $\mathbf{B}_f$ and $\mathbf{B}_s$ by the same amount. From this discussion it is evident, that a small change in inductance result in a significant change in ω_0. The consequence is that the magnetic field from the transmitting coil becomes weaker.

The effects of moving a spherical object with magnetic properties through the transmitting coil is now analysed. It was found, that because of the RLC circuit, the resonance frequency changes due to changing inductance. The consequence of this is a decrease in current in the transmitting coil, which reduces the MSNR.

[1] system, i.e. detector head, food articles and foreign metal objects.

Appendix E

Signal Model and Analysis

This appendix is related to Cha. 2 in Part 1 of the report. The appendix consists of three sections, handlling The OOB analysis filter, an analysis of steady objects in the detector head, and an analysis of sampling of the amplitude/phase trajectory.

E.1 OOB Analysis Filter

The OOB compensation filter is made from an all-pass filter with constant gain A, and phase shift ϕ in ω_d is used. The difference equation of a first order allpass IIR section is given as,

$$\widehat{u}[n] = -\lambda \cdot u_{OOB}[n] + u_{OOB}[n-1] + \lambda \cdot \widehat{u}[n-1] \tag{E.1}$$

where $-1 \geq \lambda \leq 1$ is the all-pass filter coefficient. The computation of the filter coefficient λ is based on the phase estimate $\phi(\omega_d)$. The phase response of the first order section is the derivative of its frequency response, and can be shown to be [Härmä et al., 2000],

$$\phi(\omega) = \arctan \frac{(1-\lambda^2)\sin(\omega)}{(1+\lambda^2)\cos(\omega) - 2\lambda} \tag{E.2}$$

where $\omega = \frac{\omega_d}{f_s}$ is the normalised radian frequency and f_s being the sampling frequency. An expression for λ, and thereby a design rule, can be obtained from Eq. E.2 by isolating it. As it is seen from Eq. E.2 it represents a second order polynomium in λ, which gives two solutions. The second order polynomium is given as,

$$(\tan(\phi)\cos(\omega) + \sin(\omega))\lambda^2 - 2\tan(\phi)\lambda + \tan(\phi)\cos(\omega) - \sin(\omega) = 0 \tag{E.3}$$

The solutions to Eq. E.3 can be shown to be,

$$\lambda = \frac{\sin(\phi) \pm \sin(\omega)}{\sin(\omega + \phi)} \tag{E.4}$$

Using the constraint $|\lambda| \leq 1$ the following is obtained by Euler's identities,

$$\left|\frac{e^{j\phi} - e^{-j\phi}}{2j}\right| \pm \left|\frac{e^{j\omega} - e^{-j\omega}}{2j}\right| \leq \left|\frac{e^{j(\omega+\phi)} - e^{-j(\omega+\phi)}}{2j}\right| \tag{E.5}$$

where e is Euler's number and j is the complex number. Since the length of the exponential function is unity and using the fact that $\sin(-\phi) = -\sin(\phi)$, the two solutions for λ can be identified as,

$$\lambda_- = \frac{\sin(\phi) - \sin(\omega)}{\sin(\omega + \phi)} \quad \text{for } \phi \text{ positive} \tag{E.6}$$

$$\lambda_+ = \frac{\sin(\phi) + \sin(\omega)}{\sin(\omega + \phi)} \quad \text{for } \phi \text{ negative} \tag{E.7}$$

for the filter to be stable. The two solutions for λ represent a causal (λ_-) and a non-causal (λ_+) filter respectively. The fact that λ_- is seen as the causal solution is because a positive ϕ implies that that $u[n]$ has to be shifted negatively and thereby delayed in the detector head system. The non-causal part is equal to the causal part, but with ϕ shifted by $\pm\pi$ depending on the sign.

The first order section of the allpass filter is capable of performing a phase shift of nearly $\pm\frac{\pi}{2}$, because of the periodicity of the tan-function. The first order section can be cascaded to obtain higher orders and thereby larger phase shift. The tan-function in $\pm\frac{\pi}{2} \rightarrow \infty$, why two sections is not enough to obtain a total phase shift of $\pm\pi$. Thereby three first order sections are cascaded to satisfy the requirement. The coefficient of each of the three first order sections, $\widetilde{\lambda}$, is calculated as,

$$\widetilde{\lambda}_- = \frac{\sin(\frac{\phi}{3}) - \sin(\omega)}{\sin(\omega + \frac{\phi}{3})} \quad \text{for } \phi \text{ positive} \tag{E.8}$$

$$\widetilde{\lambda}_+ = \frac{\sin(\frac{\phi}{3}) + \sin(\omega)}{\sin(\omega + \frac{\phi}{3})} \quad \text{for } \phi \text{ negative} \tag{E.9}$$

Because each of the first order filters are stable ($|\lambda| \leq 1$) the resulting difference equation for the filter becomes,

$$\widehat{u}[n] = - \widetilde{\lambda}^3 u_{OOB}[n] + 3\widetilde{\lambda}^2 u_{OOB}[n-1] - 3\widetilde{\lambda} u_{OOB}[n-2] + u_{OOB}[n-3] \ ... \tag{E.10}$$

$$+ \ 3\widetilde{\lambda}\widehat{u}[n-1] - 3\widetilde{\lambda}^2\widehat{u}[n-2] + \widetilde{\lambda}^3\widehat{u}[n-3] \tag{E.11}$$

The third order all-pass filter is now capable of performing a phase shift of ϕ_{OOB}, but the amplitude ratio between $u_{OOB}[n]$ and $d[n]$ has to be incorporated in the filter as well. This is done by dividing the present input tap $u_{OOB}[n]$, with the amplitude estimate A_{OOB}.

In the following steady objects in the detector head is analysed.

E.2 Steady Objects in the Detector Head

In this appendix section steady objects in the detector head is analysed. The amplitude feature, caused by the object, is observed at the output terminal of the receiving coils. The phase feature is observed as the phase shift between the current in the transmitting coil and the voltage at the receiving coils.

E.2.1 Partitioning of Objects

The different objects will be parted into classes. The partitioning is performed to be able to describe classes of objects instead of specific objects with specific properties. The objects are grouped in

classes due to their permeability and conductivity, which are some of the object properties, which separates them physically. There will be defined four classes: conductive metal class, iron class, wet food class and dry food class.

The Conductive Metal Class

The class represents objects with high conductivity and relative permeability close to one. Objects placed in this class are called para-/diamagnetic materials. Objects of this class is aluminium, copper, silver, gold and brass. etc. The values for conductivity and permeability for this class used for simulation is:

- $\mu_r = [0.999\ 0.9999\ 0.99999\ 0.999999\ 1.0000001\ 1.000001\ 1.00001\ 1.0001]$
- $\sigma_r = [0.1\ 0.3\ 0.7\ 1\ 3\ 4\ 5\ 7] \cdot 10^7$

The Iron Class

The iron class represents objects which has ferromagnetic properties and high conductivity. Objects belonging to this class are different types of iron. The values for conductivity and permeability used for this class is:

- $\mu_r = [100\ 250\ 750\ 1000\ 2000\ 300\ 4000\ 5000]$
- $\sigma_r = [0.1\ 0.3\ 0.7\ 1\ 3\ 4\ 5\ 7] \cdot 10^7$

The Dry Food Class

The dry food class represents objects which has weak ferromagnetic properties, but low conductivity. Objects belonging to this class is ferromagnetic food articles. The values of permeability and conductivity used for simulation is:

- $\mu_r = [1.1\ 1.15\ 1.2\ 1.25\ 1.3\ 1.4\ 1.5\ 1.6]$
- $\sigma_r = [0.001\ 0.005\ 0.01\ 0.05\ 0.075\ 0.1\ 0.25\ 0.4]$

The Wet Food Class

The wet food class describes objects which neither have conductive or magnetic properties. The objects belonging to this class could be objects as animal-tissues and salt enriched products. The values used for the simulations are is:

- $\mu_r = [0.999\ 0.9999\ 0.99999\ 0.999999\ 1.0000001\ 1.000001\ 1.00001\ 1.0001]$
- $\sigma_r = [0.01\ 0.04\ 0.08\ 0.2\ 0.35\ 0.7\ 2\ 4]$

The conductivity is based on values for animal-fat, -muscle and salt resolutions as seawater, [Kraus and Fleisch, 1999].

It is assumed that the classes are able to describe objects belonging to the class in general. This means that simulations of the classes are able to describe the features of the class in general. In the next section the features are examined in steady state and discussed.

E.2.2 Amplitude Feature

The examination is based on simulations using the four classes of objects, to see how they affect the system. The conductive metal class and iron class are considered as the objects wanted to be detected. The mean amplitude of the model output is plotted as function of frequency for different sizes of objects. The figures are shortly described in the following paragraphs.

The conductive metal class Fig. E.1 shows the amplitude of the conductive metal class as a function of frequency for different object sizes.

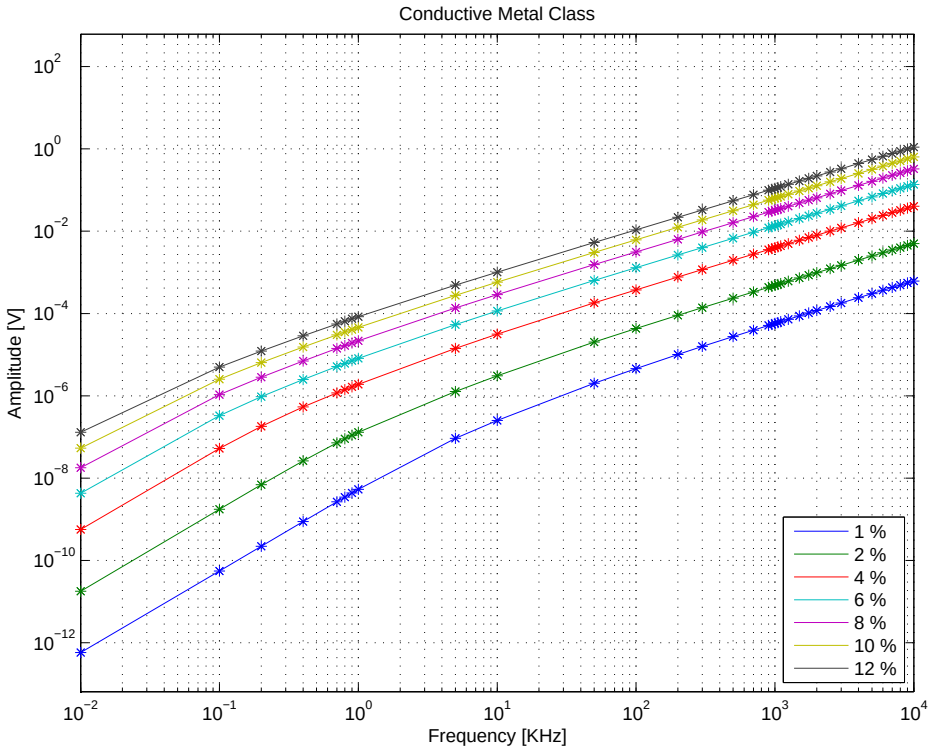

Figure E.1: Plot of the amplitude of the conductive metal class of materials, given as a function of frequency. Each curve represent the size of the object, given as percentage size of the detector head opening.

The figure shows that the amplitude of the signal from the receiving coils depends on the size of the object. When the size is increased the amplitude of the signal is increased. The amplitude also depends of frequency, because when the frequency is increased the amplitude increases.

The iron class Fig. E.2 shows the amplitude of the iron class as a function of frequency for different object sizes.
The figure shows, that amplitude increases, when the frequency increases. The amplitude decreases, when the size decreases, which means that the amplitude is size dependent. The amplitude also depends on frequency, because when the frequency is increased the amplitude increases.

The wet food class Fig. E.3 shows the amplitude as function frequency of different size objects belonging to the wet food class.

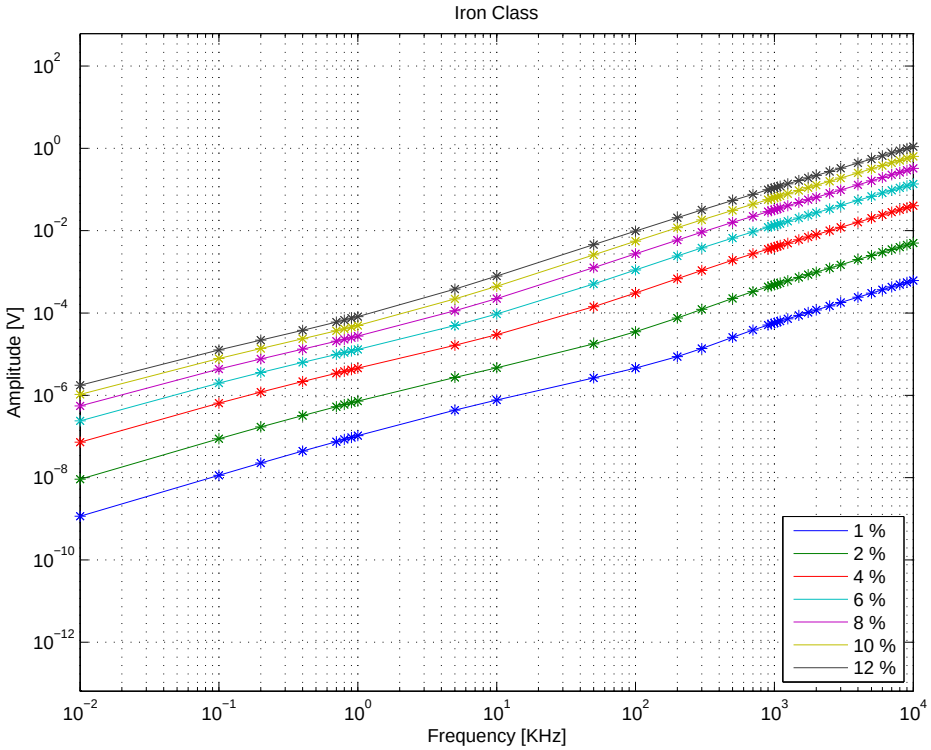

Figure E.2: Plot of the amplitude of the magnetic conductive class of materials, given as a function of frequency. Each curve represent the size of the object, given as percentage size of the detector head opening.

The figure shows, that the amplitude of the signal increases as function of frequency, which means, that the amplitude is dependent of frequency. The amplitude of the signal is size dependent, which means that the amplitude increases as the size of the object is increased.

The dry food class Fig. E.4 shows the amplitude as function of frequency, for different object sizes, belonging to the dry food class.
The figure shows, that the amplitude increases when frequency increases. The class has a frequency, at which the amplitude starts to decrease (local maximum) and a frequency, where the amplitude starts to increase (local minimum). The figure shows, that the amplitude of the signal depends on size of the object, as it is seen that the amplitude decreases, when the size decreases.

Amplitude Feature Discussion

In Fig. E.5 the change in permeability as function of frequency is shown.
The permeability decreases when the frequency increases, as described in Sec. 1.3. If Fig. E.5 is compared to Fig. E.4, the local maxima in Fig. E.4 is when the relative permeability approaches 1. This means, that the amplitude of dry food objects depends on the permeability when the frequency is below the local maxima and is independent of the permeability when the frequency is above the local minima. This can be interpreted as the object loses the magnetic properties at frequencies above the local minima. Objects from the iron class also loses the magnetic properties, but it does not affect the amplitude in the same way, because the high conductive properties of the

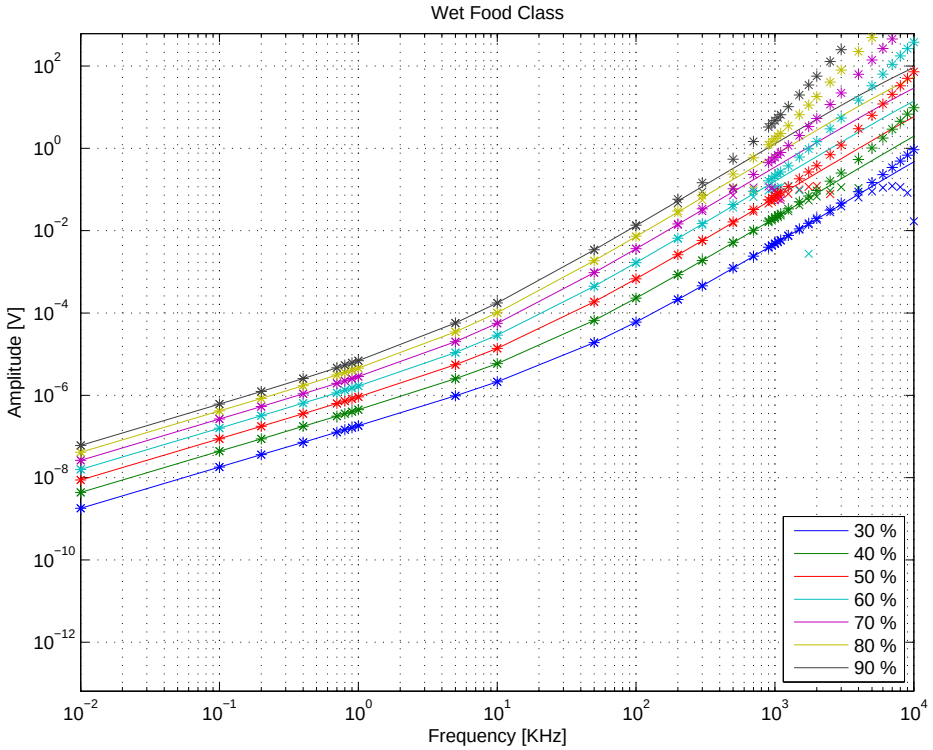

Figure E.3: The figure shows the amplitude as function of frequency for different object sizes belonging to the wet food class.

class is maintained.

If the amplitude of the different classes is compared, the amplitude of the dry food class is approximately a factor 1000 larger that the wet food class at low frequencies, and approximately equals at frequencies above the local minima. The iron class is approximately a factor 10 larger at low frequencies than the conductive metal class. In Fig. E.2 and Fig. E.1 it is seen, that the iron class and conductive metal class converges at higher frequencies. If the two opposite groups of classes are compared[1], the iron and conductive metal classes are approximately a factor 100 and 10000 smaller respectively. At low frequencies the wet food class is a factor 10 smaller than the iron class and a factor 10 larger than the conductive metal class. The amplitudes of the classes are approximate equal at high frequencies.

In the next section the phase feature of the classes is described.

E.2.3 Phase Feature

In this section the phase feature will be examined. The examination will be done with use of the four classes of materials described in Sec. E.2.1. The phase feature is based on the phase between the current in the transmitting coil and the voltage in the receiver coils.

[1] note the differences in object size between the classes

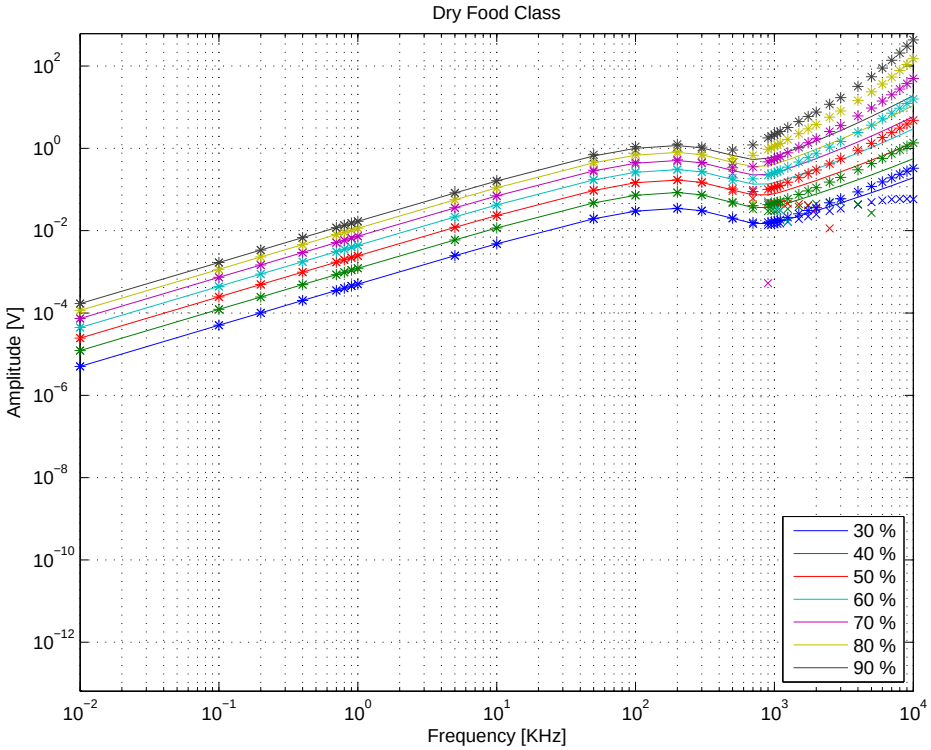

Figure E.4: Plot of the amplitude of the magnetic non-conductive class of materials, given as a function of frequency. Each curve represent the size of the object, given as percentage size of the detector head opening.

The conductive metal class Fig. E.6 shows the phase as function of frequency for different object sizes.
The figure shows, that the phase converges at $-\frac{\pi}{2}$, when the frequency is increased. The phase shift depends of the size of the object, which means that when the size is decreased, the phase will converge to $-\frac{\pi}{2}$ at a higher frequency.

The iron class Fig. E.7 shows the phase of objects from the iron class for different object sizes.

The figure shows that the phase at low frequencies is approximately $\frac{\pi}{2}$ and converges to $-\frac{\pi}{2}$ at high frequencies. The frequency at which the phase shifts from $\frac{\pi}{2}$ to $-\frac{\pi}{2}$ is dependent on the size. When the phase has converged to $-\frac{\pi}{2}$ the phase becomes independent of size.

The wet food class In Fig. E.8 the phase as function of frequency is shown.
The figure shows, that the mean of the phase is zero. This means, that the phase is independent of frequency. It shall be noted, that the spread of this case is much bigger that the other class, which means that the phase outcome of this class is strongly dependent of the specific material properties. The spread decreases with frequency, which means that the phase gets more independent of the materiel properties, when the frequency is increased.

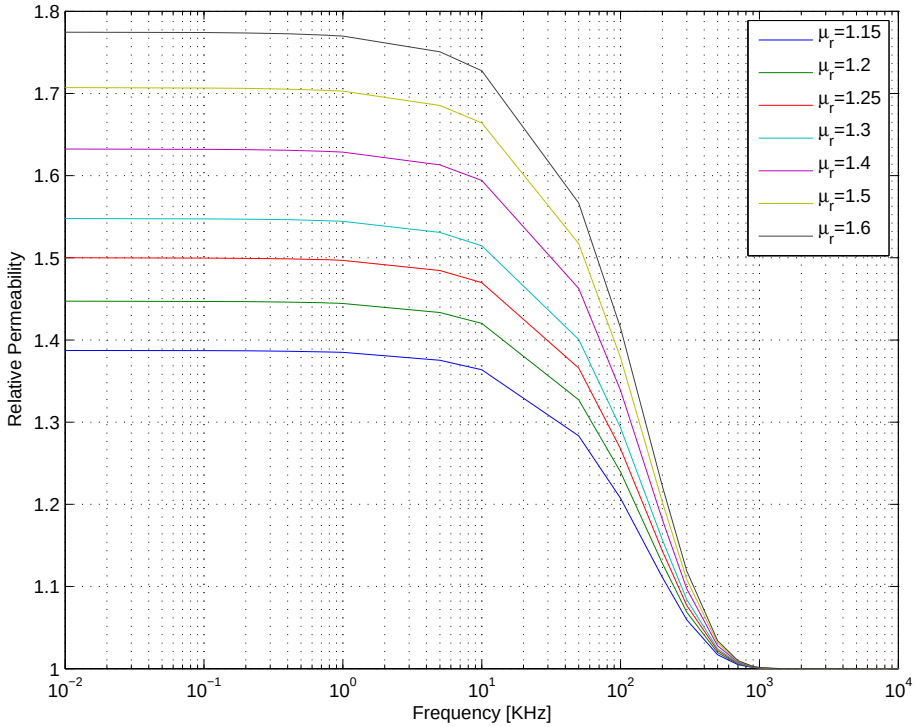

Figure E.5: Illustration of the relative permeability as a function of frequency.

The dry food class Fig. E.9 shows the phase of objects from the class of dry food objects as function of frequency for different object sizes.
The figure shows, that at low frequencies the phase shift is $\frac{\pi}{2}$. At a frequency, the phase starts to shift rapidly to 0 and then it approaches $-\frac{\pi}{2}$ less rapidly. The figure shows, that the phase is independent of size when the frequency is below the frequency at which the phase starts to shift away from $\frac{\pi}{2}$.

Phase Feature Discussion

The development in phase of the classes can be interpreted by use of Faraday's and Lenz's law. Pure magnetic non-conductive objects will make a phase shift of $\frac{\pi}{2}$ due to the fact that the objects will act as conductors for magnetic fields. This means, that the phase shift seen at the receiving coils will follow Faraday's law, which implies, that a time varying field moving through a closed loop will induce an electromotive force in the loop, which is phase shifted $\frac{\pi}{2}$, compared to the current in the loop producing the field.
For pure non-magnetic conductive objects, the field seen at the receiving loops will be phase shifted $-\frac{\pi}{2}$. This means, that these objects will follow Lenz's law, which means, when a field is moving through a conductive object, the object will produce a field in the opposite direction due to Eddy currents in the object. The field seen by the receiving coils is phase shifted π compared to the field seen for a purely magnetic non-conductive object. Objects which is both conductive and magnetic, will be phase shifted between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$, because these are the phase shift limits, due to Faraday's and Lenz's law. The phase shift of these objects will depend on frequency, conductivity, permeability and size.

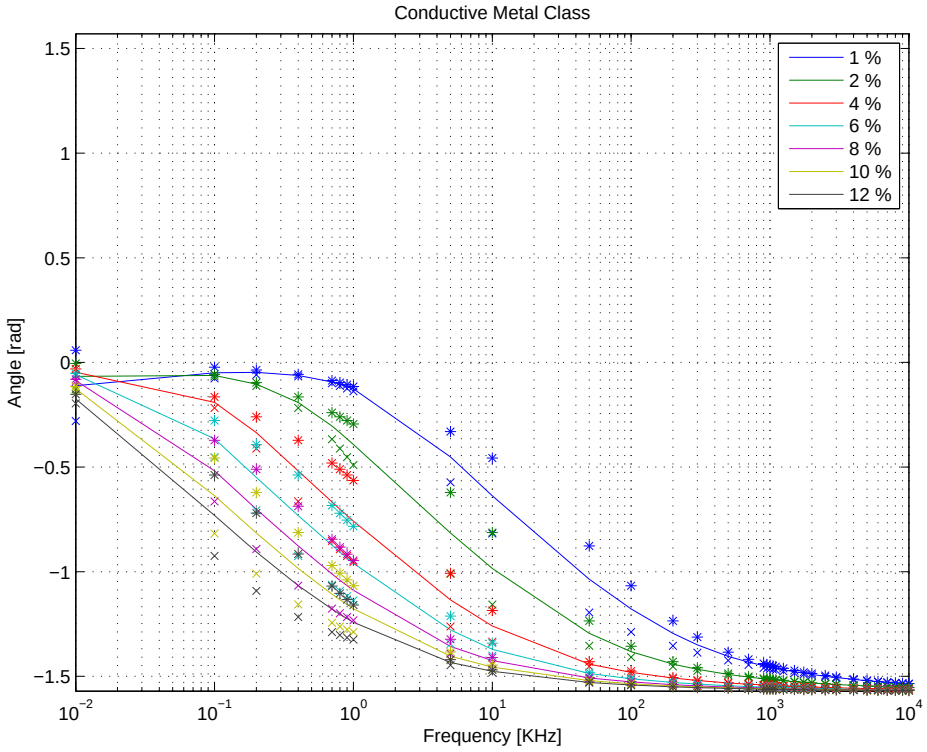

Figure E.6: Plot of phase of the conductive metal class, given as function of frequency. Each curve represent a given size of the obejct, given as percentage of the detector head opening.

The class of conductive metal objects can be expected to follow Lenz's law, which means that the phase is shifted $-\frac{\pi}{2}$. Fig. E.6 shows that at low frequencies, the class is dependent on the permeability, but converges to $-\frac{\pi}{2}$, which means the object is independent of the permeability and only dependent on the conductivity at high frequencies.
The class of iron objects shown Fig. E.6 follows Faraday's law at low frequencies, and as the permeability decreases with frequency the phase shifts toward $-\frac{\pi}{2}$, because the object starts to act as a non-magnetic conductive object.
The wet food class is not expected to act in a special way, because the objects of this class neither have strong magnetic or conductive properties. Fig. E.9 shows, that the phase strongly depends of the magnetic and conductive properties at low frequencies, because the spread of the simulation due to the fact that non of the properties are dominant. The phase of the class in mean cannot be expected to be the right phase due to the strong dependence of the properties. The mean estimate is a better estimator at high frequencies due to the decrease i spread.
The dry food class is expected to follow Faraday's law. Fig. E.9 shows that the class follows Faraday's law, but starts to shift in phase at higher frequencies, because the permeability decreases with frequency. When the magnetic properties have disappeared, the object starts to act like the wet food class.

In the next section the two features are discussed.

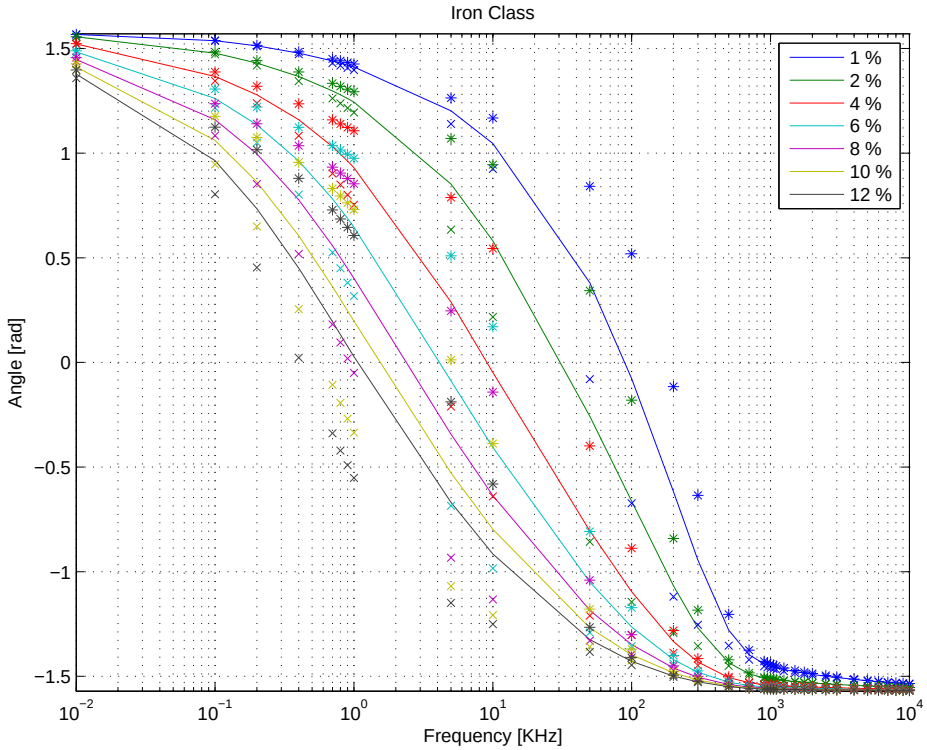

Figure E.7: Plot of phase of the iron class, given as function of frequency. Each curve represent a given size of the obejct, given as percentage of the detector head opening.

E.2.4 Feature Discussion

In this section the optimum frequency by means of the features are identified. The optimum frequency is the frequency at which the features of the conductive metal and iron classes are most correlated and most uncorrelated with wet and dry food classes, because it is only the conductive metal and iron classes which is wanted to be detected. Looking at the amplitude feature the optimum frequency is the local minima of Fig. E.4, because the conductive metal and iron classes are monotonically increasing, when the dry food class has local minimum. Looking at the phase feature, the optimum frequency is when the conductive metal and iron classes has converged to $-\frac{\pi}{2}$ and the dry food class has not converged to $-\frac{\pi}{2}$. In general it can be concluded, that the optimum frequency is placed at the local minimum of the dry food class. It can also be concluded that the conductive metal and iron classes are approximately independent of size in means of the phase feature, at this frequency. From the simulations, the optimum frequency will be placed between 1 MHz and 2 MHz.

E.3 Sampling of Trajectory

In this appendix sampling of the amplitude and phase trajectory of a moving object in the metal detector head is analysed. The trajectory represents the amplitude ratio and phase difference between the reference current $d[n]$ and the differential observation voltage $u[n]$. The complete trajectory is given by the Short Time Fourier Transform (STFT) $X[m]$ of length M, $m = \{q : q - M + 1\}$, in the *a priori* transmitting frequency (omitted, because of narrowband application) at present

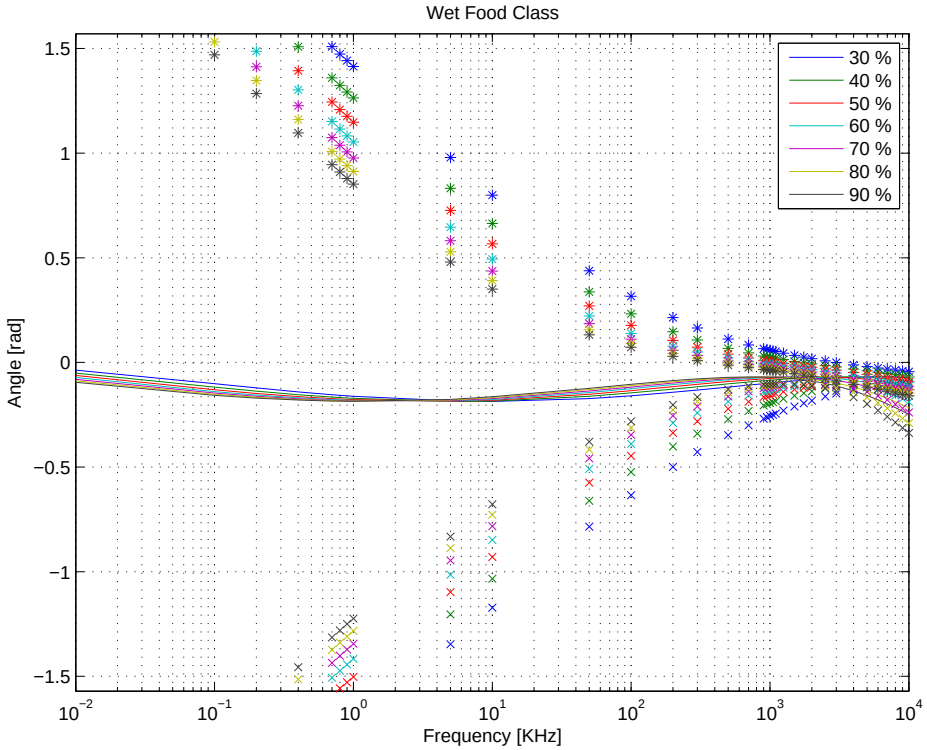

Figure E.8: Plot of phase of the wet food class, given as function of frequency. Each curve represent a given size of the obejct, given as percentage of the detector head opening.

time instance q. The minimum length of $X[m]$ is determined from the frequency content of the trajectory, because Shannon's sampling theorem should be meet.

To determine the necessary sample frequency of the trajectory, the model developed in Cha. 1 and measurements obtained through App. C are used for analyses. The analysis is based on Fourier transform of the real part of the complex trajectory $X[m]$. The measurements used are for a 5 mm iron sphere, which can be compared to the model, but also a package of unpopped microwave popcorn among two other products. Also noise is included. The trajectory of a metal sphere was determined in Cha. 2 to have two maxima, and the popcorn was determined to have four. Therefore it is expected, that the frequency content of the popcorn trajectory is larger than that of the metal sphere.
The time of each trajectory is 1.28 sec, when the speed of the conveyor is set to approximately 25 meters per minute. In Fig. E.10 the DFT of the real part of the trajectories are shown, where the x-axis is scaled to Hz .

In the upper part og the figure, the iron sphere and model is shown, and in the lower part three products are shown, together with noise. In all, 31 bins are included, giving the highest frequency of 30 Hz . For the metal and the model, the highest and significant bin is approximately 2 to 3 Hz . Looking at the products and the two types of noise, a significant component is seen around 27 Hz . This component is also present in the noise types, so it is not concluded as part of the signal

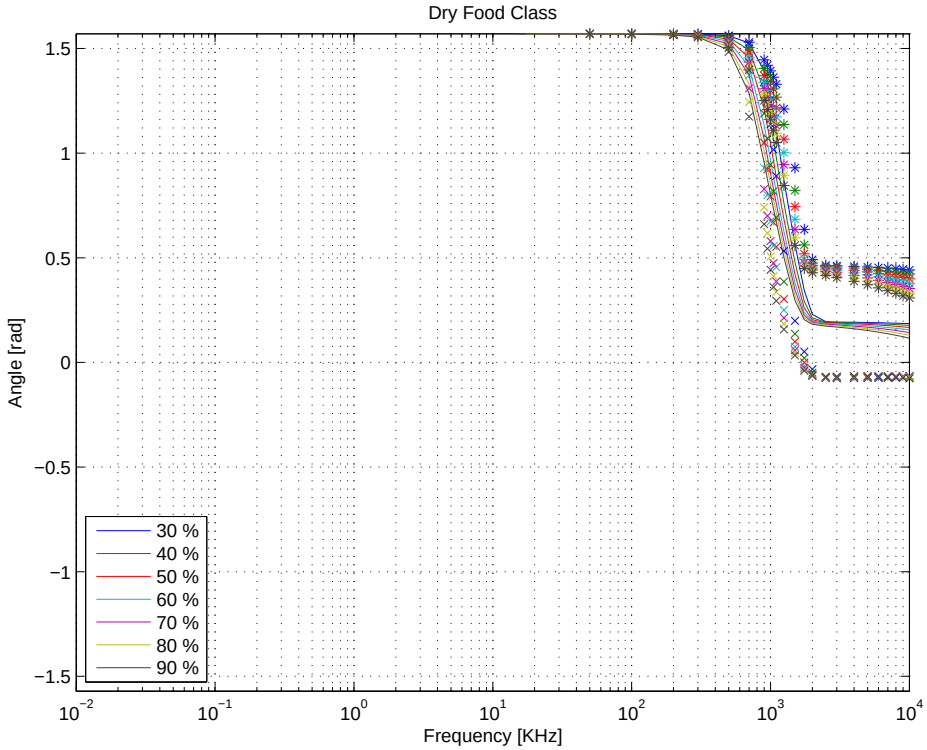

Figure E.9: Plot of phase of the dry food class, given as function of frequency. Each curve represent a given size of the obejct, given as percentage of the detector head opening.

component. However, because it is present in the transmitting frequency, it cannot be removed, before the spectra has been computed. The highest frequency for the products, is seen to be approximately 5 to 6 Hz .
To meet Shannon's sampling theorem the sample frequency should be larger than or equal to twice the Nyquist frequency. The highest frequency component was observed in the noise, which gives a minimum sample frequency of $f_{s,min} > 2f^{Nyquist} = 54$ Hz , corresponding to a minimum length of $X[m]$ of $M_{min} = 1.28 f_{s,min} \approx 69$ complex samples. Including a "security factor" on the sample frequency the conclusion becomes,

$$f_{s,min} = 200 \text{ Hz} \tag{E.12}$$

$$M_{min} \approx 250 \text{ samples per complete trajectory} \tag{E.13}$$

which is valid, when the speed of the conveyor belt is approximately 25 meters per minute or less. Increasing the speed of the conveyor belt will require a larger sample frequency of the trajectory. When the trajectory is obtained, it can be lowpass filtered and downsampled if needed.

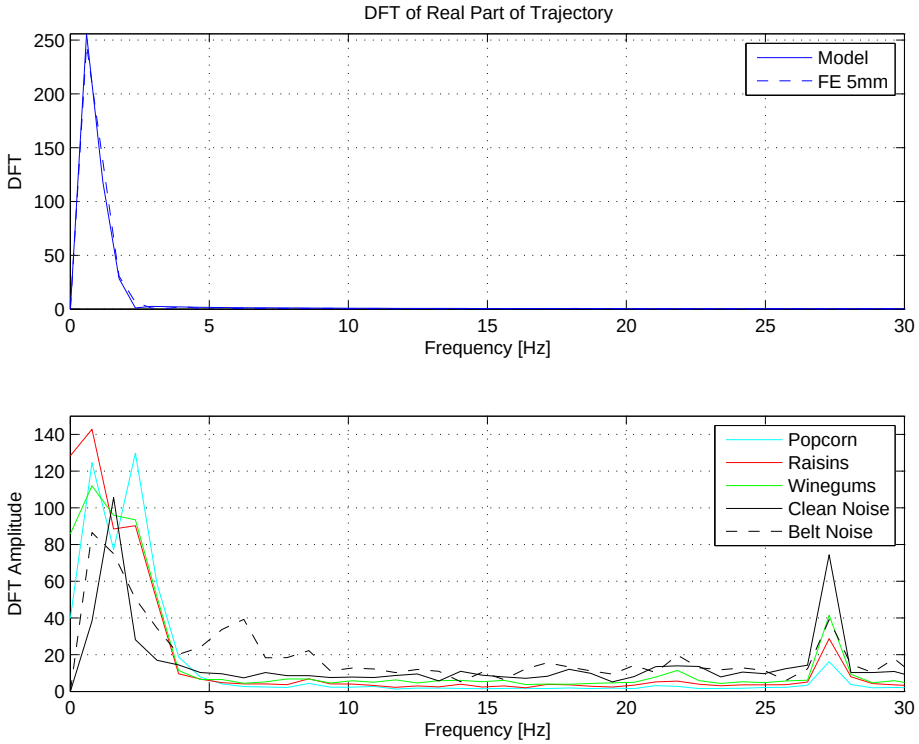

Figure E.10: DFT of the real part of the trajectories based on the model and 5 mm FE in the upper figure, and three products and noise in the lower figure.

Appendix F

Noise Model of Spectrum Computations

This appendix models the effects of truncation before and after the accumulator in terms of quantisation noise. The reason for modelling this topic is to see the effect of reducing the size of the accumulator. The section also handles the topic of coefficient quantisation, which occur due to the limited representation of the coefficients in a the fixed point implementation. The first topic handles the aspect of truncation before and after the accumulator.

F.1 Modelling of Truncation Error before and after the Accumulator

The topic is analysed to see the effect of truncation before and after the accumulator in terms of truncation noise introduced by truncation. In Fig. F.1 two noise models are setup. Part a)

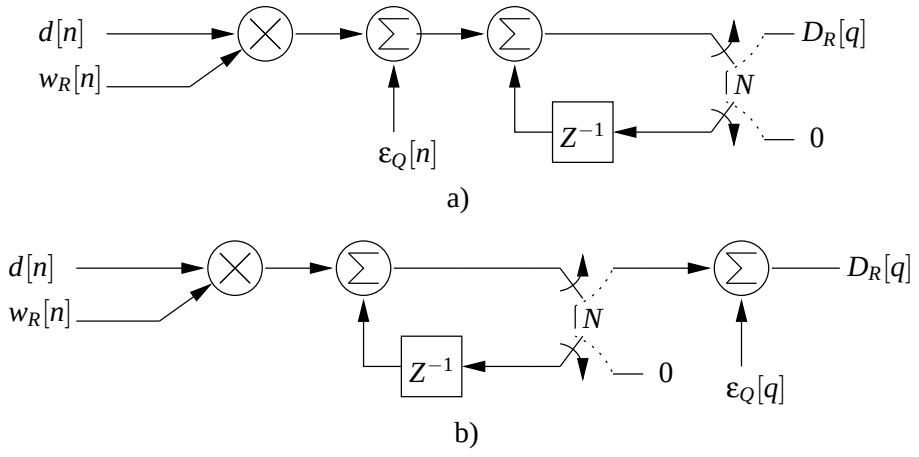

Figure F.1: Signal flow graph of the noise model

describes the truncation before the accumulator and part b) the truncation after the accumulator. The terms $\epsilon_Q[n]$ and $\epsilon_Q[q]$ is a white noise variable modelling the computation noise added as a

consequence of truncation. The noise term is expressed as,

$$\epsilon_Q[n] = w_R[n] \cdot d[n] - Q(w_R[n] \cdot d[n]) \tag{F.1}$$

where the function $Q()$ truncate the result to B bits. From Eq. F.1 it can be seen that the error term $\epsilon_Q[n]$ for all n will be positive or zero, because $w_R[n] \cdot d[n] \geq Q(w_R[n] \cdot d[n])$ caused by truncation. Truncation implies infinite precision values to be placed on some fixed values between [-1:1[. The resolution of the steps between [-1:1[is determined by the number of bits used in the truncation. The step size between [-1:1[is defined as $2^{(-(B-1))}$. The minus 1 origin from the fact the one of the bits in the word is a sign bit. Due to the step size, the interval for the noise term $\epsilon_Q[n]$ can be defined as $2^{(-(B-1))}$,

$$0 \leq \epsilon_Q[n] < 2^{(-(B-1))} \tag{F.2}$$

The noise in assumed to be uniformly distributed, wide sense stationary and independent of the input to the function $Q()$. [Oppenheim et al., 1999] discusses the independent assumption and state that the assumption is valid for wideband signals. It is assumed that a quantisation sinusoidal satisfies this assumption, as long as the sinusoidal is quantified to eight or more bits, see [Oppenheim et al., 1999] page 195 for a graphical presentation. The mean and variance of the model can be calculated as in [Oppenheim et al., 1999],

$$\mu_{\epsilon_Q} = \int_0^{2^{(-(B-1))}} x \cdot P_{\epsilon_Q}(x)dx = [\frac{x^2}{2 \cdot 2^{(-(B-1))}}]_0^{2^{(-(B-1))}} = \frac{2^{(-(B-1))}}{2} \tag{F.3}$$

$$\sigma^2_{\epsilon_Q} = \int_0^{2^{(-(B-1))}} x^2 \cdot P_{\epsilon_Q}(x)dx - \mu^2_{\epsilon_Q} = [\frac{x^3}{3 \cdot 2^{(-(B-1))}}]_0^{2^{(-(B-1))}} - \mu^2_{\epsilon_Q} = \frac{2^{(-(B-1))}}{12} \tag{F.4}$$

where it is seen that the mean and variance of the computation noise is dependent of the probability density function $P_{\epsilon_Q}(x)$. If the uniform distribution assumption does not hold, the variance and mean will be biased. The mean and variance of the computation noise is now known. The following sections handle the truncation before and after the accumulator. The goal in the two approaches is to find the influence of the noise term at $D_R[q]$.

F.1.1 Truncating before the Accumulator

The output $D_R[q]$ is in term of the truncation noise and the signal expressed as,

$$D_R[q] = \sum_{n=0}^{m \cdot p-1} p[n] - \epsilon_Q[n] \tag{F.5}$$

where it is seen that $D_R[q]$ is a sum of the error terms and signal terms. To describe the effect of truncation, $p[n]$ is set to zero for all n, which implies that the mean quantisation noise μ_{D_R} at the output is expressed as,

$$\mu_{D_r} = \sum_{n=0}^{m \cdot p-1} \mu_{\epsilon_Q} = \mu_{\epsilon_Q} \cdot m \cdot p \tag{F.6}$$

where it is seen that the mean error at the input is summed up at the output of the accumulator. This means that the input mean error increases proportional with the number of accumulations performed. To find the mapping of the input error variance to the output, the power spectrum

of the error terms has to be observed, because the variance expresses the power of a signal. The mapping in form of power spectrum is expressed as,

$$S_{D_r D_r}(\omega) = |H(\omega)|^2 S_{\epsilon_Q \epsilon_Q}(\omega) \tag{F.7}$$

where S denote the power spectrum and $H(\omega)$ is the frequency response of the function, which transfers ϵ_Q to D_r. The error term is assumed to be white, which implies that the energy is spread uniformly in the power spectrum. Eq. F.7 can be rewritten to,

$$\sigma^2_{D_R} = \sum_{n=0}^{m \cdot p - 1} |h[n]|^2 \sigma^2_{\epsilon_Q} \tag{F.8}$$

$$\sigma^2_{D_R} = m \cdot p \cdot \sigma^2_{\epsilon_Q} \tag{F.9}$$

where it is seen that the quantisation error variance increases proportional with the number of accumulations. The proportional increase in both mean and variance is an unwanted effect. The mean value can be considered as an under estimation of all estimates of $D_R[q]$ and the variance $\sigma^2_{D_R}$ is the variance around the underestimated value. In the next section the case of truncation after the accumulator is described.

F.1.2 Truncating after the Accumulator

The truncating before the accumulator showed proportional increase in both variance and mean value. The truncation after the accumulator is basically a truncation without the sum of means and variances. The mean and variance can be expressed as,

$$\mu_{D_R} = \mu_{\epsilon_Q} \tag{F.10}$$

$$\sigma^2_{D_R} = \sigma^2_{\epsilon_Q} \tag{F.11}$$

where it is seen that the mean and variance at the output is equal to the mean and variance of the truncation noise term.

F.1.3 Coefficient Quantisation

The effect of coefficient quantisation noise occurs, when the coefficients only can be represented in a limited number of bits. The coefficient quantisation noise cannot be modelled as a linear noise model, because it depends of the input of the filter. To find the effect of coefficient quantisation the frequency response of the filter has to be observed. The filter of the DFT spectrum computation is a FIR filter where the coefficients are a sine or cosine. The coefficients are shown periodic in a interval of p samples, in Sec. 5.1.1, which implies.

$$w_R[n] = w_R[n + p] \tag{F.12}$$

If p is even. The definition of $w_R[n]$ can further be extended to

$$w_R[n] = -w_R[n + p/2] \tag{F.13}$$

which implies that $w_R[n]$ will be zero mean over one period. The coefficients consist of m periods of p samples, which imply that the coefficients $w_R[n]$ is zero mean. This implies that the amplification

of DC signals is zero. Due to the periodicity of the coefficients, $w_R[n]$ can be expressed as a trigonometric Fourier series, which states that $w_R[n]$ can be expressed as a sum of the fundamental frequency and the harmonic frequencies. The fundamental frequency in this case is the peak of the filter where the spectrum is computed. The optimum filter, is the filter where only the fundamental frequency is present in the frequency response. This can be interpret as the pass band of a bandpass filter. The influence of quantisation is addition of harmonic frequencies in the frequency response of the filter. This means that the bandpass filter due quantisation get more than one pass band, where the new pass bands are placed in the harmonic frequencies. When quantisation is introduced and

$$Q(w_R[n]) = Q(w_R[n+p]) \tag{F.14}$$
$$Q(w_R[n]) = -Q(w_R[n+p/2]) \tag{F.15}$$

is satisfied, which is possible for two and more bits, the fundamental frequency of the filter will still be the same regardless of the quantisation of the coefficients. This means that the peak (fundamental frequency) in the bandpass filter does not move in frequency. It can be explain by an example. A square wave consists of a fundamental frequency determined by the period time T and some harmonic frequencies. A sinusoidal also consists of a fundamental frequency determined by the period time T. If the sinusoidal is quantified such that Eq. F.14 and Eq. F.15 are satisfied, that period time T will not change which implies that the fundamental frequency does not change. If the sinusoidal is quantified to two bits, the sinusoidal will be a square wave with period T. This property is seen as an advantage, because the fundamental frequency of the filter is wanted to be independent of the quantisation. To find the influence of the harmonic components in the frequency response the power distribution in the spectrum has to be observed. It shall be noted, that the power in time has to be the same as the power in the frequency domain. This means that by coefficient quantisation in time, the power in time should equal the power in frequency, which implies that the power of the fundamental frequency is shared with the harmonic frequencies. This turns out as an reduction in gain in the fundamental frequency. To describe the share between the fundamental and harmonic frequencies a simulation has been performed, where the power in the fundamental frequency is seen as the signal component, and the power outside the fundamental frequency as noise. This lead to the following definition of the power share ratio.

$$SNR_s = 10 \cdot log(\frac{P_s}{P_n}) \tag{F.16}$$

The simulation results are shown in Fig. F.2. The figure shows that the sharing ratio increases, as the size in bits of the word increases. This means that the power in the harmonic frequencies decrease as the size of the word is increased. The power decrease in the harmonic frequencies results in an increase in power in the fundamental frequency, due to power preservation.

The cost of choosing $Q(w_R[n])$ such that Eq. F.14 and Eq. F.15 is satisfied, is a reduction in power of the coefficients, compared to the un-quantised $w_R[n]$. The power loss ratio is expressed as,

$$SNR_l = 10 \cdot log(\frac{\sigma^2_{w_R[n]}}{\sigma^2_{Q(w_R[n])}}) \tag{F.17}$$

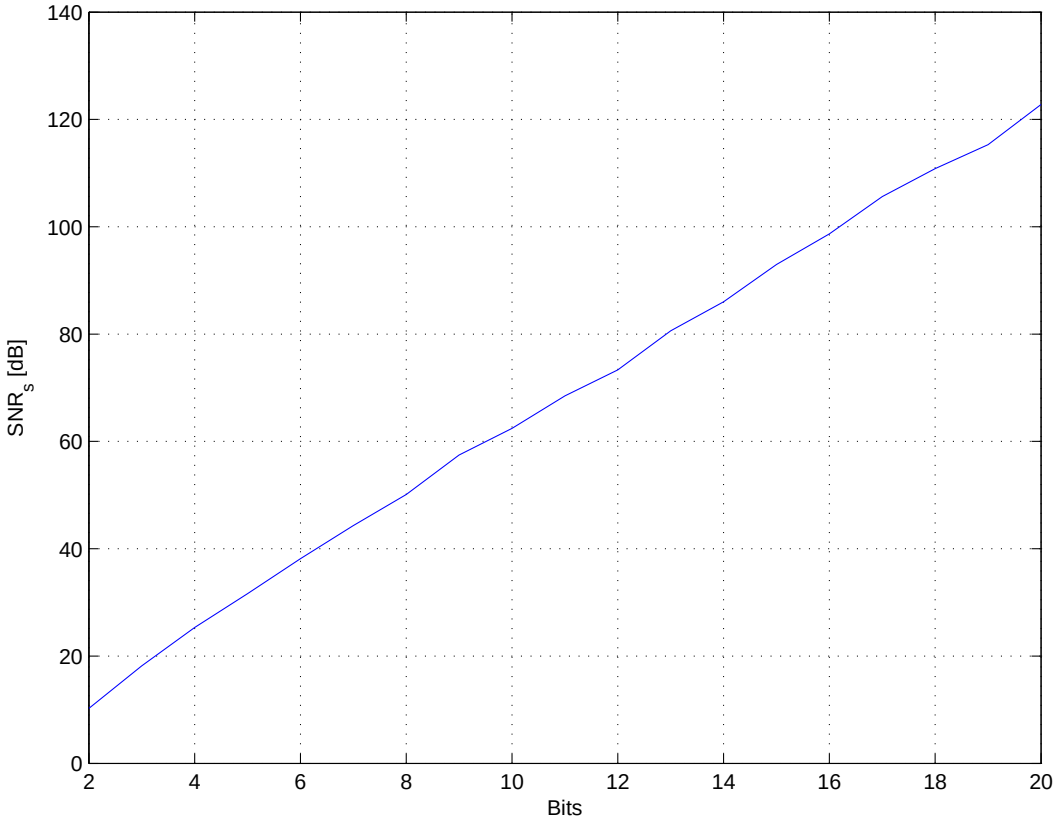

Figure F.2: The figure shows the power sharing between the fundamental and harmonic frequencies of the filter frequency response.

where $\sigma^2_{w_R[n]}$ is the variance of $w_R[n]$ in infinite precision and $\sigma^2_{Q(w_R[n])}$ is the variance of the quantised version. Fig. F.3 shows the power loss ration defined Eq. F.17 for quantisations of $w_R[n]$ from 2 to 20 bits. The figure shows that the power loss ratio is approximately zero at 11 bits, which implies that $w_r[n] \approx Q(w_r[n])$ and the loss in power gets insignificant when the word size is larger than 11 bits. In the next section the conclusions for the sections are drawn.

F.2 Conclusion

In the section the effects of truncation before/after the accumulator and coefficient quantisation was analysed. The analysis of truncation showed, that the mean and variance of the quantisation error was proportional with the number of accumulations when the truncation was introduced before the accumulator. When the truncation was performed after the accumulator, the mean and variance of the quantisation error only dependent of the number of bits for truncation. It is seen as an advantage to truncate after the accumulator, to avoid the increase in quantisation noise. The cost of truncation after the accumulator is an increase in size of the accumulator, but is assumed to be less costly than quantisation noise introduced for large numbers of $m \cdot p$. The derivation of the noise model is based in three assumptions. Two of the assumptions is that the quantisation is independent of the input of the quantifier $Q()$ and the errors are uniform distributed. The independence assumption secure that the probability density function of the error does not change. It

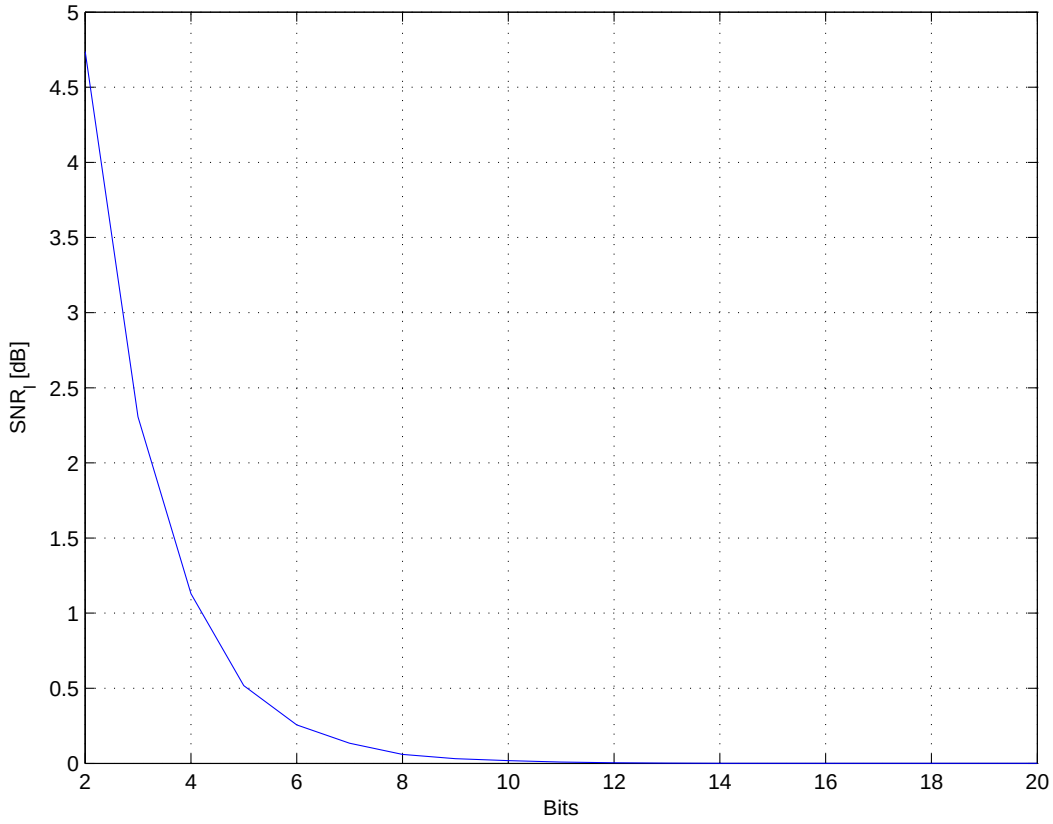

Figure F.3: The figure shows the power loss ration due to quantisation of the coefficients.

is assumed the assumptions holds when the sinusoidal is truncated to eight or more bits.

In the analysis of quantisation noise, it was shown that the coefficients can be chosen such that the peak of the filter does not change regardless the quantisation. The cost of choosing the coefficients in this way, was a reduction in power, but when the quantisation is 11 bits or more the reduction gets insignificant. The power in the harmonic frequencies was seen to increase as the number of bits decreased. Due to power preservation, this reduced the gain in the fundamental frequency.

Appendix G

Analog Devices AD9220 Schematic

This appendix documents the schematic for the AD9220 A/D converter. The design is based on Devices [2003]. In Fig. G.1 the circuit diagram is presented.

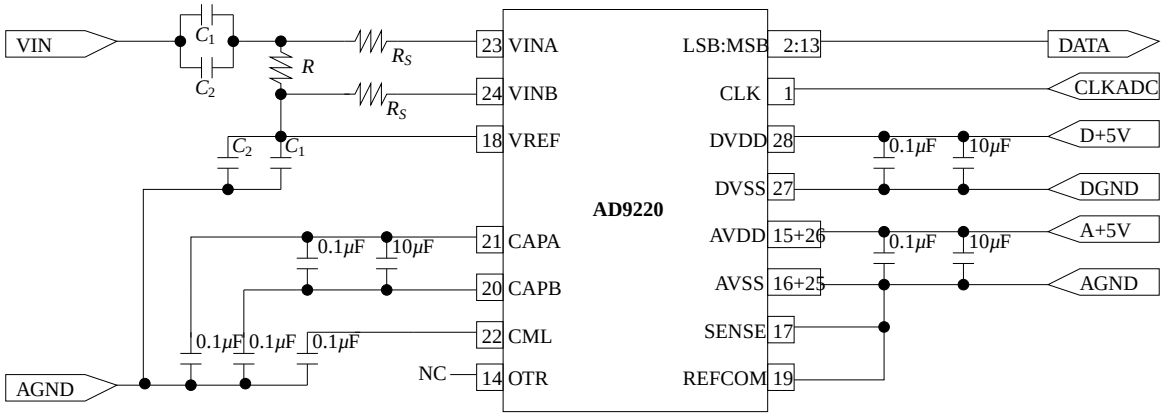

Figure G.1: Schematic for AD9220.

Details are found in Devices [2003], together with component values, except from the capacitor values C_1 and C_2, and the resistances R and R_s. In Devices [2003] the capacitor values are argued to be,

$$C_1 = 0.1\ \mu\text{F} \tag{G.1}$$

$$C_2 = 10\ \mu\text{F} \tag{G.2}$$

The resistor R and the capacitors C_1 and C_2, form a highpass filter for AC coupling. The resistor R is calculated as,

$$R = \frac{1}{2 \cdot \pi \cdot f_{3\text{dB}} \cdot (C_1 + C_2)} = 157.58\ \Omega \tag{G.3}$$

when the -3 dB highpass frequency $f_{3\text{dB}}$ is chosen to 100 Hz . The resistor R_s is in Devices [2003] argued to be optimal in the interval from 30 to 50 Ω, according to total harmonic distortion (THD). The resistor R_s is chosen to,

$$R_s = 40\ \Omega \tag{G.4}$$

The clock input to the A/D converter CLKADC is approximately 3.84 MHz, giving a Nyquist frequency of approximately 1.92 MHz. A first order lowpass filter is placed before the analog input VIN to the circuit, and the cutoff frequency is set to 35 KHz, resulting in more than 30 dB attenuation of frequencies above the Nyquist frequency.
The digital supply (DVDD and DVSS) is obtained from the DE2 Development Board through the 40 pins external connectors (Expansion Header, JP2 (GPIO 1)). From pin 11 a +5 V supply is obtained, and the digital ground is obtained from pin 12. The analog power supply is obtained from digital power supply, by adding more decoupling capacitors. The solution for the analog supply is not optimal, due to noise from the digital hardware. It is however chosen to avoid the extra cables to a external power supply.
The overflow pin OTR (out-of-range) can be used for the level detection in the digital hardware, but this is chosen to be done in the software instead. So this output pin is floating.

Appendix H

Simulation of Gate Delays for Cyclone II FPGA

The following propagation delays through various logic gates,

Delay	Input Pin	Output Pin
9.381 ns	AND1_in	AND_out
9.135 ns	AND2_in	
5.244 ns	NAND1_in	NAND_out
5.420 ns	NAND2_in	
8.837 ns	OR1_in	OR_out
8.578 ns	OR2_in	
9.267 ns	XOR1_in	XOR_out
9.423 ns	XOR2_in	

Table H.1: Propagation Delays for various gates simulated in Quartus for the Cyclone II FPGA. The gates simulated are: AND, NAND, OR and XOR.

The propagation delays are obtained from simulation in Quartus, where the AND, NAND, OR and XOR gates are synthesised. The input pins are denoted by GATE1_in and GATE2_in, and the output pin GATE_out. The largest value for the propagation delay should be used.