Eteläpää, Jere

# Engine monitoring based on harmonic saliencies

School of technology and innovation management
Master's thesis in Technology
Automation and Computer science

Vaasa 2021

| | |
|---|---|
| **UNIVERSITY OF VAASA** | |
| **School of technology and innovation management** | |
| **Author:** | Eteläpää, Jere |
| **Title of the Thesis:** | Engine monitoring based on harmonic saliencies |
| **Degree:** | Master of science in technology |
| **Programme:** | Automation and Computer science |
| **Supervisor:** | Timo Mantere |
| **Instructor:** | Leif Strandberg |
| **Year:** | 2021 **Pages:** 69 |



**ABSTRACT:**

The fundamental frequency of the powertrain's rotational velocity (speed) corresponds to one of the peaks in a frequency spectrum. By processing the speed signal data in the frequency domain, the signal can be decomposed into separate relevant components. Harmonic orders of the fundamental frequency correspond to various engine internal events, for example combustion events. The main objectives of this thesis are, to define relevant engine events based on their harmonic saliencies. Create a concept method for performing monitoring and speed estimation on UNIC HW and evaluate the reliability and performance of the method.

First the concept method is designed with the help of Dewesoft and MATLAB softwares, which help in the process of analysing real engine speed signal data. The harmonics are also further investigated, to find if they hold relevant information that can be used in engine monitoring. After the design of the method is complete, it is then implemented on the UNIC module. When the implementation is loaded to the module testing and evaluation can begin, and it was done with a minirig. Minirig consists of a multiple UNIC system modules such as CCM-30 and COM-10.

The concept method appears to be promising in calculating the engine speed. This method could be used on a real engine to provide relatively robust speed estimation and possibly replace the current methods in that area. The largest negative side of the method is the fact that in the relatively low flywheel revolutions per minute, the frequency domain data becomes unclear and difficult to perform any calculations with. Nevertheless, when calculating the speed, harmonics can also be calculated at the same time and from the same frequency domain data. This makes the concept even more appealing, because the harmonics include beneficial information about the engine's state. The information related to harmonics can mainly be used to monitor if the engine is in normal condition, and no faults are present.


**KEYWORDS:** engine harmonics, frequency domain, speed signal

**VAASAN YLIOPISTO**
**Tekniikan ja innovaatiojohtamisen yksikkö**
**Tekijä:** Eteläpää, Jere
**Diplomityön nimi:** Moottorin seuranta ylä-äänten avulla
**Tutkinto:** Diplomi-insinööri
**Oppiaine:** Automaatio ja tietotekniikka
**Valvoja:** Timo Mantere
**Ohjaaja:** Leif Strandberg
**Vuosi:** 2021 **Sivumäärä:** 69

**TIIVISTELMÄ:**

Voimansiirron perustaajuus (nopeus) vastaa yhtä piikkiä taajuus spektrissä. Prosessoimalla nopeus signaalin dataa taajus alalla, signaali pystytään jakamaan eri komponentteihin, jotka sisältävät hyödyllistä tietoa. Perustaajuuden ylä-äänet vastaavat moottorissa tapahtuvia tapahtumia, esimerkiksi sytytykseen liittyviä ilmiöitä. Tämän diplomityön pää tarkoitus on määrittää olennaiset moottorin tapahtmuat perustuen niitä vastaaviin ylä-ääniin. Tehdä konsepti menetelmä, jolla voi seurata moottorin tilaa ja sen nopeutta UNIC laitteistolla. Myös menetelmän luotettavuus ja toimintakyky tulee arvoida.

Aluksi konsepti menetelmä suunniteltiin Dewesoft ja MATLAB ohjelmistojen avulla. Nämä ohjelmistot auttoivat moottorin nopeus datan analysoinnissa. Ylä-ääniä tutkittiin, jotta niissä mahdollisesti olevan tiedon voisi hyödyntää moottorin seurannassa. Konsepti menetelmän suunnittelun jälkeen, menetelmää alettiin rakentamaan UNIC moduulille. Kun implementaatio tultua valmiiksi menetelmän testaus ja arviointi voitiin aloittaa. Testauksessa hyödynnettiin minirig laitteistoa, joka koostuu monesta UNIC systeemin moduulista, esimerkiksi CCM-30 ja COM-10 moduuleista.

Konsepti menetelmä vaikuttaa olevan lupaava moottorin nopeuden laskemiseen. Menetelmää voisi hyödyntää oikealla moottorilla, sillä se on suhteellisen vakaa ja toimiva laskiessa nopeutta, ja menetelmä voisi mahdollisesti korvata nykyiset moottorin nopeuden laskemiseen käytetyt menetelmät. Isoin negatiivinen asia menetelmässä on se, että vauhtipyörän kierrosten ollessa todella alhaalla, taajuus-alan datasta tulee epäselvää ja sillä on vaikea tehdä kunnollisia laskelmia. Kuitenkin, kun moottorin nopeutta lasketaan voidaan samalla laskea ylä-äänet, samasta taajuus-alan datasta. Tämä tekee menetelmästä vielä vetoavamman, koska ylä-äänten datassa on hyödyllistä tietoa koskien moottorin tilaa. Ylä-äänten informaatiota voi pääasiassa käyttää hyödyksi, kun halutaan tarkistaa onko moottori normaalissa tilassa ilman vikoja.

**AVAINSANAT:** moottorin ylä-äänet, taajuus-ala, nopeus signaali

## Contents

## Pictures

## Tables



## Abbreviations

| | |
|---|---|
| DSP | Digital signal processing |
| FPGA | Field programmable gate array |
| ADC | Analog to digital converter |
| ASIC | application specific integrated circuit |
| HW | Hardware |
| DFT | Discrete Fourier transform |
| FFT | Fast Fourier transform |
| STFT | Short-time Fourier transform |
| Rpm | Revolutions per minute |
| Hz | Frequency |

# 1 Introduction

The fundamental frequency of the powertrain's rotational velocity (speed) corresponds to one of the peaks in a frequency spectrum. By processing the speed signal data in the frequency domain, the signal can be decomposed into separate relevant components. Harmonic orders of the fundamental frequency correspond to various engine internal events, for example combustion events.

The objectives of this thesis are, to define relevant engine events based on their harmonic saliencies. Create a concept method for performing monitoring and speed estimation on UNIC HW. Afterwards, it is important to evaluate the robustness, reliability and performance of the method.

Motivation for this thesis includes the problem that current methods are ineffective, supposedly error prone and unreliable, thus in need of improvements. If the concept method of this study is promising, the current methods could ultimately be replaced with this new method.

For additional help and source material, pre-existing material can be used as a starting point. There is also an old implementation developed on another platform, that can be used as an example in developing the new concept methods.

## 1.1 Objective of the thesis

The main objective of this thesis is to define relevant engine events based on their harmonic saliencies. This is done by creating a concept method for performing the monitoring and speed estimation on UNIC HW. After creating the concept method, the characteristics such as robustness, reliability and performance of the method should be evaluated. Afterwards the new concept method could replace current methods that are ineffective and unreliable.

## 1.2 Structure of the thesis

This thesis begins with the theoretical background, going through important parts of the related theory. Chapter 2 introduces the diesel and gas engine and it's applications, in order to get better understanding of the machinery at hand.

Chapter 3 goes through digital signal processing related theory and information, whereas the chapter 4 gives insight on the Wärtsilä UNIC engine control system.

Chapter 5 focuses on Frequency analysis. It includes topics like frequency domain, Fourier transform, fundamental frequency and harmonics.

After explaining theory and related information, the thesis proceeds to the part where the concept method is designed. First the real data from an engine is being examined with Dewesoft. Then the concept method is designed with MATLAB.

After the design part comes implementation of the method, and that is followed by the evaluation. The thesis ends in discussing conclusions and possible future work.

## 1.3 Sources

This thesis uses sources like ScienceDirect to find relevant information to gain knowledge from. The thesis utilizes also source like Wärtsilä, in order to get more detailed information about the topic and related subjects. The University course material can also be used for example to explain mathematical formulas related to frequency domain and specifically Fourier transform. Articles from the web, are also used to give information about general subjects related to the topic of the thesis.

# 2 Diesel and gas engine

## 2.1 Diesel engine

Diesel engine is a type of an internal combustion engine. This means that the energy supplied by a burning fuel is directly converted into mechanical energy. This is caused by the controlled burning of the fuel in an enclosed space. In diesel engine, the ignitions are caused by the injection of the fuel into the hot and high-pressure air in the combustion chamber. The fuel is injected in the form of a fine spray using a nozzle into the combustion chamber. After injection, the fuel ignites by the heat of the compressed air, which the chamber has been charged with. (Wärtsilä 2020a & 2020b)

The diesel engine operates within a fixed sequence of events. These events can be achieved with either in two or four strokes. The four-stroke diesel engine has some similar characteristics as gasoline engine as it also works on the four-stroke cycle. This cycle includes admission, compression, power and exhaust. (Wärtsilä 2020a)

In the admission phase piston gets down and the lower pressure in the cylinder allows a charge of air into the cylinder through the inlet valve, which opens just before top dead centre. Then the compression phase begins, when piston has passed the bottom dead centre and begins to ascend. The inlet valve closes and the upward movement of the piston compresses the air charge in the cylinder causing a swift rise of the temperature. Before the second stroke (compression) is over, the charge of fuel is gradually injected into the cylinder by the injector. (Wärtsilä 2020a)

The burning of the air-fuel charge makes the gases expand, this causes the piston to be pushed downwards and the power stroke is created. Before the piston reaches the bottom dead centre, the exhaust valve opens. Then when the piston moves up again, the exhaust gases are forced out through the exhaust valve. Right before the top dead centre the inlet valve opens and the cycle begins anew. (Wärtsilä 2020a)

## 2.2 Gas engine

Gas engine, also known as the spark ignition engine, and the diesel engine differ from each other in how they supply and ignite the fuel. In spark ignition, the fuel is mixed with air and then inducted into the cylinder during the intake process. After the piston has compressed the fuel-air mixture, the spark ignites it and causes combustion. The expansion of the gases then pushes the piston during power stroke. (Energy.GOV 2013)

## 2.3 Main components of an engine

Engine consists of several components. The main components are Engine block, which can be made from modular cast iron in one piece for all cylinder numbers. Engine block can also be called as the ”backbone” of the engine and its material is often aluminium or steel. (Wärtsilä 2020 a) (Energy Education 2020)

Crankshaft is one of the most important parts of the engine. It is forged in one piece and it connects other engine parts together, thus enabling the engine to create power. The main idea of the crankshaft is to turn the linear motion of the pistons into rotational motion. (Energy Education 2020)

Connecting rod is the part that connects the piston and the crankshaft. The pistons up and down linear motion then moves the crankshaft rotationally with connecting rod. (Wärtsilä 2020 a)

Cylinder liner can be centrifugally cast, which leads to a high and rigid collar to minimize deformations. The liner material can for example be a special grey cast iron alloy developed for excellent wear resistance and high strength, in order to withstand the explosions caused by the constant ignitions of the fuel. Cylinder is the place where the work of the engine is done. This work is done by the piston that moves from the power of the fuel’s ignitions. (Wärtsilä 2020 a) (Energy Education 2020)

Pistons are the devices that move up and down in the cylinder. Pistons slide up and down inside the cylinder and are connected to the crankshaft by the connecting rod. Piston can be for example of a composite design with nodular cast iron skirt and steel crown. (Wärtsilä 2020 a) (Energy Education 2020)

Fuel injectors turn the liquid fuel into a mist, that greatly increases the fuels surface area. This enables the fuel to combust faster, thus generating greater impulse to the piston. (Energy Education 2020)

**Picture 1.** Wärtsilä 31 is a 4-stroke medium speed diesel engine. (Wärtsilä 2020c)

### 2.3.1 Powertrain

Powertrain includes all of the moving components in a vehicle to are critical vehicle’s success. The sole objective of the powertrain is to transform kinetic energy into propulsion motion. (Maxwell Ford 2020)

## 2.4 Use cases of diesel and gas engines

According to U.S Energy information administration, diesel fuel is important for U.S economy. Many of the products that are used in the U.S are being transported by trucks and trains equipped with diesel engines. Most of the vehicles in the construction, farming and military are also using diesel engines. Diesel engines are also used in the average pedestrian cars. Diesel fuel also has a greater energy density than other fuels, which provides it to be more useful energy per unit of volume. (U.S Energy Information Administration 2020)

Diesel engines can also be used to generate power, and they are one of the most common reciprocating engines used in power generating applications. High speed diesel engines can be used for example as an emergency and backup generators to provide power during electricity grid outages. Medium-speed diesel engines can also be used to both backup supply and supply power to remote communities. However the medium-speed engines are larger and more expensive than the high-speed engines, thus creating important economic decisions. The medium-speed engines are often used to provide power in industrial units that require their own power supply or losing the grid power is not an option. Medium-speed diesel engines can also be used in supporting the electricity grid, often in the case when the grid-connected renewable power is not available. (Breeze 2018)

Slow-speed diesel engines are the largest diesel engines and they can often be used as the base-load power. The slow-speed engines can also be used in the grid support role, if there is a fluctuating power demand. One advantage of the low-speed diesel engine is that their part load efficiency varies just slightly over the output range between 50% to 100% load. In comparison to gas and stream turbine outputs typically fall as the load decreases, when the low-speed diesel engine sustains relatively flat output over similar load range. The efficiency of the low-speed diesel engine can drop at the lower loads, but if this becomes an issue, a power plant can be built using several engines. When the load falls engines can be taken out of service as the load drops in order to sustain higher load for remaining engines. (Breeze 2018)

But for gas and diesel engines alike, both of these engines can be used in the automotive transportation. Different types of cars can utilize either the gas or the diesel engine in order to be efficient in their application.

In 2019, distillate fuel (primarily diesel fuel) consumption by the U.S transportation sectors was approximately 47.2 billion gallons (1.1 billion barrels). The amount accounts for 15% of the total U.S petroleum consumption and when looking at the energy content basis, for about 23% of the total energy consumption by the transportation sector. (U.S Energy Information Administration 2020)

Wärtsilä 31 diesel engine can be presented as an example engine. The 31 is a medium speed and 4-stroke engine, that is available in 8 to 16 cylinder configurations. The power output rating is between 4.7 to 9.8MW, at speeds of 720rpm and 750rpm. The 31 is said to have outstanding performance over the complete operating range. The applications for this engine are to work as the main propulsion engine, diesel electric configurations and auxiliary engine. (Wärtsilä 2020c)

For Wärtsilä gas engines, a Wärtsilä 31 SG can be used as an example. The 31 SG is a medium speed gas engine that can be used in a variety of ship types and as a main propulsion engine or as an axillary engine. In the offshore applications this gas engine can be used in for example in drilling. (Wärtsilä 2020d)

## 2.5 Common engine faults

According to Jian Chen’s doctoral thesis, most engine faults can be categorized into two categories. The two categories are combustion faults and mechanical faults. Misfires and engine knock are common combustion fault for engines. Engine knocks are uncontrolled self-ignition of the air and fuel mixture occurring in the middle of the combustion cycle, causing high combustion pressure spikes. Misfires are a result of spark plugs malfunctioning, loss of compression or faults in fuel injector. When measuring these faults, the most direct way to measure the combustion condition is to use a cylinder pressure sensor. Cylinder pressure sensors are expensive and prone to get damaged in the demanding combustion environment. Other methods to measure cylinder conditions are a translation acceleration signal measured on the engine block. Yet Another method is based on the torsional vibration signal of the crankshaft. (Chen, 2013)

Common mechanical faults include piston slap and bearing knock faults. These two faults are the end result of the oversized clearance that is caused by wear. For example under the conditions of an external shock load or contaminated lubricant, the metal to metal contact between the piston and the inner wall will lead to a high wear and quickly cause an oversized clearance. The oversized clearance further increases the impact forces and this leads to even greater wear. The bearing knock fault shares similar characteristics, but it is related to the wear of the journal bearings. Major wear mechanisms of the journal bearings are a gradual loss of material of the bearing lining in result of the mechanical contact with and sliding friction against the journal, erosion, abrasion and surface fatigue. (Chen, 2013)

### 2.5.1 Torsional vibration

According to Sun, Liming & Luo, Fuqiang & Shang, Tansu & Hen, Hongtao & Moro Adams 2017 research, torsional vibrations are usually used to describe torsional deformation movements of rotating shafts, as in internal combustion engine crankshaft. The causes of torsional vibrations can be divided into two categories, which are internal and external causes. These two causes are due two elastic deformations of the crankshafts body and the periodic effect of torques acting on the crankshaft respectively. In short, the crankshaft's torsional vibration can be defined as the elastic torsional deformations caused by the periodic acting excitation torques on the crankshaft during operation. (Sun, Liming & Luo, Fuqiang & Shang, Tansu & Hen, Hongtao & Moro Adams 2017)

For example diesel engines used in agricultular machinery have shown torsional vibration related problems. These problems result in engine vibration, crankshaft failure, undesirable engine noise and decreasing engine power. (Sun, Liming & Luo, Fuqiang & Shang, Tansu & Hen, Hongtao & Moro Adams 2017)

# 3 Digital signal processing

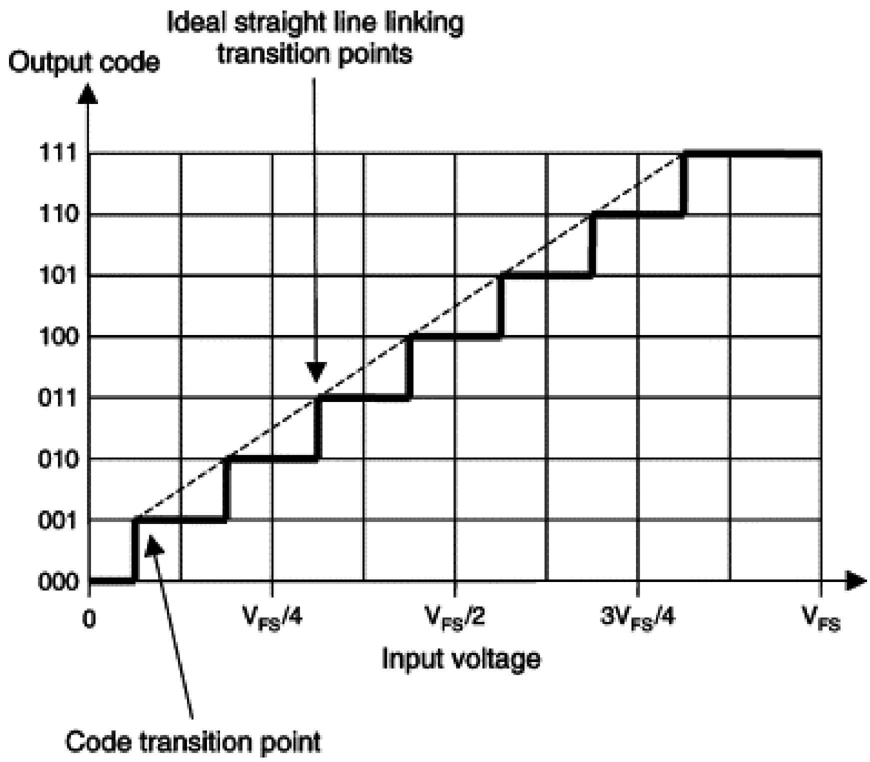
## 3.1 Digital signal processing

In digital signal processing, also known as DSP, the DSP processors take real world signals such as voice, video, temperature, pressure or position that have been digitized in analogue to digital converters. After taking these signals the DSP processors manipulate the signals and perform mathematical functions like addition, subtraction, multiply and division rapidly. (Analog 2020)

In order to process signals with digital signal processors, the analogue to digital conversion must be made. The analogue input as voltage or current represent the temperature, video, pressure or position. When the analogue signal enters the analogue to digital converter (ADC), The signal will be quantized. There will also be a quantization error that is caused by the conversion of an infinitely variable analogue input signal to a discrete level output signal. The quantization error is closely related to the analogue to digital converters resolution. The higher the resolution, or in other words the number of bits to be represented, the smaller the quantization error. (Grout 2008)

The analogue to digital conversion process can be considered to have two main operations, sampling and quantization. In sampling operation the analogue input signal is sampled at some fixed sampling rate. This process converts a continuous time signal into a discrete time signal. After sampling, the sampled signal is fed to a quantizer that produces the digital output. (Grout 2008)

When looking more closely to the quantization, a three bit ADC can be used as an example. Input voltage that ranges from 0V to +5V, three bits to represent the voltage values and an unsigned binary output code. The input signal conversion range is divided into $2^n$ values, where *n* is the ADC resolution. In a three bit converter, the voltage range is split into eight equal segments, as $2^3=8$. The value 5V would represent the maximum unsigned binary number $111_2$. Whereas the 0V would equal to $000_2$. For a comparison there could be an eight bit ADC that would have the voltage range split into $2^8$=256 segments. (Grout 2008)



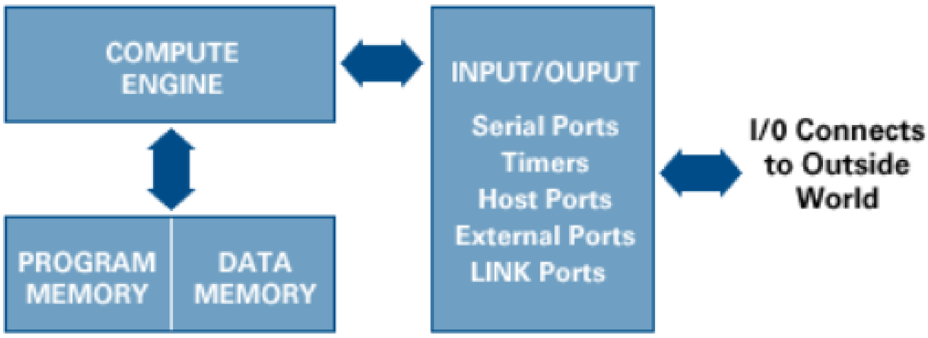

**Picture 2.** 3 bit quantization, $V_{FS}$=5V (full scale). (Grout 2008)

After the sampling and quantization processes, the digital signal processing starts. In DSP, the digitized information is then processed in the way described by the application. The information processed by the DSP can then be used to control things such as security, telephone home theater systems, video compression etc. Signals may also be enhanced or manipulated. This can improve the signals quality or provide information that human senses are not able to capture. Applications from enhancing and manipulating signals can for example be computer-enhanced medical images and echo cancellation for cell phones. Although signals can be processed in their analogue form, processing signals digitally enables the advantages of high speed and accuracy. (Analog 2020)

### 3.1.1 A typical DSP

A digital signal processing unit contains these important components:

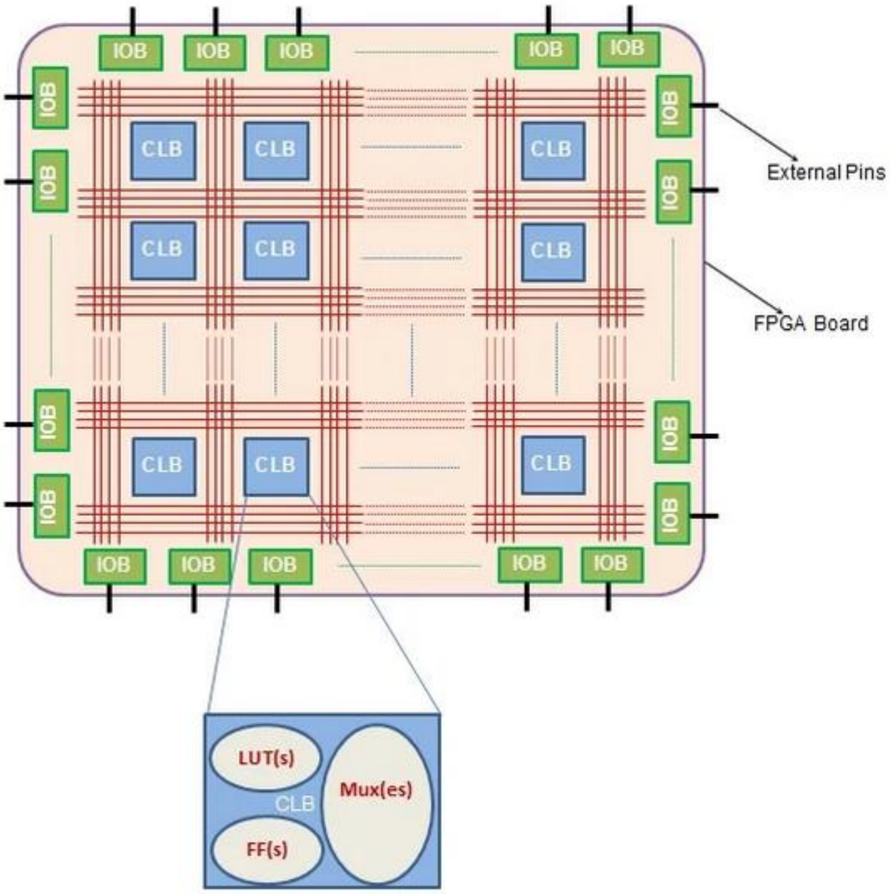
- Program memory: Stores the programs the DSP uses to process data.
- Data Memory: Stores the information to be processed.
- Compute Engine: Performs the mathematical processing and accessing the program from the program memory and the data from data memory.
- Input/Output: Serves functions to connect to the outside world. (Analog 2020)




**Picture 3.** DSP's key components. (Analog 2020)

## 3.2 FPGA

A FPGA (Field programmable gate array) is a semiconductor device composed of logic blocks, which are interconnected via programmable connections. These logic blocks consist of look-up tables, also known as LUT. LUTs have a fixed number of inputs and are constructed over simple memories, usually SRAM or Flash, which store Boolean functions. Each LUT is accompanied with a multiplexer and a flip- flop register in order to be able to work as a sequential circuit. Several LUTs can be combined to implement complex functions. (Tiete, J.& Dominguez, F. & da Silva, B. & Touhafi, A & Steenhaut, K. 2017)




**Picture 4.** Architecture of a typical FPGA where CLB stands for configurable logic block. (Sneha, H.L 2017)

The modern FPGAs are capable devices with support for myriad of I/O standards such as I2C, SPI, CAN, PCIe. The I/O in FPGAs are grouped in banks where each bank can independently support different I/O standards. (Tiete, J.& Dominguez, F. & da Silva, B. & Touhafi, A & Steenhaut, K. 2017)

FPGAs are reprogrammable to match desired application or functionality requirements. The hardware description elaborated by the designer is used by the vendor's synthesizer. This has to be done, in order to find an optimized setup of the FPGA's resources, that implement described functionality. This feature differentiates FPGAs from application-specific integrated circuits (ASICs), which are custom made for specific tasks. (Tiete, J.& Dominguez, F. & da Silva, B. & Touhafi, A & Steenhaut, K. 2017)

FPGAs have been originally used in network packet analysis and signal processing. Due to high-speed embedded resources such as DSP slices and fast memories, FPGAs are nowadays also utilized for algorithm acceleration either as standalone systems or coprocessors. Theoretically, there is no limitation in the calculation speed of an FPGA, since it uses parallel processing. Only the available FPGA resources determine the number of operations, that can be implemented. (Tiete, J.& Dominguez, F. & da Silva, B. & Touhafi, A & Steenhaut, K. 2017)

### 3.2.1 Benefits and disadvantages

FPGAs offer many benefits over other implementation methods such as ASICs and the off-shelf DSP and microcontroller chips. Here are some of the benefits of using FPGA instead of these alternative methods. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

Performance in FPGAs offer significantly enhanced computational speed compared to processor-based solutions. This is because of the FPGA's logic structure that incorporates parallelism in designs. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

Reliability of real-time systems is increased in FPGAs, due to the fact that FPGA-based designs consists of dedicated hardware for performing tasks with predictable delays. In processor-based designs, instructions are used to perform task using shared hardware resources. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

There are also benefits regarding the long-term maintenance of FPGA designs. FPGAs provide flexibility in upgrading the design in case of a change in the specification of an application over time. The time that is spent in redesigning or enhancing a FPGA-based design is greatly decreased compared to the time spent on redesigning or enhancing ASIC design. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

The cost of the non-recurring engineering (NRE) cost for designing a custom ASIC is enormous compared to FPGA-based solutions. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

Time to market for FPGA-based solutions is relatively quick. The FPGA technology provides flexibility for rapid prototyping of the design, by avoiding fabrication and other processing delays. (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018)

There is a one major limitation in FPGA-based designs compared to ASIC-based designs. The limitation is the overall performance, which is worse in the FPGA compared to ASIC (Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed 2018). Also the FPGA based digital signal processors (DSPs) are more expensive for large scale production than the ASIC designs. Depending from the application FPGA based solutions can consume more power than the off-the-shelf DSPs performing the same task. (Mendez, Arnaldo & Sawan, Mohamad 2015)

### 3.2.2 Applications

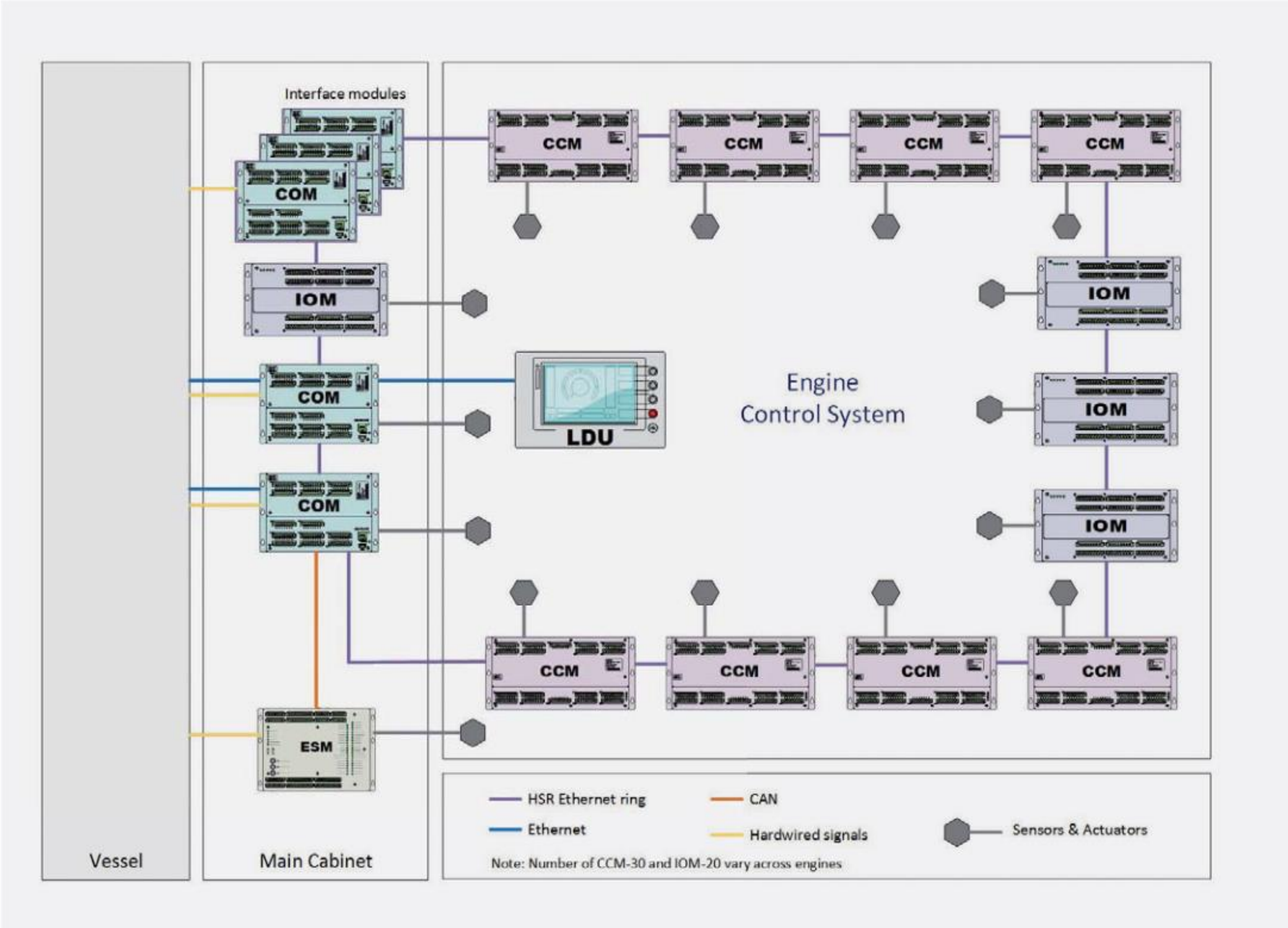
According to the Xilinx website, FPGAs can be used in several applications such as

- ASIC prototyping
- Video & image processing
- Medical diagnostics, monitoring and therapy applications
- Flexible industrial applications
- Automotive applications, for example driver assistance, comfort, convenience and in-vehicle infotainment
- Data center: High bandwidth and low latency servers, networking and storage applications
- Solutions for network attached storage (NAS) and storage area network (SAN)

- Applications for wired/wireless communications
- Aerospace applications (Xilinx 2020)

## 3.3 Wärtsilä engine control system, UNIC

The Wärtsilä UNIC is an embedded engine control system for Wärtsilä 4-stroke engines. The UNIC-system has been in development for decades, and it makes engines safe, environmentally energy efficient, reliable and flexible. The main tasks of the UNIC-system are to protect the machinery and ensure the safety of the engine room crew. The system also allows the smart engine to produce high amount of power with less fuel. UNIC-system is also designed for various ambient conditions, and it adapts to different fuels and qualities. (Wärtsilä 2017)




**Picture 5.** Overview of the Wärtsilä UNIC system. (Wärtsilä 2017)

The UNIC-system is designed to be directly mounted on the engine, thus it does not require much space and does not require any mounting components in dispersed external cabinets or panels. The modular design of the system allows easy access when installing the system or when doing service. Some pats and functions in the UNIC-system are optional, depending on the engine and the installation requirements. (Wärtsilä 2017)

### 3.3.1 Modules

The Wärtsilä UNIC-system consists of different modules that communicate with each other. Here is a table describing all the modules.

**Table 1.** The modules in the UNIC-system. (Wärtsilä 2017d)

| | |
|---|---|
| LOP-Local operator panel | Local operator panel consists of a display unit and an emergency stop button. The display unit has pushbuttons and touchscreen, in order to allow the operator to operate the engine and display engine information. |
| COM-Communication module | The communication module is the main gateway to the UNIC-system from vessel systems. It supports multiple interfaces such as Modbus, OPC, hardwired I/O, etc. COM is a key module for UNIC system communication and responsible for several control functions. It is also responsible for managing the configuration and software updates. Two COM modules are typically used in the UNIC-system for redundancy. |
| CCM-cylinder control module | The cylinder control module is mainly responsible for combustion control. CCM monitors and controls all the injection and combustion functions, and the inlet valve timing for the cylinders. The number of modules varies according to the number of cylinders. The CCM is typically located on the side profiles of the engine, inside the Wärtsilä terminal boxes (WTB) |
| IOM-Input/Output module | The input/output module is responsible for handling all measurements in certain areas of the engine. These modules are |

| | |
|---|---|
| | placed close to sensors and measurable devices. The number of IOM modules varies according to the number of cylinders, engine type and the application. IOM is typically located on the free or driving ends of the engine, and also enclosed in WTB. |
| ESM-Engine safety module | Engine safety module takes care of functions related to the safety of the crew in case of failures related to the engine. ESM provides safety functions like shut down due to over speed or low lubricating oil pressure etc. ESM is most often located in the main cabinet of the engine. |

### 3.3.2 Enabling engine performance

Due to several factors, like tightening emission legislation, fuel consumption, variations in fuel quality and overall performance, second generation UNIC closed-loop controls have been optimised to achieve fast-acting and robust control dynamics. The main aspects in optimizing these controls were:

- Real-time control modules
- Sensors
- Measurement signal sampling
- Communication network
- Control algorithms
- Actuator control (Vuollet 2017)

All of the above mentioned aspects need to be designed with precision, in order to ensure high quality control. It is also important to ensure that the different control modules of the system are synchronised, to maximize their functionality. Efficient communication between control modules enables to fulfill strict timing requirements, and to make sure that control algorithms are provided with valid measurement data. (Vuollet 2017)

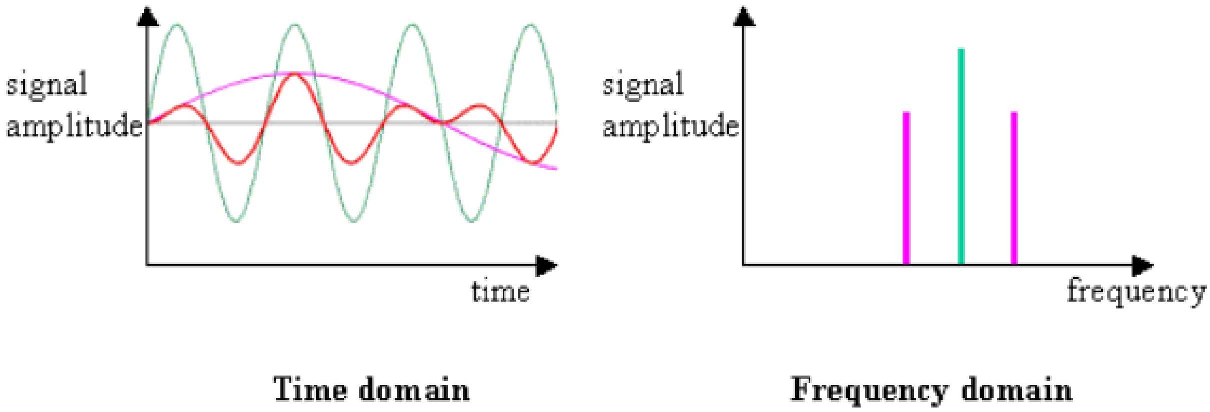
# 4 Frequency analysis

## 4.1 Frequency Domain

The Frequency domain refers to the analytic space in which mathematical functions or signals are conveyed in terms of frequency, rather than time. For example, a time domain graph may display changes over time, whereas a frequency domain graph displays how much the signal is present among each given frequency band. (Deep AI 2020)

It is possible to convert the information from a time domain to frequency domain. One way to do this transformation is a Fourier transform. The Fourier transform named after Joseph Fourier, converts the time function into a set of sine waves that represent different frequencies. The frequency domain signal representation of a signal is known as the "spectrum" of frequency components. (Deep AI 2020)

The frequency domain functions by allowing a representation of the qualitative behavior of a system, as well as characteristics of the way the system response to changes in bandwith, gain, phase shift, harmonics, etc. One area where frequency domain is often used for graphical representation is music. Audio producers and engineers can display an audio signal within a frequency domain, in order to obtain a better understanding of the shape and character of an audio signal. (Deep AI 2020)




**Picture 6.** Same signal represented in time- and frequency domain. (Deep AI 2020)

### 4.1.1 Fourier transform

The Fourier transform is capable of decomposing a complicated waveform into a sequence of simpler elemental waves (in more depth, a weighted sum of sines and cosines). This is analogous to how a wave representing a music chord can be expressed in terms of the properties of its base notes. If these notes would be graphed via Fourier transform on a frequency versus intensity graph, there would be visible peaks corresponding to these music notes. An original function and its transformed pair are collectively known as Fourier pairs. (Deep AI 2020b)

One of the most common uses of the Fourier transform is to find the frequency range of a signal that changes over time. This form of signal processing is used in many fields, for example in cryptography, oceanography, speech patterns, communications and image recognition. The Fourier transform can also be a useful technique, when solving differential equations. (Deep AI 2020b)

The frequency content of a discrete signal can be analysed with Discrete Fourier Transform (DFT). DFT transforms a finite discrete function into another discrete function. In short, DFT transforms a time domain signal into a frequency domain series (Brilliant.org 2020)

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi ikn/N}, \qquad \text{for } 0 \leq \text{k} \leq \text{N} - 1$$

The above formula represents the Discrete Fourier Transform (Brilliant.org 2020). Another version of the Fourier transform is the short-time Fourier transform (STFT), which can be utilized to calculate the Fourier transform of time windows of longer signals. The STFT of a time signal x(t) over a time window w(t) with a length of T can be defined as follows (Zhenuy 2015).

$$STFT(t, f) = \int_{t-T/2}^{t+T/2} x\,(\tau) w(\tau \; - \; t) e^{-j2\pi f\tau}\, d\tau.$$

There is also a Fast Fourier transform, which computes the Discrete Fourier Transform in quicker way. It computes the Fourier transformations by factoring the DFT matrix into a product of factors. (Deep AI 2020b)

The Fast Fourier Transform reduces the complexity of DFT from $O(N^2)$ to O(NlogN). This is a great difference in speed if processed datasets are in size range of thousands or even millions. (Deep AI 2020b)

#### 4.1.1.1 Windowing

When using FFT to measure the frequency component of a signal, the analysis is based on a finite set of data. The FFT assumes that it is a finite data set, a continuous spectrum that is one period of a periodic signal. When the measured signal is periodic and an integer numbers of periods fill the acquisition time interval, the FFT comes out fine. But however, usually the measured signal is not a integer number of periods and therefore the finiteness of the measured signal results in a truncated waveform with different characteristics from the original continuous-time signal. The finiteness can introduce sharp transition changes into the measured signal and these sharp transitions are called discontinuities. (National instruments)

When the number of periods in the acquired signal is not an integer, the endpoints are discontinuous. These artificial discontinuities are visible in the FFT as high-frequency components, that are actually not present in the original signal. These frequencies can possibly be higher than the Nyquist frequency, and they are aliased between 0 and half of the sampling rate. The spectrum obtained by using the FFT is therefore not the actual spectrum *f* the original signal, it is a smeared version of it. It appears as if energy from one frequency leaks into other frequencies. This phenomenon is known as the spectral leakage, and it causes the fine spectral lines to spread into wider signals. (National instruments)

This effect can be minimized by using windowing, which reduces the amplitude of the discontinuities at the boundaries of each finite sequence acquired by the digitizer. Windowing consists of multiplying the time records by a finite-length window with an amplitude that varies smoothly and gradually towards zero at the edges. This causes the endpoints of the waveform meet, and results in a continuous waveform without sharp transitions. (National Instruments)

There are several different windowing functions that can be applied depending on the type of the signal. Each of the different windowing functions have their own characteristics and suitability for different applications. In general though, the Hanning window is satisfactory in 95 percent of the cases and it should be used if the nature of the signal is not entirely clear, but smoothing of it is required. (National Instruments)

## 4.2 Mechanical waves

A mechanical wave is a disturbance that travels through some substance or material called the medium for the wave. As the wave travels through the medium, the particles forming the medium undergo displacements of various kinds, depending on the nature of the wave. (Young & Freedman 2012: 473)

The two types of the mechanical waves are a transverse wave and a longitudinal wave. In transverse wave the displacements of the medium are perpendicular or transverse to the direction of travel of the wave along the medium. A simple example of a transverse wave would be when shaking a string up and down, the wave created is transverse wave. (Young & Freedman 2012: 473)

In the longitudinal wave, motions of the particles of the medium are back and forth along the same direction that wave travels. For example, the movement of a piston inside a rigid container filled with liquid can present longitudinal waves. The longitudinal wave moves in the liquid, and each particle of the liquid moves forward and then back, parallel to the motion of the wave. (Young & Freedman 2012: 473)

## 4.3 Fundamental frequency and Harmonics

When an object is forced into resonance vibrations at one of its natural frequencies, it vibrates in such a way that a standing wave pattern is formed within the object. Whether this object is a guitar string or some other object, the vibrating medium vibrates so that a standing wave pattern results. (the Physics Classroom 2020)

Each natural frequency that an object or instrument produces has its own characteristic vibrational mode or standing wave pattern. These patterns are only created within the object or instrument at specific frequencies of vibration. These frequencies are known as harmonic frequencies, or harmonics. (the Physics Classroom 2020)

At any frequency other than a harmonic frequency, the resulting disturbance of the medium is irregular and non-repeating. For objects that vibrate in a regular and periodic way, the harmonic frequencies are related to each other by whole number ratios. (the Physics Classroom 2020)

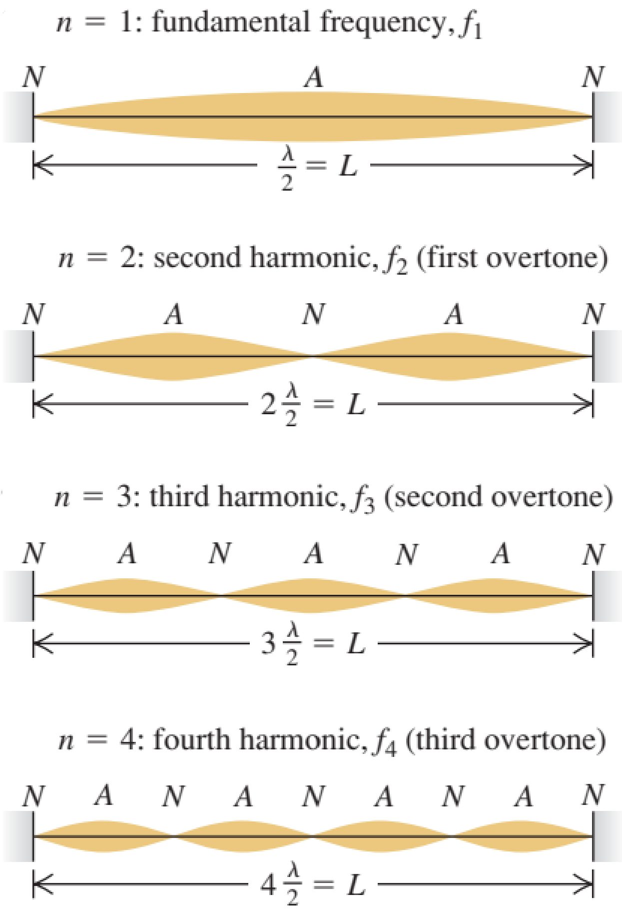
When considering a guitar string vibrating at its natural frequency or harmonic frequency, and both of the ends of the string are attached and fixed in place to the guitar's structure which leads to the ends of the string being unable to move. As a result, these ends become nodes (points of no displacement). In between these two nodes, at the end of the string, there must be at least one antinode. The most fundamental harmonic for a guitar string is the harmonic associated with a standing wave having only one antinode positioned between the two nodes on the end of the string. This would be the harmonic with the longest wavelength and the lowest frequency. The lowest frequency produced is known as the fundamental frequency, which can also be called as the first harmonic. (the Physics Classroom 2020)

For a string that is fixed from the both ends, the fundamental frequency can be expressed in a mathematical formula. (Young & Freedman 2012: 496)

$$f_1 = \frac{v}{2L}$$

The other standing-wave frequencies are all integer multiples of the fundamental frequency $f_1$. There can for example be $2f_1$, $3f_1$, $4f_1$ and so forth. All of these frequencies can be expressed with following formula. (Young & Freedman 2012: 496)

$$f_n = n\frac{v}{2L} = nf_1 \quad (n = 1,2,3 \ldots)$$




**Picture 7.** Presentation of harmonics from fundamental frequency $f_1$ to fourth harmonic $f_4$ (Young & Freedman 2012: 496)

### 4.3.1 Complex standing waves

If string could be displaced so that its shape is the same as one of the normal-mode patterns and then release it, the string would vibrate with the frequency of that mode. Such a vibrating string would displace the surrounding air with the same frequency, producing a traveling sinusoidal wave that ears would perceive as a pure tone. But when a string is struck or plucked, the shape of the displaced string is not as simple as one of the patterns in the Picture 5 (above Picture). The fundamental as well as many harmonics are present in the resulting vibration. This motion is therefore a combination or superposition of many normal modes. Several simple harmonic motions of different frequencies are present simultaneously, and the displacement of any point on the string is the sum, or in other word superposition, of the displacements associated with the individual modes. (Young & Freedman 2012: 497)

The sound produced by the vibrating string is likewise a superposition of the traveling sinusoidal waves, which one perceives as a rich and complex tone, with the fundamental frequency $f_1$. The standing wave on the string and the traveling sound wave in the air have similar harmonic content, which depends on how the string is initially set into motion. (Young & Freedman 2012: 497)

It is possible to represent every possible motion of the string as some superposition of normal-mode motions. Finding this representation for a given vibration pattern is called harmonic analysis. The sum of sinusoidal functions that represents a complex wave is called a Fourier series. (Young & Freedman 2012: 497)

### 4.3.2 Vibration and sound

When the human ear hears a sound, it is caused by the vibrations of the air molecules. In order to produce a sound, the source object has to vibrate. This causes the air molecules to vibrate and collision to their neighbouring molecules. The collision cause a progression of in the air, called as the sound wave. (Scienceworld 2021)

### 4.3.3 Nyquist Frequency

The Nyquist-Shannon sampling theorem explains that a signal sampled at a rate F can be entirely reconstructed if it only contains frequency components below half of that sampling frequency F/2. The F/2 is called as the Nyquist frequency. If a component of the signal is higher than the Nyquist frequency, a sampling error called aliasing occurs. (Gatan)

# 5 Concept method design

## 5.1 Software used

### 5.1.1 MATLAB

MATLAB is a programming language designed for engineering and science. MATLAB is a matrix-based language, which allows natural expression of computational mathematics. MATLAB combines a desktop environment that is tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly. (MathWorks 2020)

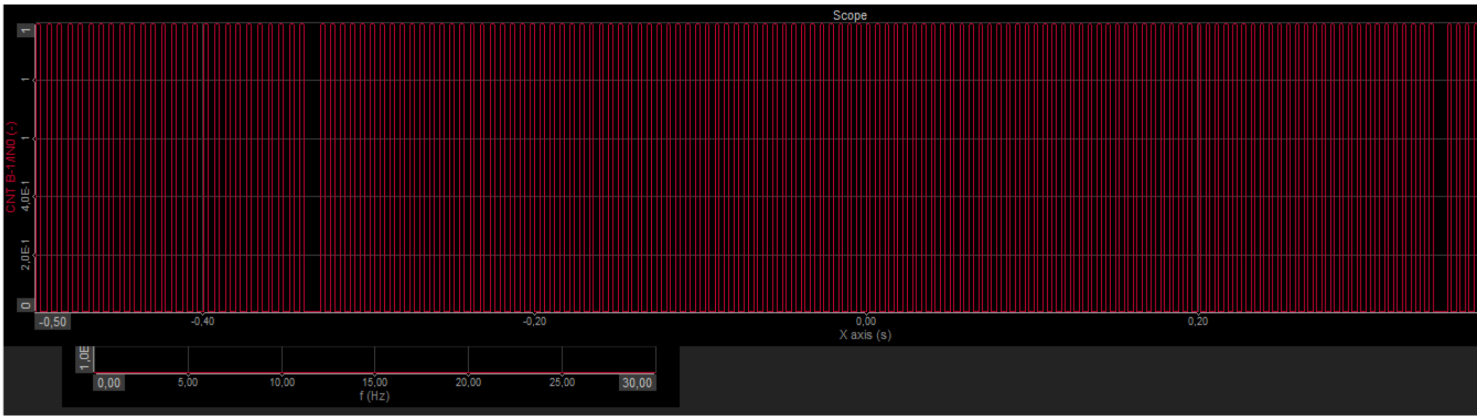
The main use cases for MATLAB are to analyse data, develop algorithms and create models and applications. MATLAB can be used in a range of applications, for example deep learning, machine learning, signal processing, communications, image and video processing, control systems, testing and measurement, computational finance and computational biology. (MathWorks 2020)

### 5.1.2 Dewesoft X3

The Dewesoft X3 is a data acquisition software, used to record, visualize and analyse data. The software can be used in applications such as order tracking analysis, rotational and torsional vibration analysis, FFT spectrum analysis and frequency analysis.

The base data for the concept method is the engine flywheel's speed signal. The method should be able to do the short-time Fourier transform, or possibly the fast Fourier transform for the speed signal, and extract harmonics from the signal and also know how to calculate the most important harmonics in a way that they give meaningful information about the engine's current state.

## 5.2 Speed signal

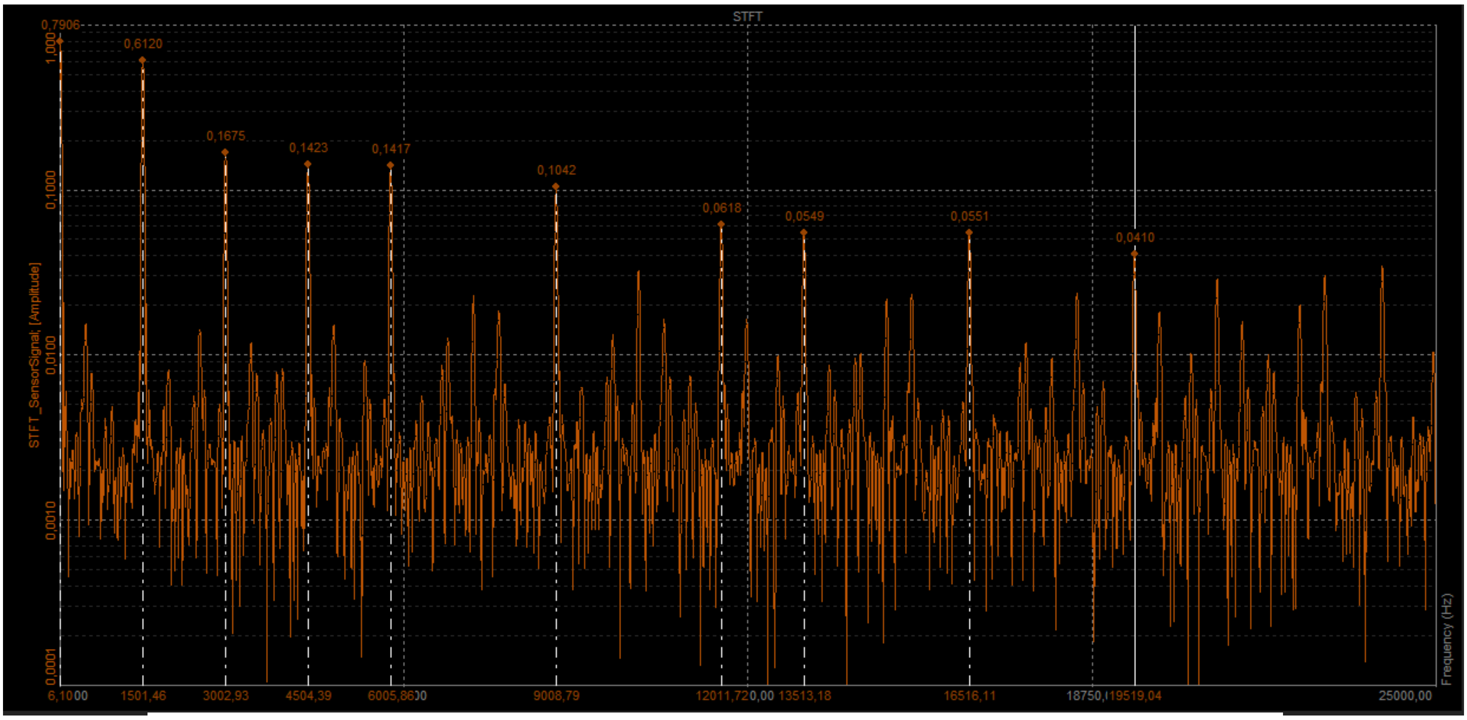



**Picture 8.** Plot from the flywheel's speed signal analyzed in Dewesoft X3. Speed signal is from 20V31SG engine.

In order to measure engine speed and the harmonics that hold information from the engine's processes, the speed signal is needed. The above plot shows a part of the flywheel's speed signal against time X (s), and it can be seen as the base data for this thesis. It is taken from a real test engine, that is ramping from 0 to 750 rpm. The signal consists of the flywheel's 120 teeth hitting a sensor, which show as the red square waves in the plot. There is also a one missing tooth that can be seen as the gap in the different parts of the plot.

The flywheel's speed signal can be used as a base for different measurements from the engine. As stated in the Kim Lewis's article the frequency in which the engine is rotating can be converted into rpm when multiplied with 60rpm (Lewis 2017). This can be explained by an example that the above speed signal's frequency is at some point 12.5 Hz, corresponding to 12.5 revolutions per second. In order to change this to revolutions per minute (rpm). The frequency is multiplied with 60. The conversion is 12.5Hz*60=750rpm.

The frequency can also be extracted directly from the short-time Fourier transformation of the speed signal. The STFT of the speed signal shows the different harmonics related to this signal, and the speed is one of these harmonics.




**Picture 9.** A short-time Fourier transformed waveform showing a part of engine’s speed signal, analyzed in Dewesoft X3.

In the picture 8, the engine’s speed signal has been short-time Fourier transformed to exhibit the frequency domain of the signal. The STFT utilizes Hanning window and overlap of 75% to reduce discontinuities and make the transformation more accurate. The sample rate of the signal is 50kHz and thus the graph is set to be half of that at 25kHz, in order to be equivalent to the Nyquist frequency of this signal.

The speed of the signal can be seen from the frequency axis at point 1501.46Hz, and it has amplitude of 0.6120. In order to get the speed in revolutions per minute, the frequency has to be first multiplied with 120 (teeth) and then with 60 to turn seconds into minutes. The division by 120 has to be done because the 1501.46Hz is representing the frequency of only one tooth. After the division the frequency appears as the frequency of the whole flywheel. Then the frequency is multiplied with 60 to move from 1/s to rpm. When doing the calculations, the rpm is 1501.46Hz/120teeth*60= 750,73rpm.

The speed is a great example of how the harmonics can be utilized and gain information from. The other harmonics should also be known and calculated in order to gain as much knowledge as possible from the speed signal. The concept method should be able to extract the speed signal and the other harmonics as well.

## 5.3 Engine harmonics

When looking more closely to the engine and the harmonics, previous studies have shown that when in an ideal state where all engine cylinders are contributing uniformly to the total engine torque. The first three harmonic orders (K= 0.5, 1, 1.5) have a significant role in the frequency spectrum of the total gas-pressure torque, which leads to them appearing with a low contribution in the frequency spectrum of the crankshaft's speed. (Gawande, S.H & Navale, L.G & Nandgaonkar, M.R & Butala, D.S & Kunamalla, S 2012)

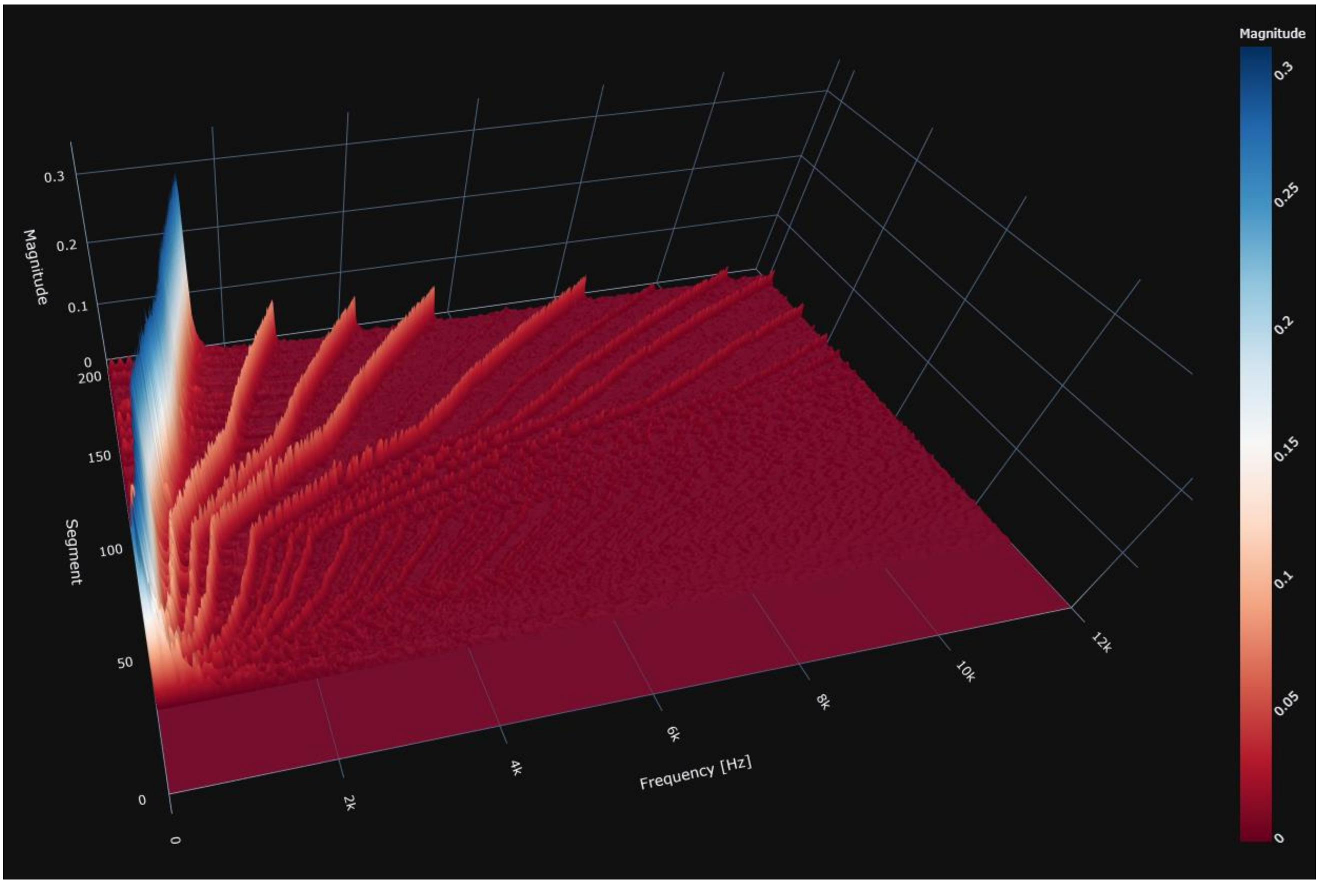
If the frequency spectrum of the crankshaft's speed corresponding to uniform cylinders operation is compared to the spectrum which corresponds to a faulty cylinder, one can see that the major difference is produced by the amplitudes of the first three harmonic orders. As long as the cylinders operate uniformly, the amplitudes are being maintained under a certain limit. Once a cylinder starts to decrease its contribution, the amplitudes of the first three harmonics orders start to increase. It is also possible to use these amplitudes to determine the degree by which a cylinder decreases its contribution to the total gas pressure torque. The identification of the faulty cylinder may be achieved by analysing the phases of these three first harmonics. (Gawande, S.H & Navale, L.G & Nandgaonkar, M.R & Butala, D.S & Kunamalla, S 2012)

Torsional vibrations can also cause harmonics in an engine. As stated in the Dr. -Ing. Dejan Arsic (2016) article, torsional vibrations, which is simply angular vibrations of rotating object in the powertrain, can often lead to failures at couplings and rotating shafts if not controlled. Ideally this torque should be smooth and only applied to the rotational part. But in reality this is not the case and it has an effect on the component on the plane. Rotating systems also create periodically repeated phenomena, which usually happens only in certain positions of the system. For example a defective gear or an explosions in the cylinder during combustion process (Dr. -Ing. Dejan Arsic 2016)

It is also worth noting that mechanical wear of the moving parts in the engine, for example crankshaft, pistons and other parts, can cause deformations and lead to causing a different kind of a frequency spectrum compared to normal good condition operation.

### 5.3.1 Waterfall plot

Here is an waterfall plot showing the harmonics of an Wärtsilä engine, that is ramping up from 0 to 750 rpm. (Jonas Mäntylä, personal conversation, 21.9.2020)

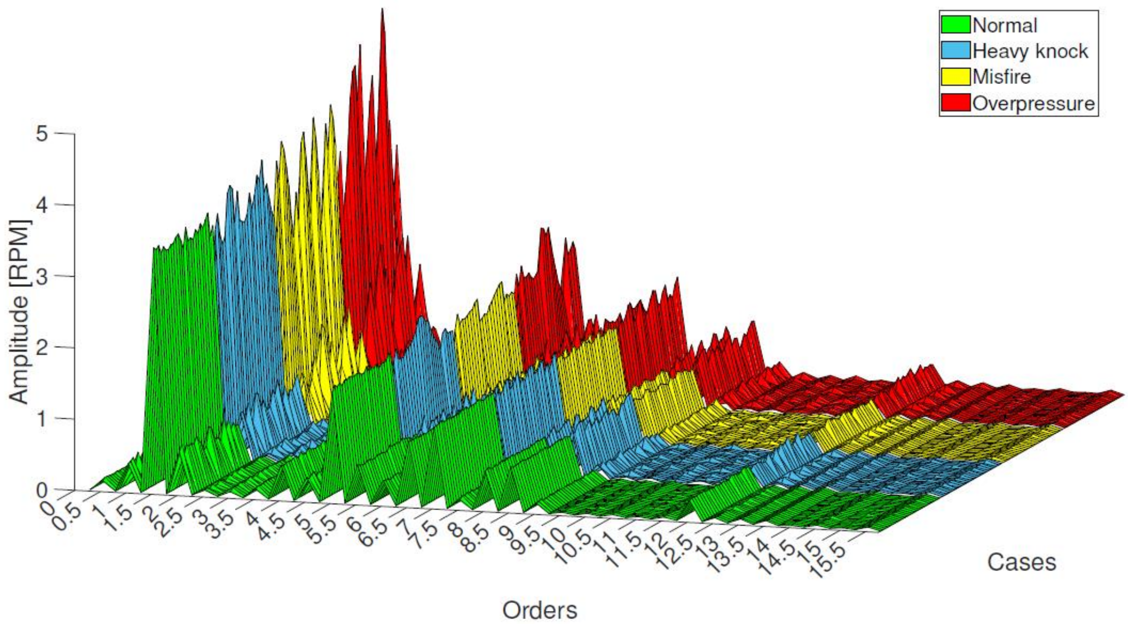



**Picture 10.** Waterfall plot showing the harmonics of an engine ramping up from 0 to 750 rpm. (Jonas Mäntylä, personal conversation, 21.9.2020)

In the Picture 6, the blue ridge corresponds to the flywheel tooth frequency, and the other ridges are the harmonics of the tooth frequency. Magnitude axis explains the magnitude of the frequency. The segment axis represents the output of the Fourier transformed portion of the speed signal samples (time windows). A segment is samples X to Y of the speed signal in time domain transformed to frequency domain. Here the X is a sample number of 1024 and Y is number 1536. This causes the number of samples to be 1536-1024=512. The speed signal itself is sampled with a sampling frequency $F_S$ of 10kHz. The time window for segment is then 512*(1/10000) = 0.0512s. A segment represents the frequency components for 0.0512s of the speed signal. (Jonas Mäntylä, personal conversation, 21.9.2020)

### 5.3.2 Fault detection

In Palestini Cesare's study for the 20V31SG engine, he found that amplitudes can be used to examine if there is some type of fault happening in the engine. The study was done with a help of simulation and the faulty conditions were simulated. Palestini Cesare says that in normal conditions the speed has a consistent through the whole domain, and when fault occur it can be detected from speed's behaviour. When the orders of a speed signal were analyzed the resulting graph looked like following. (Palestini 2018)




**Picture 11.** A graph showing the behaviour of the speed signal's orders. $0^{th}$ order is the speed and it is off the chart. (Palestini 2018)

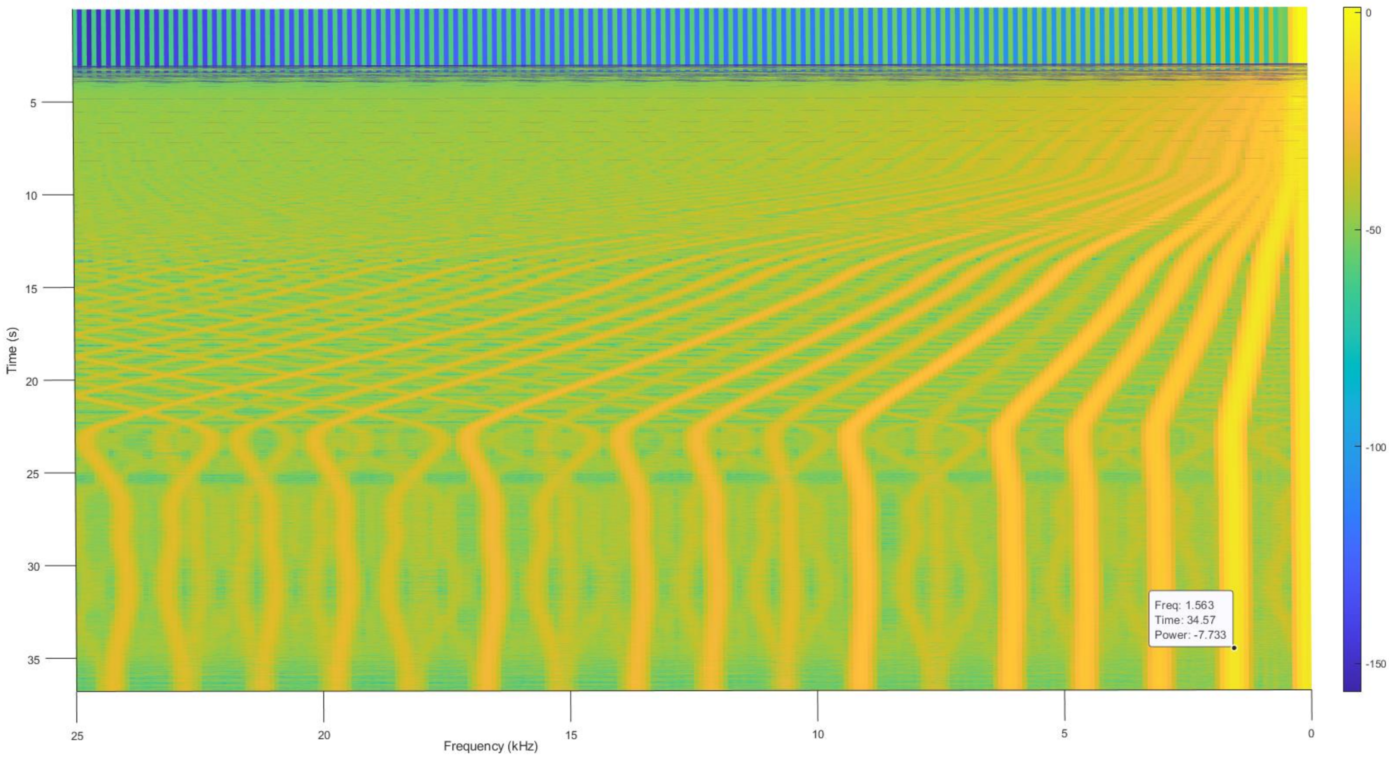
In normal conditions the green graph is relatively consistent through the analysis. When looking at the graphs of different colors however, the amplitudes start to show greater amounts of variance. This can show for example if the engine is misfiring or there is overpressure. (Palestini 2018)

So the behaviour of the harmonics can explain if the engine is not running correctly. This can accordingly be seen from the consistency of the order graph in picture 9. Possibly, in some situations the differences in the variances in the amplitudes of the orders between a faulty engine and a good one can be small if the fault is not yet great enough to cause harm.

In a personal conversation with Palestini Cesare, he explained that some of the important harmonic orders are firing frequency, which corresponds to the order 0.5. The engine bank frequency, can be calculated by using the harmonic 0.5 and multiplying it with number of cylinders in the bank. He also said that in the above graph the harmonic order at 1.5 is predominant, because it is matching the natural frequency of the coupling. (Palestini Cesare, personal conversation 4.11.2020)

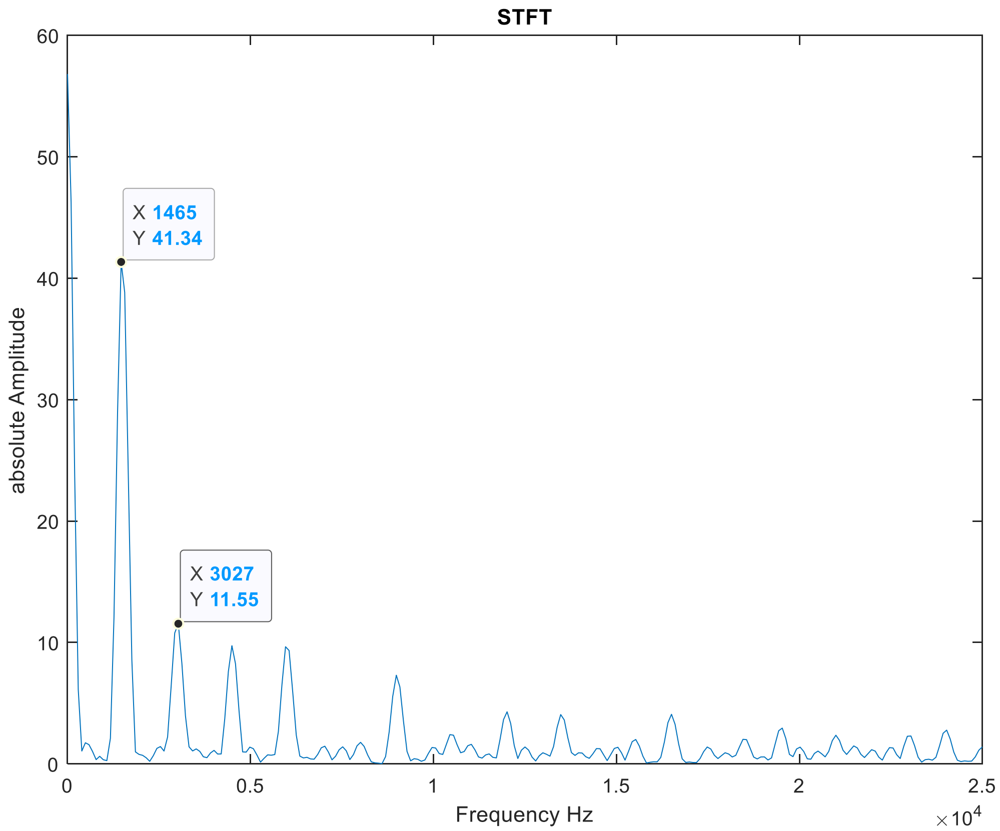
## 5.4 Calculation of the harmonics

One way to calculate the harmonics could be to first calculate the short-time Fourier transform from the speed signal in MATLAB, using the stft(), pspectrum() or spectrogram() functions. After that the findpeaks() function could be used to extract the harmonics from the Fourier transformed data.




**Picture 12.** Spectrogram of the speed signal after the MATLAB's spectrogram function.

In the above picture. The MATLAB's spectrogram() function is used to calculate the short-time Fourier transform of the whole speed signal. The signal can be divided to different window sizes, overlap can be adjusted and also other settings can be modified. The spectrogram shows the different harmonics, which are displayed as the yellow lines going forward in time. The graph also shows the power of the signal. There are also other ways to calculate the STFT in MATLAB, such as the stft() and pspectrum functions.




**Picture 13.** One short-time Fourier transformed part of the speed signal. Y-axis is the absolute amplitude and x-axis is the frequency in Hz.

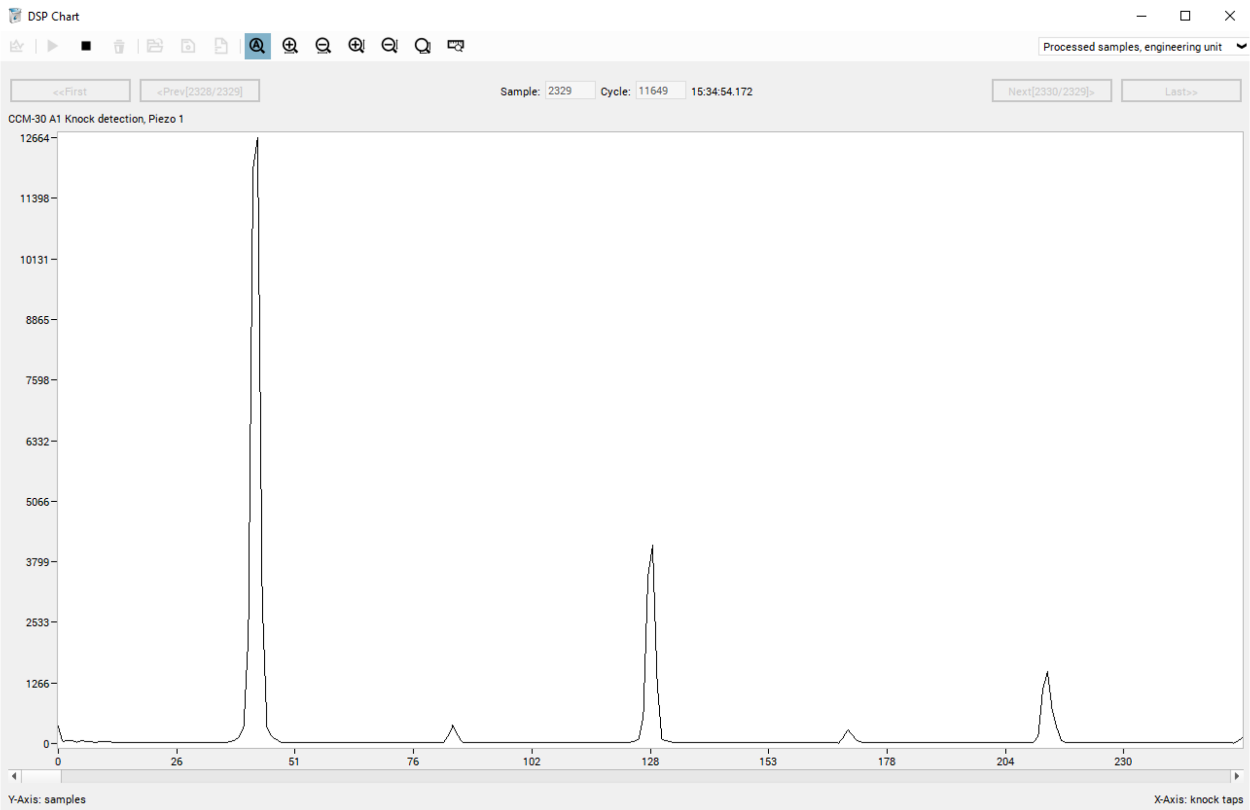
In the above graph showing one short-time Fourier transformed part of the speed signal, the harmonic peaks are clearly visible.  For example in the graph there are two points of interest at Positions X 1465 Hz and X 3027 Hz. These two positions show the two first harmonics in the above graph, and a great way to extract this kind of information from short-time Fourier transformed signals is to use MATLAB's findpeaks() function. After obtaining the harmonics from the findpeaks() function, the data can be used to for example calculate the speed or more generally, to get information related to each harmonic.

# 6 Implementation

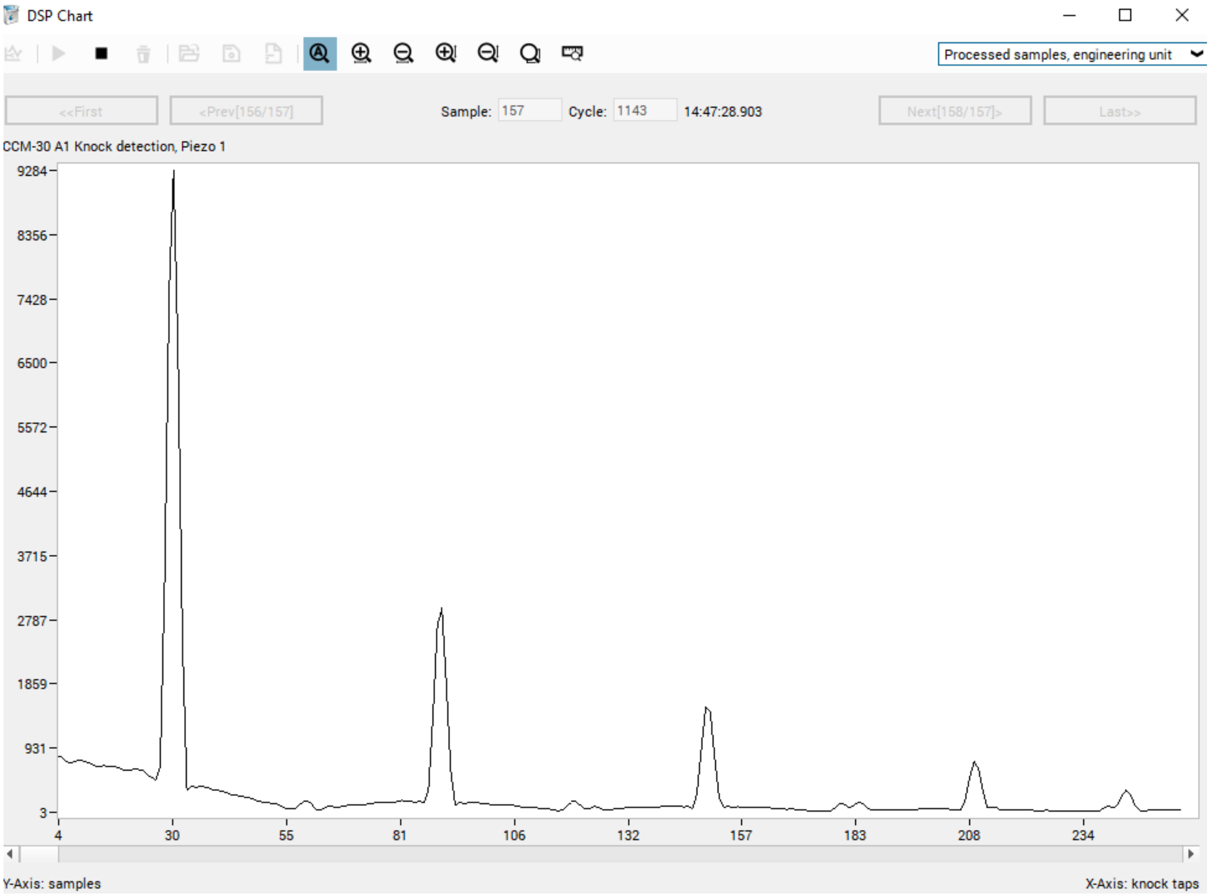
## 6.1 Minirig test

The CCM-30 modules Piezo channel was tested to see how the ready made Fast fourier transform works. A Wärtsilä speed simulator was used and configured to mode 120-1 (120 tooth, where one is the missing tooth). The frequency range of the simulator was 10-4000Hz.

The speed simulator was then attached to a minirig. Minirig consist of multiple Wärtsilä UNIC modules for simulation and testing purposes. After configuring the minirig, the speed simulator was installed to one of the CCM-30 piezo channels that is used for knock detection. The resulting FFT's were the following graphs.




**Picture 14.** CCM-30's piezo channels FFT.




**Picture 15.** CCM-30's piezo channels FFT.

In both of these graphs, the y-axis stands for samples and the x-axis for knock taps. Because this is the piezo channel there is a calculation to be done before the frequency can be extracted. The formula for this equation is the following.

$$Knock\_frequency = KNOCK\_TAP * (sample\_frequency/fft\_length).$$

In the above formula the KNOCK_TAP represents the x-axis value, sample_frequency is a hard coded 25KHz for the piezo channel and the fft_length is configured to be 512.

For an example, if the KNOCK_TAP is 30, like in the above graph. The Knock_frequency is 30*(25KHz/512)=1464,8Hz. This can be turned into rpm by multiplying with 60 and dividing by 120 teeth. 1464,8Hz*60/120=732rpm.

To be sure that the speed simulators frequency and the FFT match each other, it was tested that when the simulators speed was set visually to 50%. The result for the KNOCK_TAP was 42, which resulted the Knock_frequency to be 2050Hz. There can be some error from the 50%, but the test was done visually, which can lead to some minor inaccuracy. On the other hand, the bin size of this FFT was 25000Hz/512. This is roughly 49Hz. Which corresponds to a frequency step of 49Hz, and this would lead the 2050Hz be actually just one bin over the 50%. But still this result is acceptable and close enough for further use.

Another remark about the above FFT graphs can be made though. The Fast fourier transform seems to be skipping every second harmonic. This can be a result of the piezo channels configuration or because of the fact that the speed signal is just simulated and does not fully correspond the speed signal of the real engine.

## 6.2 Setup

In order to start developing the concept method, the correct repository had to be cloned to my PC using GIT. After cloning and making sure the repository is installed correctly with submodule updates and dependencies, the IDE environment had to be setup. For this work, the Eclipse IDE was used. The cloned repository was imported to the Eclipse workspace. After the import, the build configuration and setting needed to be configured, and then the project was ready to be built for the first time.

## 6.3 Concept method

The MATLAB method, developed in the design part, can be used as a base for the UNIC based method. In the MATLAB method the tachometer signal is processed in a short-time Fourier transform function, because of a large speed signal data that needs to be separated into different parts. In the method done for UNIC the fast Fourier transform will be used, because of the continuous input of the speed data.

In the minirig test, the CCM-30 piezo channel had the functioning FFT and that will be used as a base for the method done for UNIC hardware. To show the speed and other harmonics as values in the UNITOOL, code changes to the module are required to complete the concept method.

The FPGA collects the data from the piezo channel, and then it passes that data to the CPU where the fast Fourier transform and other calculations, such as the concept method, are done. The implementation of the concept method is done with embedded C, and is can be tested with a minirig and with the Wärtsilä Unitool software.

### 6.3.1 Programming

After setting up the environment and finding the correct place for the concept method, the programming of it can be started. The pictures shown in the chapter 6.1 minirig test show the DSP snapshot functionality, which is closely related to this method.

There is a for-loop, which handels the FFT data and stores it to the graph. It can be utilized to store and alter the data for the purpose of calculating speed and other harmonics.

```
for (i = 0; i < DSP_NUM_OF_WINDOW_COEFF / 2; i++)
```

The loop goes through half of the data, which can for example be 512 samples long. Referring to the minirig test chapter, the i in this for loop stands for the knock taps, which are helpful in calculating the wanted results.

```
tmp = DSP_GET_UPPER_BYTES_FROM_U32(fftBuff[i]);
img = DSP_GET_S16_FROM_U16(tmp);
tmp = DSP_GET_LOWER_BYTES_FROM_U32(fftBuff[i]);
real = DSP_GET_S16_FROM_U16(tmp);
```

In the above part of the program, the upper and lower bytes of the FFT buffer are stored into two variables, in order to get the imaginary and real parts of the data.

```
DSPSnapshotInfo.pDSPSnapshotRt->data[i] = hypotf(img, real);
```

Then the hypothenuse of the imaginary and real part is stored and this causes them to be displayed in the pictures seen in the minirig test chapter. Important thing to notice is that the calculation of the hypotenuse results in getting the amplitude of the imaginary and real parts of the signal.

In order to get the speed and harmonics of the data, the above code is used as a base for the concept method. A new for loop is created to go through the already stored data, and perform peak detection to it. But first, there was a need of having variables that can be used to show the stored values in Unitool. The fastest way to get values like this was to use already existing dummy variables found the same project. The names of the values do not correspond perfectly to the information that is stored into them, but they still function perfectly. To change the names of these values would have been a complicated task, because there are multiple locations that would require the name to be changed.

```
for (i = 0; i < DSP_NUM_OF_WINDOW_COEFF / 2; i++)
```

First There are two basic variables, integer shortCounter and boolean firstPeak, that are helpful in order to keep track what is happening in the for-loop.

```
if(shortCounter<4&&DSPSnapshotInfo.pDSPSnap-
shotRt->data[i]>1500)
```

The first if sentence checks if the short counter is below 4, because there is room for 4 values in the array. The other condition is for making a certain threshold that only allows amplitudes that are high enough to pass.

Next there is a layer that checks if the values of i are the first value, middle values or the last value. Depending on what the vaule of i is there is a slightly different operation done.

```
//if greater than the next value
if(DSPSnapshotInfo.pDSPSnapshotRt->data[i]>DSPSnap-
shotInfo.pDSPSnapshotRt->data[i+1])
{
//store value, the amplitude.....
DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i];
//one place gone from array
shortCounter=shortCounter+1;
}
```

Above is the operation done if the first value in the data is a peak. This actually should not be the case in normal operation but it is till calculated to cover all possibilities. The amplitude is then stored to a tdcOffset array inside DSPGetRtPtr(), and it has space for 4 values.

When i is not the first or the last value the operation differs from the above first values operation.

```
if(DSPSnapshotInfo.pDSPSnapshotRt->data[i]>=DSPSnapsho-
tInfo.pDSPSnapshotRt->data[i-1]        &&        DSPSnapsho-
tInfo.pDSPSnapshotRt->data[i]>DSPSnapshotInfo.pDSPSnapsho-
tRt->data[i+1])
```

If-condition checks if the data is greater than or equal to the previous data (i-1) and it also checks if the current data is larger than the next data. If this condition holds, the data can be stored. The harmonics are almost certainly only here in the middle values of i, which leads to storing the amplitudes of the harmonics into the tdcOffset array.

```
//store value
DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i];
//one place gone from array
shortCounter=shortCounter+1;
```

The fundamental frequency has to be in the middle values, so here the firstPeak variable is used to check if this is the first peak in the data.

```
if(firstPeak && DSPSnapshotInfo.pDSPSnapshotRt->data[i]>5000)
```

There is also a greater threshold for amplitude, because the fundamental frequency has the highest amplitude, and it is thus larger than the other harmonics. The threshold is also made bigger in order to reduce possible errors caused by noise. If the above two conditions are met, then the current data is the fundamental frequency. The data can be then used to calculate the rpm and the frequency.

```
//rpm
pRtParam->cylpres_2ndSF_delta_slope_den_el=(i*(25000/512))*
60/120;
//frequency
pRtParam->cylpres_2ndSF_delta_const_el=i*(25000/512);
//Knock frequency = KNOCK_TAP*(sample_frequency/fft_lenght)
//to rpm=Knock_frequency*60/120 teeth
firstPeak=false;
```

The rpm can be calculated using the information of point in x-axis, which is the value of i. i corresponds to the KNOCK_TAP variable in the knock frequency = KNOCK_TAP*(sample_frequency/fft_lenght) formula. To calculate rpm from the knock frequency, it can be first multiplied with 60 and then divided with 120 teeth. After storing these two values to the two different cylpress variables in pRtParam, the firstPeak variable can be set to false.

When the for loop reaches the last value of i, the calculations are done similarly as in the first value, but instead of comparing to next value the comparison is done to the previous value. The last value is also unlikely to hold any key information, but it is still checked for full coverage.

Another possibility for calculating the harmonics is to first utilize the part of the program that finds the fundamental frequency. When the fundamental frequency, or in other words the first harmonic has been found, the value of I can then be multiplied with 2,3,4 and so forth in order to find the rest of the harmonics.

```
DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i];
```

```
shortCounter=shortCounter+1;

DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i*2];

shortCounter=shortCounter+1;

DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i*3];

shortCounter=shortCounter+1;

DSPGetRtPtr()->pIntermRes->pGenericInfo->tdcOff-
set[shortCounter]=DSPSnapshotInfo.pDSPSnapshotRt->data[i*4];
```

Here the fundamental frequency (first peak) is stored first. Then the i*2, i*3 and i*4 are stored in the tdcOffset array. This method can actually be a better method because the fundamental frequency is quite easy to find, because of its characteristics (mainly amplitude) that make it stand out more than the other harmonics. Then the location of it can be used to find the other harmonics in a simple way. To make this method even better at finding the peaks, conditional statements can be used to check if the values around the multiplicative point such as i*2 have greater values in them and use those greater values in the calculation. This can be done because there is rounding and the peaks are not necessarily exactly in the multiplicative spot.

### 6.3.2 Build the project and transfer to modules

After the concept method was done, the project had to be built in eclipse. When the build is done, the resulting binary files needed to be reloaded in the Unitool, in order to get the changes made to the project. Then the software has to be downloaded to the used modules (CCM-30, COM-10 1 and 2) in the minirig. When the download process is finished the concept method can be tested.

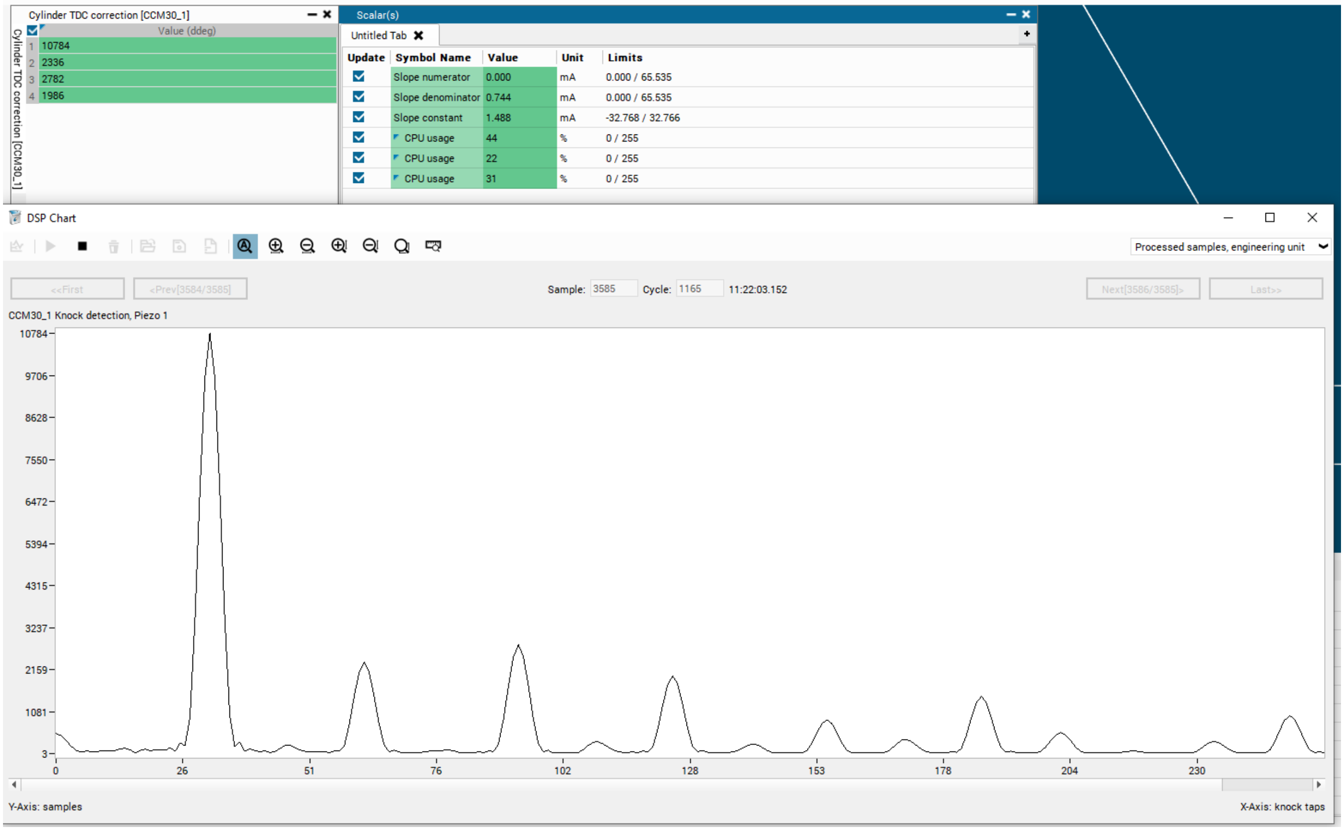
# 7 Evaluation

## 7.1 Speed measurement

The speed (Rpm) measurement can be achieved with the implementation of the concept method. The speed itself, which is the fundamental frequency can be accurately extracted from the FFT calculation. The Fundamental frequency represents the first, and most often the greatest peak in the frequency spectrum, and these characteristics lead to it being simple and clear to obtain.

### 7.1.1 Testing

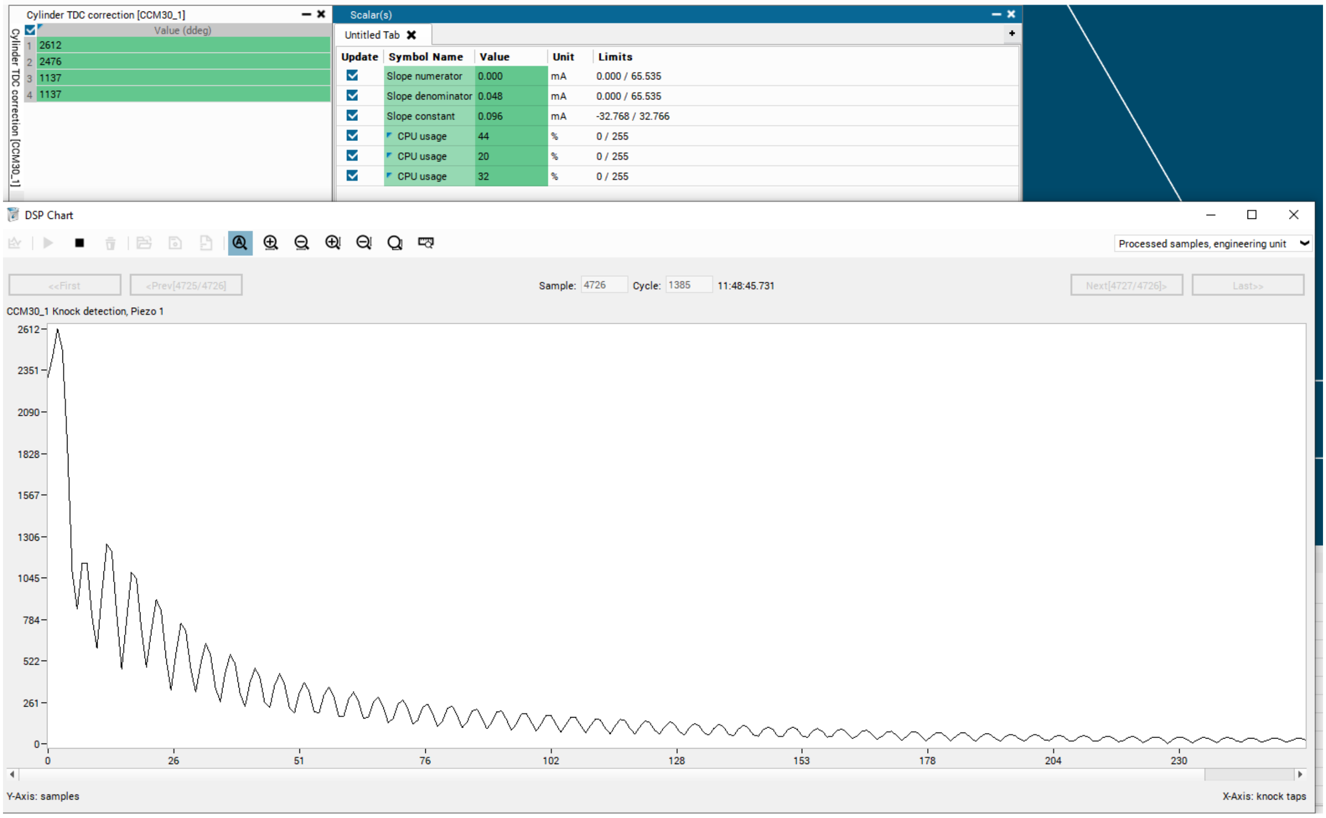
The testing was done with a minirig and the setup was the same as in the implementation chapter minirig test.




**Picture 16.** Unitool image from the minirig test. Slope denominator represents the KRrpm, and the Slope constant is the fundamental frequency in KHz. Cylinder TDC correction table shows the first 4 harmonic's amplitudes.

Picture 16 shows an example case, where the speed simulator is being set to 1488Hz. At 1488Hz the Rpm is 1488Hz*60/120=744Rpm. The amplitude of the fundamental frequency can then be seen from the cylinder TDC Correction table's first value, which is 10784. The RPM is being showed correctly in this case, and as in the implementation, the peak detection finds the fundamental frequency's peak if the peak has high enough threshold and if the peak is the first peak in the data.

The testing shows that the speed can be estimated with no errors, in favorable frequency ranges. For example at frequencies lower than 100Hz (corresponding o 50 Rpm), the data itself starts to become unclear to do any calculations with.




**Picture 17.** Minirig test, the data starts to become unclear at 96Hz.

In the picture 17 the data shows an example what the limit of readable is. Here the RPM is 48 (Slope denominator) and the frequency is 96Hz (slope constant). When setting frequency even lower than the picture 17's 96Hz, the data becomes unreadable. But in this point, the speed was still obtainable.

For comparison, when setting the simulators frequency to 3000Hz, the speed was still clearly readable, and no problems occurred.

Important aspect to take into account is that the FFT bin size is 25000Hz/512 (fft length)=48.83Hz. The resolution of the data is then 48.83Hz, which means that one step in the frequency axis is 48.83Hz. This leads to the fact that the smallest step in Rpm is roughly 24 Rpm. Smaller steps cannot be taken with these parameters (sampling frequency and fft-length). For example, if one would want to see the Rpm fluctuate between 740-760Rpm in steps of one Rpm, these parameters are not suitable for that.

In order to counter this the fft length could be increased to 4096, and the sampling frequency could be changed to 8 kHz. This would lead to the bin size of 8000Hz/4096=1.95Hz. The steps in Rpm would then be 0.97 Rpm, which is lower than 1Rpm. Of course the sampling frequency can be left at 25kHz, but then the fft length should be increased to 8192 to produce a frequency resolution of 3Hz and rpm step of 1.5Rpm.

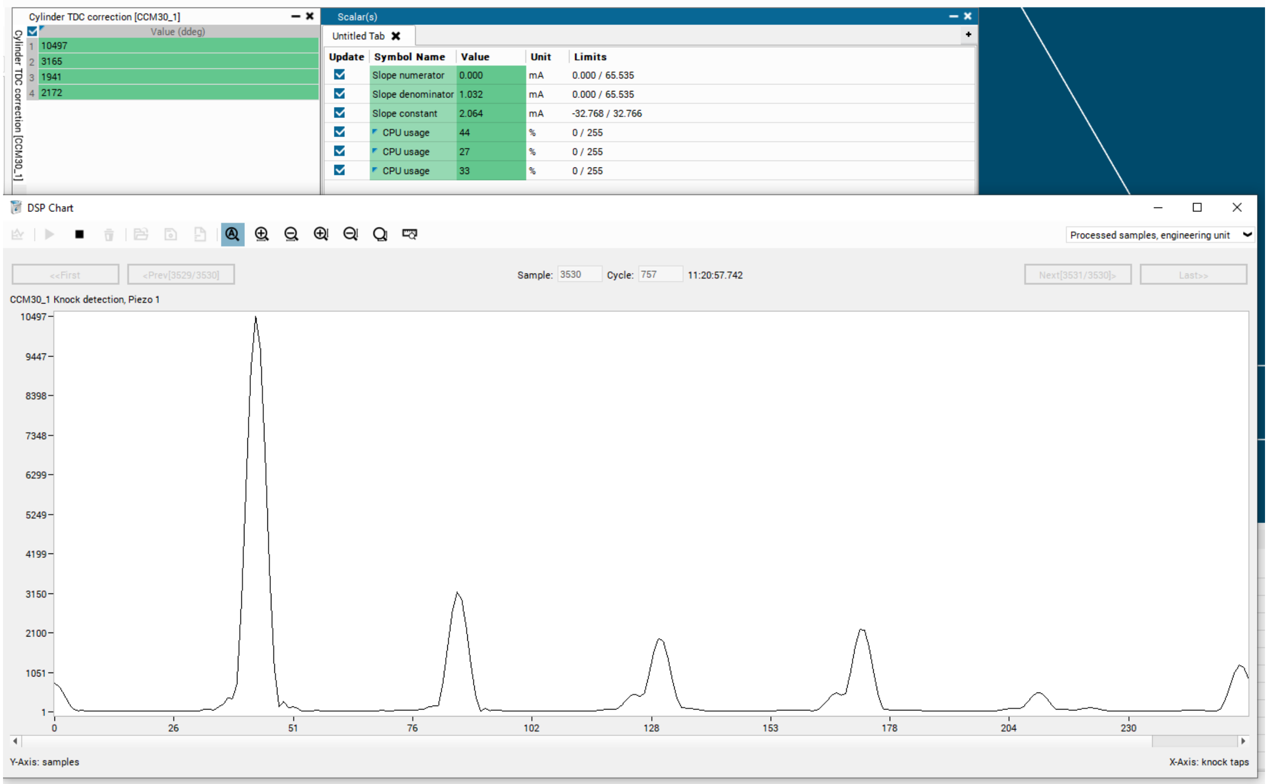
Other possibilities to increase the resolution are for example Gaussian interpolation or zero padding. Zero padding increases the Fast Fourier transforms accuracy by adding zeros to the signal (Hilbert 2013). The Gaussian interpolation produces an approximation of the continuous spectrum (Marsar 2015).

Nevertheless, the method would still function in the same manner if compared to a higher/lower resolution data. For example as shown in the design chapter, the MATLAB model functions similarly even with accurate real engine data. The peaks of the speed and harmonics are very similar in Picture 13 compared to picture 16, and the method should work similarly in both cases.

## 7.2 Harmonics

### 7.2.1 Testing

The harmonics were also accurately measured in the normal frequency range of operating engine (1000-3000Hz). This means that the peaks of the different harmonics, such as 2nd, 3rd and so on were clearly visible and there were no errors in detecting them.




**Picture 18.** Minirig test, showing the FFT graph, Rpm and harmonic amplitudes.

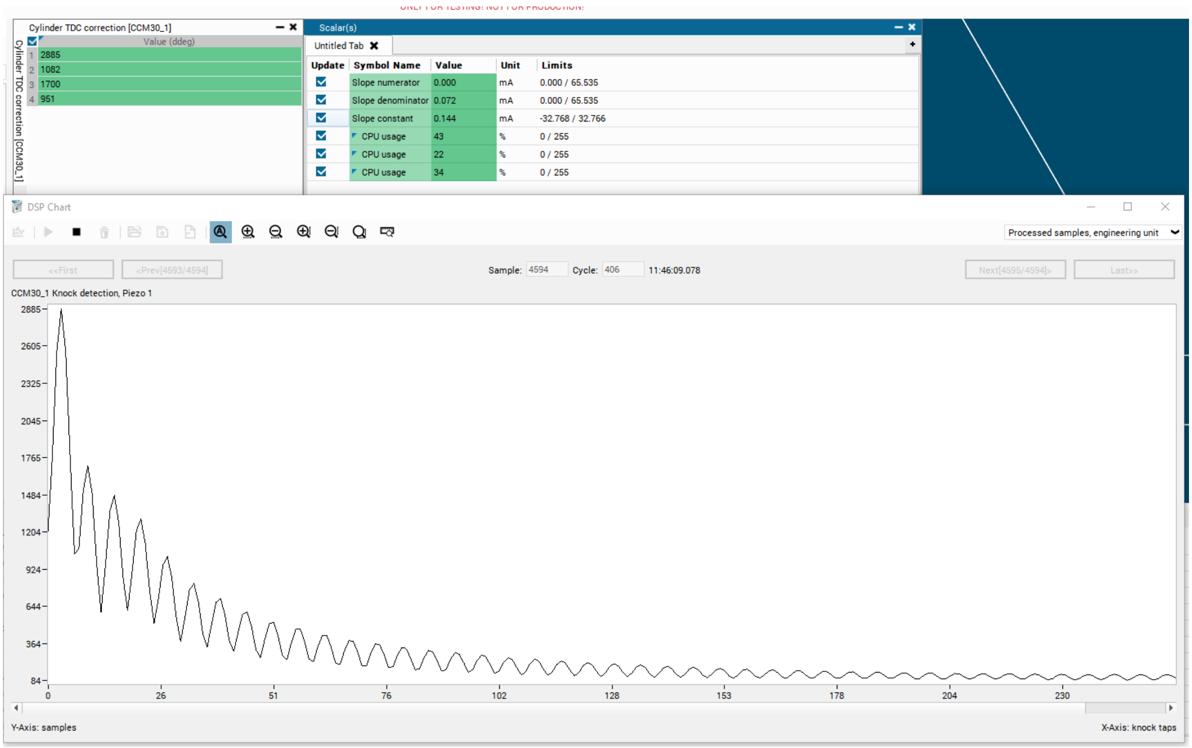
As shown in the picture 18, simulator was set to 2064Hz, and the results were accurately obtained. Rpm was 1032, and the amplitudes were equal to the amplitudes in the graph (Cylinder TDC Correction table compared to the graph).

**Table 1.** Amplitudes of the harmonics from picture 18.

| | |
|---|---|
| 1st Harmonic | 10497 |
| 2nd Harmonic | 3165 |
| 3rd Harmonic | 1941 |
| 4th Harmonic | 2172 |

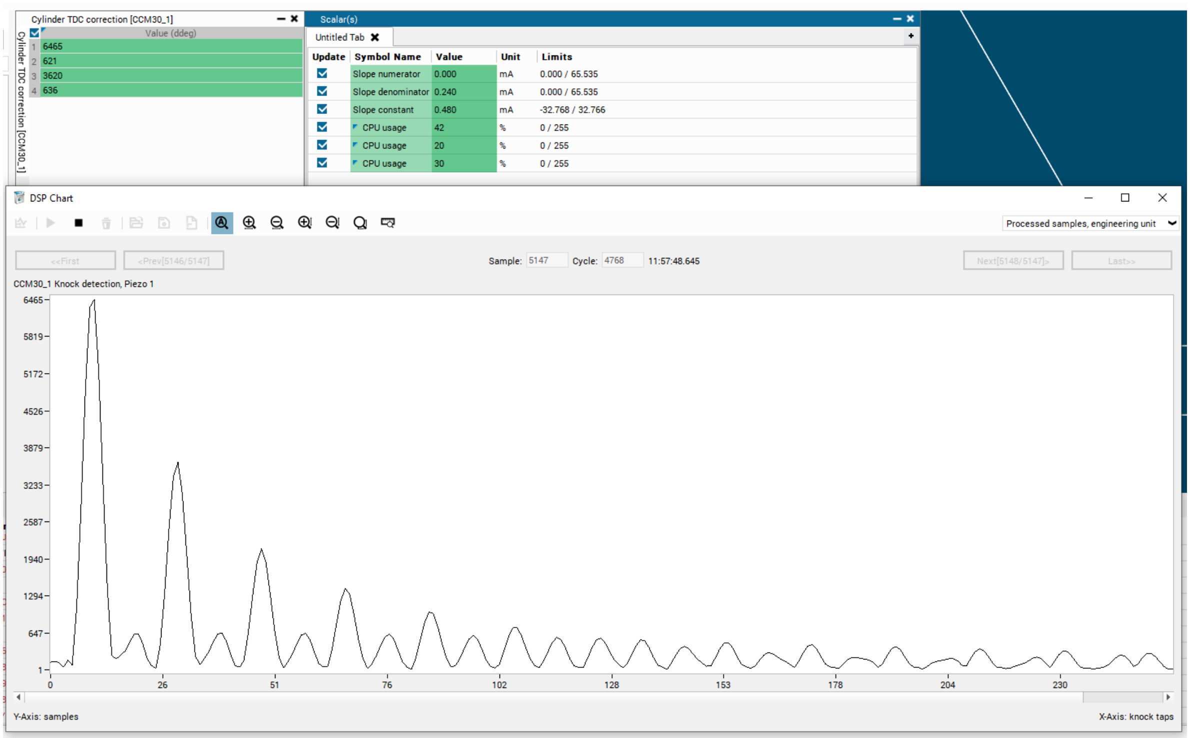
Here in the above minirig test pictures 16, 17 and 18. The method is used where the fundamental frequency is multiplied to find the other harmonics. The peak detection is not finding any false peaks and count them as the harmonics. This makes this method robust in finding the correct peaks, when the frequency is in the suitable range.

When the frequency is too low, the data starts to become unclear just like in the speed measurement chapter.




**Picture 19.** Minirig test, The data at 144Hz.

For example, picture 19 shows that the method finds the harmonics, but the information might not be usable at low frequency like this. There is of course the fundamental frequency's peak that shows the speed and the fundamental frequency can then be multiplied just like it is done in the Cylinder TDC Correction table. The harmonics are then calculated, but most likely the data is not holding much relevant information at this point. When the frequency starts to grow, then the data becomes clear again.




**Picture 20.** Minirig test, data is clearer at higher frequency (480Hz).

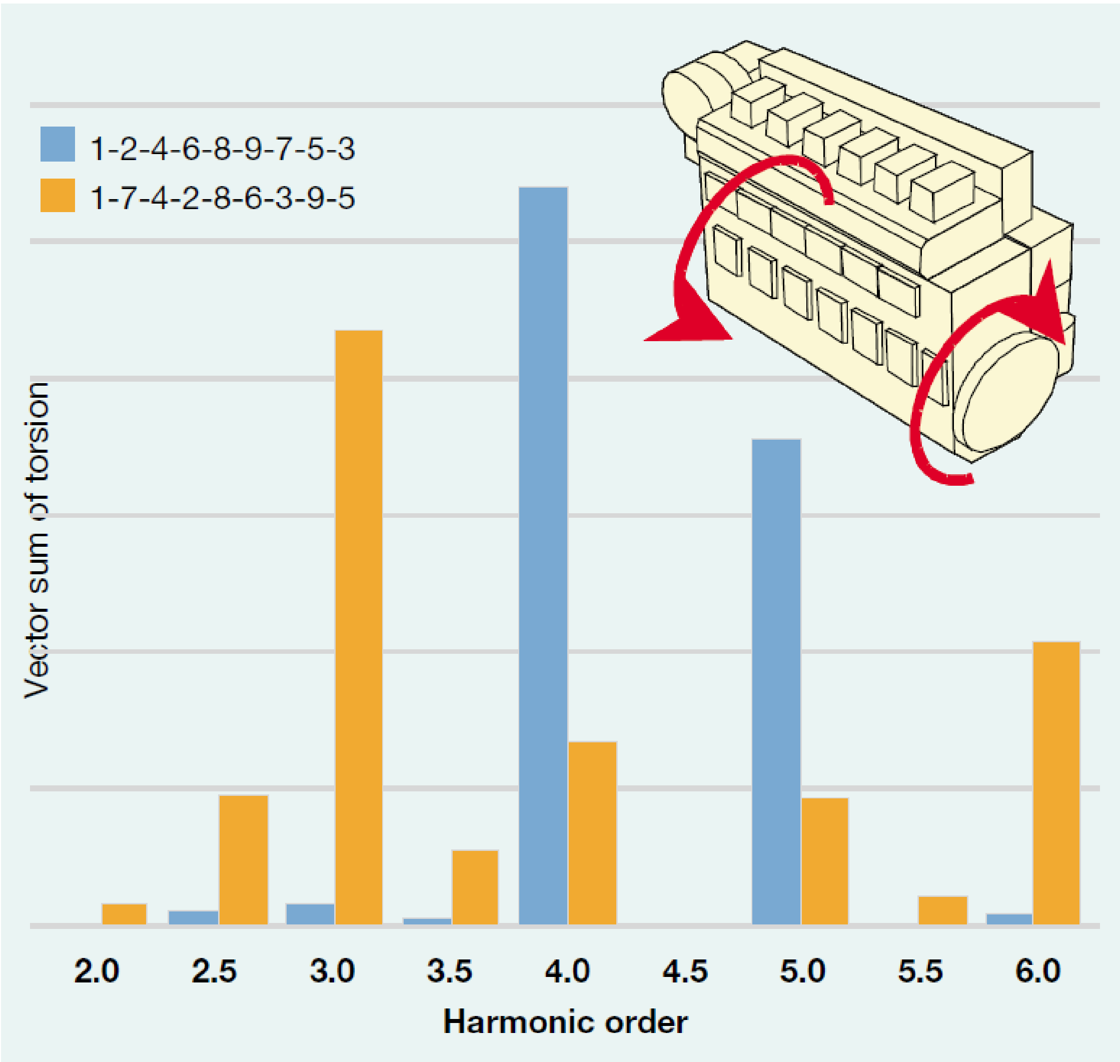
In picture 20, as the frequency grows the data becomes more clear, and the harmonics start to be visible. Then the harmonic amplitude calculations should start to give better information about the state of the engine.

### 7.2.2 Information in the harmonics

As earlier discovered, for example the 0.5 harmonic is related to the firing frequency, and it can be used to calculate the engine bank frequency by multiplying it with the number of cylinders in the bank.

But what about the harmonics at the multiplication points such as 2, 3 4 and so forth? They are related to the crankshafts movement, which is caused by the combustion events happening in the cylinders. Thus the harmonics are a result of many functions happening inside the engine. The engine might not be running perfectly balanced for example, and this could explain the amplitudes of certain harmonics. Or on the opposite side, the engine might be running in balance, and the engine creates these harmonics in normal operation.

The firing order of the engine also has an effect on the harmonics.




**Picture 21.** Firing orders and the harmonics (Tienhaara 2004).

As shown in the picture 21, the firing order has a great effect on the amplitudes of certain harmonics. In a situation where both of the engines are 9-cylinder engines, but with different firing orders. The amplitudes of the harmonics orders are entirely different between the two engines.

There can be several use cases for storing the amplitudes of the harmonics. For example one monitoring method could be to store the amplitudes of the harmonics of an engine at the base load (750Rpm). Then the amplitude values of the harmonics could be compared to the amplitudes of known normal operation amplitudes of the harmonics in that engine. Like in the Cesar Palestenis thesis, The faulty operation of an engine showed that the amplitudes of the harmonics had greater variance in them (Picture 11). This information could then be used to check, if the engine is running in good state.

Through testing, different kinds of faulty conditions and their harmonics could be stored and this information could then be used in a greater extent to have more data in comparing faulty and healthy engines. There could be a case where testing shows that a certain harmonic's amplitude grows with torsional vibration. This value of the amplitude could be a used in measurements related to analyse if crankcshaft is in good condition or not.

## 7.3 Effects on CPU load

The most demanding part of the concept method is the Fast Fourier transform. The FFT can have significant impact on the CPU load. But here measuring the CPU load caused by the FFT is not easily done, because it has program separated in many parts and deleting that in order to see how it affects the load would be a complicated task. But nevertheless, the effect on CPU load is high.

The other part of the concept method, which is the peak and harmonic detection should not cause too much excess CPU load. This is because of the fact that the method's most demanding parts are 1-dimensional for-loops. These for-loops can of course cause some increase in load, and the load increases as the FFT window length grows. When testing this part of the method in a 2-minute CPU load test, it was found that the method had no impact on the CPU load.

**Table 2.** CPU load compared to the concept method being used or not.

| CPU LOAD | ON | OFF |
|---|---|---|
| CCM-30 | 43,80% | 44,42% |
| COM-10 1 | 31,91% | 31,13% |
| COM-10 2 | 19,24% | 18,85% |

Table 2 shows that the peak and harmonic detection part of the method should not cause noticeable CPU load increase. The CCM-30 is in fact the most important of the modules, because that is where the method is performed. In the CCM-30 the average load was actually smaller when the method was used, so there should not be any significant impact on CPU performance. The two COM-10 modules CPU load was also measured, because they are part of the system. Two COM-10 modules had similar CPU load with and without the method being used.

## 7.4 Future work/Continuation

For the continuation of this thesis, an approach that utilizes the angular domain could be investigated. As said in Dr. -Ing. Dejan Arsic (2016) article, the angular domain can help in accurately locate the faulty cylinder in the engine. The Fast Fourier transforms could be synchronized with the engine's flywheel's angle to produce results from each cylinder separately. This would help in detecting the normally operating cylinders and the faulty ones with higher accuracy.

To further monitor the engine’s health, machine learning methods could be utilized to learn the different types of faults and normal operation in the engine. This could be learnt from using the amplitude values of different harmonics and the angle value of the flywheel to locate the correct cylinder. Naturally the best source of data would be a real engine with certain known fault.

# 8 Conclusions

By analysing the flywheels speed signal in the frequency domain it was possible to extract the fundamental frequency and it's harmonics. This data includes the engine's speed which is the fundamental frequency. It can be used to calculate the Rpm in which the engine is running. The calculation can be done in a robust way, when the engine is running in high enough rpm, for example over 50 Rpm. Otherwise, in the relatively low frequency for the flywheel, the data becomes unclear to do any calculations with. This method can still be used for calculating the speed of the engine.

The harmonics can also be calculated accurately when the Rpm of the Flywheel is high enough, similar to the speed calculation. There are multiple functions in the engine that affect and cause the harmonics and the fundamental frequency. Information related to the harmonics can be used mainly to check if the engine is running normally, and that no faults are present. The faulty or normal conditions can be seen and analysed from the amplitudes of the harmonics.

The concept method appears to be promising in calculating the engine speed. This method could be used on a real engine to provide relatively robust speed estimation and possibly replace the current methods in that area. For another positive side in calculating the speed, harmonics can be calculated at the same time and from the same frequency domain data. This makes the concept even more appealing, because the harmonics include beneficial information about the engine's state.

# References

Analog (2020). A Beginners guide to Digital signal processing (DSP) [online].[28.8.2020] Available: https://www.analog.com/en/design-center/landing-pages/001/beginners-guide-to-dsp.html#

Breeze, Paul (2018). Piston engine-based power plants. ScienceDirect [online].[25.8.2020] Available: https://www.sciencedirect.com/topics/engineering/diesel-engines

Brilliant.org (2020). Discrete Fourier transform [online].[29.11.2020] Available: https://brilliant.org/wiki/discrete-fourier-transform/

Chanrasetty Arkalgud, Vikram & Aziz Mahfuzul, Syed (2018). Resource efficient LDPC Decoders. ScienceDirect [online].[27.8.2020] Available: https://www.sciencedirect.com/topics/engineering/field-programmable-gate-arrays

Chen Jian (2013). Internal combustion engine diagnostics using vibration simulation [doctoral thesis, University of New South Wales].[24.9.2020]

Deep AI (2020). Frequency Domain [online].[29.8.2020] Available: https://deepai.org/machine-learning-glossary-and-terms/frequency-domain

Deep AI (2020b). Fourier transform [online].[29.8.2020] Available: https://deepai.org/machine-learning-glossary-and-terms/fourier-transform

Dewesoft (2020). Products [online].[30.9.2020] Available: https://dewesoft.com/products

Dr. -Ing. Dejan Arsic (2016). Rotational analysis in the angular domain, Muller-BBM microacoustic systeme [online]. [25.11.2020] Available: https://www.mbbm-vas.com/fileadmin/Publications/White_Paper___Use_Cases/PAK_RotAn_WP_1605_EN.pdf

Energy Education (2020). Diesel engine [online].[25.8.2020] Available: https://energyeducation.ca/encyclopedia/Diesel_engine

Energy.GOV (2013). Internal combustion engine basics [online].[2.11.2020] Available: https://www.energy.gov/eere/vehicles/articles/internal-combustion-engine-basics

Gatan. Nyquist Frequency [online].[9.10.2020] Available: https://www.gatan.com/nyquist-frequency

Gawande, S.H & Navale, L.G & Nandgaonkar, M.R & Butala, D.S & Kunamalla, S (2012). Fault detection of inline reciprocating diesel engine: a mass and gas-torque approach [23.9.2020]

Grout, Ian (2008). Digital Systems Design with FPGAs and CPLDs. ScienceDirect [online].[28.8.2020] Available: https://www.sciencedirect.com/topics/engineering/analog-to-digital-converter

Hilbert, Shannon (2013). FFT Zero Padding, Bitweenie [online].[30.11.2020] Available: https://www.bitweenie.com/listings/fft-zero-padding/

Lewis, Kim (2017). How to convert Hertz to motor RPM, Sciencing [online].[7.10.2020] Available: https://sciencing.com/convert-hertz-motor-rpm-5728009.html

MAN Diesel and Turbo (2013). Two-stroke Low Speed Diesel Engines for Independent Power Producers and Captive Power Plants [online].[9.1.2021] Available: https://www.mandieselturbo.com/docs/default-source/shopwaredocumentsarchive/two-stroke-low-speed-diesel-engines.p

Marsar, J. (2015). Improving FFT frequency resolution [online].[30.11.2020] Available: http://www.add.ece.ufl.edu/4511/references/ImprovingFFTResoltuion.pdf

MathWorks (2020). What is MATLAB [online].[9.9.2020] Available: https://se.mathworks.com/discovery/what-is-matlab.html

Maxwell Ford (2020). What's the difference between powertrain and drivetrain? [online].[31.8.2020] Available: https://www.maxwellford.com/powertrain-and-drivetrain.html

Mendez, Arnaldo & Sawan, Mohamad (2015). Implantable biomedical microsystems. ScienceDirect [online].[27.8.2020] Available: https://www.sciencedirect.com/topics/engineering/field-programmable-gate-arrays

National instruments. Understanding FFT's and windowing [online].[8.10.2020] Available:https://download.ni.com/evaluation/pxi/Understanding%20FFTs%20and%20Windowing.pdf

Palestini, Cesare (2018). Engine crankshaft torsional vibration analysis for anomalies detection [online]. [13.11.2020] Available: https://aaltodoc.aalto.fi/handle/123456789/32444

Scienceworld (2021). Sound = Vibration, Vibration, Vibration [online].[9.1.2021] Available: https://www.scienceworld.ca/resource/sound-vibration-vibration-vibration/

SciPy.org (2020). Scientific computing tools for Python [online].[9.9.2020] Available: https://www.scipy.org/about.html

Sneha, H.L (2017). Purpose and internal funtionalit of FPGA Look-Up tables. ALL ABOUT CIRCUITS [online].[27.8.2020] Available:

https://www.allaboutcircuits.com/technical-articles/purpose-and-internal-functionality-of-fpga-look-up-tables/

Steven W, Smith (1997). The Scientist and Engineer's Guide to Digital Signal Processing. ISBN 0-9660176-3-3 [online].[22.10.2020]. Available: http://www.dspguide.com/ch13/4.htm

Sun, Liming & Luo, Fuqiang & Shang, Tansu & Hen, Hongtao & Moro Adams (2017). Research on torsional vibration reduction of crankshaft in off-road diesel engine by simulation and experiment. Jve journals [online]. [24.9.2020] Available: https://www.jvejournals.com/article/18318

The Physics classroom (2020). Fundamental frequency and harmonics [online].[31.8.2020] Available: https://www.physicsclassroom.com/class/sound/Lesson-4/Fundamental-Frequency-and-Harmonics

Tienhaara, Hannu (2004). Guidelines to engine dynamics and vibration, Wärtsilä [online].[2.12.2020] Available: https://www.maintenance.org/fileSendAction/fcType/0/fcOid/399590942962914489/filePointer/399590942964822834/fodoid/399590942964822832/Wartsila_Guideline_Dynamics_Vibrations.pdf

Tiete, J.& Dominguez, F. & da Silva, B. & Touhafi, A & Steenhaut, K. (2017). MEMS microphones for wireless applications. ScienceDirect [online].[27.8.2020] Available: https://www.sciencedirect.com/topics/engineering/field-programmable-gate-arrays

U.S Energy Information Administration (2020). Diesel fuel explained: Use of diesel [online].[2.8.2020] Available: https://www.eia.gov/energyexplained/diesel-fuel/use-of-diesel.php

Vuollet, Tomi (2017). Next generation UNIC automation system enables Wärtsilä 31 performance [online].[2.9.2020] Available:

https://www.wartsila.com/twentyfour7/in-detail/next-generation-unic-automation-system-enables-wartsila-31-performance

Wolfram MathWorld. Nyquist Frequency [online].[9.10.2020] Available: https://mathworld.wolfram.com/NyquistFrequency.html

Wärtsilä (2020a). Diesel engine [online].[24.8.2020] Available: https://www.wartsila.com/encyclopedia/term/diesel-engine

Wärtsilä (2020b). Internal combustion engine [online].[24.8.2020] Available: https://www.wartsila.com/encyclopedia/term/internal-combustion-engine

Wärtsilä (2020c). Wärtsilä 31 [online].[25.8.2020] Available: https://www.wartsila.com/marine/build/engines-and-generating-sets/diesel-engines/wartsila-31

Wärtsilä (2020d). Wärtsilä 31SG [online].[16.11.2019] Available: https://www.wartsila.com/marine/build/engines-and-generating-sets/pure-gas-engines/wartsila-31sg

Wärtsilä (2017). Wärtsilä 31 [online].[1.9.2020] Available: https://cdn.wartsila.com/docs/default-source/product-files/engines/brochure-o-e-unic-ms.pdf?sfvrsn=7ab39545_4

Xilinx (2020). Field Programmable Gate Array (FPGA) [online].[27.8.2020] Available: https://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.html

Young, Hugh D. & Freedman, Roger A. (2012). University Physics. 13. edition. Plano: Pearson ISBN 13: 978-0-321-69686-1.

Zhenyu Yang (2015). Automatic Condition Monitoring of Industrial Rolling-Element Bearings Using Motor's Vibration and Current Analysis [online]. [16.10.2020] Available: https://www.researchgate.net/publication/282305689