



Recording and Reproduction of Engine Knock Signal

Matts Finnholm

Bachelor's Thesis
Electrical Engineering
Vaasa 2017



BACHELOR'S THESIS

Author: Matts Finnholm

Degree Programme: Electrical Engineering, Vaasa

Specialization: Automation Technology

Supervisors: Matts Nickull

Title: Recording and Reproduction of Engine Knock Signal

Date April 20, 2017

Number of pages 49

This thesis is about engine knock and knock control on Wärtsilä gas engines in general. The purpose of the thesis work was to develop a method to record and send pre-recorded knock sensor signals to the engine control unit. The signals must also be synchronized with the engine speed, to simulate the knock signals that are present in the engine naturally at certain values of the crank angle.

Measurement equipment and software from Dewesoft were used as a part of this thesis to record and reproduce the knock sensor signals. If needed, hardware adapters would also be developed to ensure interference-free communication between the measurement device and the engine control module.

This thesis contains information about the causes and effects of knocking on an engine, and a brief introduction to gas engines and knock sensors. The result from this thesis was a method to send pre-recorded knock sensor signals, which are synchronized with the engine speed, to the engine control unit.

Language: English

Key words: knock, signal manipulation, data acquisition

EXAMENSARBETE

Författare: Matts Finnholm

Utbildningsprogram och ort: Elektroteknik, Vasa

Inriktningalternativ: Automationsteknik

Handledare: Matts Nickull

Titel: Inspelning och återgivning av knocksensornsignaler

Datum 20.04.2017

Sidantal 49

Examensarbetet handlar om motorknock och motorknockkontrollsystem på Wärtsiläs gasmotorer i allmänhet. Syftet med examensarbetet var att utveckla en metod för att spela in och skicka förinspelade knocksensornsignaler till motorstyrenheten.

Signalerna måste också synkroniseras med motorns varvtal, för att efterlikna de knocksignaler som naturligt uppstår i motorn vid vissa vevaxelpositioner.

Mätutrustning och programvara från Dewesoft användes som en del av detta arbete för att registrera och återge knocksensornsignalerna. Vid behov kommer även hårdvaruadaptorer att tas fram för att säkerställa störningsfri kommunikation mellan mätutrustningen och motorstyrenheten.

Arbetet innehåller information om orsaker och effekter av motorknock på en motor, samt en kort introduktion till gasmotorer och knocksensorer. Resultatet från detta arbete var en metod för att skicka förinspelade knocksensornsignaler, som är synkroniserade med motorns varvtal, till motorstyrenheten.

Språk: engelska

Nyckelord: knock, signalbehandling, datainsamling

OPINNÄYTETYÖ

Tekijä: Matts Finnholm

Koulutusohjelma ja paikkakunta: Sähkötekniikka, Vaasa

Suuntautumisvaihtoehto: Automaatiotekniikka

Ohjaaja: Matts Nickull

Nimike: Nakutusantureiden signaalien nauhoittamisesta ja toistaminen

Päivämäärä 20.04.2017

Sivumäärä 49

Tämä opinnäytetyö käsittelee moottoreiden nakutusta ja nakutuksen ohjausta Wärtsilän kaasumoottoreissa. Opinnäytetyön tavoitteena oli kehittää tapa, miten voi nauhoittaa ja lähettää nakutusantureiden signaalit moottoriohjausyksikköön. Signaalit pitää synkronoida moottorin nopeuteen matkiakseen nakutussignaaleja jotka esiintyvät moottorissa normaalisti tietyissä kampiakselin kohdissa.

Mittauslaitteita ja ohjelmistoja Dewesoftilta käytettiin osana tätä työtä nakutusantureiden signaalien tallentamiseen ja toistamiseen. Tarvittaessa kehitetään myös sovittimia, jotta voidaan varmistaa häiriötön viestintä mittauslaitteen ja moottorin ohjausyksikön välillä.

Tämä työ sisältää tietoa moottorien nakutuksen syistä ja seurauksista, ja lyhyt johdatus kaasumoottoreihin ja nakutusantureihin. Työn tulos on menetelmä lähettää moottorin ohjausyksikköön nauhoitettuja nakutussignaaleja, jotka ovat synkronoituja moottorin nopeuteen.

Kieli: englanti

Avainsanat: nakutus, signaalinkäsittely, tiedonkeruu

Table of Contents

1	Introduction	1
1.1	Objective	1
1.2	Organization	2
1.2.1	Marine solutions	2
1.2.2	Energy Solutions	2
1.2.3	Services.....	2
2	Theory.....	3
2.1	Combustion engines	3
2.1.1	Gas engines.....	6
2.1.2	Wärtsilä gas engines.....	7
2.2	Engine control UNIC.....	7
2.2.1	UNIC C1.....	8
2.2.2	UNIC C2.....	9
2.2.3	UNIC C3.....	9
2.2.4	UNIC Modules.....	9
2.2.5	WECSplorer.....	10
2.3	Engine knock.....	11
2.3.1	Causes.....	12
2.3.2	Effects.....	13
2.4	Knock detection	13
2.4.1	Pressure sensor	13
2.4.2	Piezo Knock sensor	14
2.4.3	Piezoelectric effect	16
2.5	Speed and position measurement	17
2.5.1	Position measurement.....	18
2.5.2	Phase measurement	19
2.6	Measurement equipment.....	20
2.6.1	Hardware	20
2.7	Software	21
2.7.1	Counters	21
2.7.2	Analog out.....	21
2.7.3	Combustion analysis.....	22
2.7.4	Sequencer	25
2.8	Reference curves	27
3	Implementation	29
3.1	Test setup	29

3.2	Recording	30
3.2.1	Combustion analysis.....	30
3.3	Reproduction	30
3.3.1	Function generator	30
3.3.2	A/D Channel.....	31
3.4	Importing.....	34
3.5	Synchronization.....	34
3.5.1	Math implementation.....	34
3.5.2	Delay	37
3.5.3	Sequencer	38
4	Results	44
5	Discussion	45
5.1	Future development.....	46
6	Bibliography.....	48

Abbreviations

TDC – Top Dead Center

BDC – Bottom Dead Center

UNIC – Unified Engine Control Unit

WECS - Wärtsilä Engine Control System

ESM - Engine Safety Module

MCM – Main Control Module

PDM – Power Distribution Module

LCP – Local Control Panel

CCM – Cylinder Control Module

LDU – Local Display Unit

IOM – Input Output Module

BMEP – Break Mean Effective Pressure

SG – Single Gas

DF – Dual Fuel

GD - Gas Diesel

APR – Anti-Polishing Ring

RPM – Revolution per Minute

SOC – Start of Combustion

EOC - End of Combustion

DAQ – Data Acquisition

1 Introduction

The engine control systems of today are designed to minimize emissions while maximizing power and getting the best fuel economy. The capability to get the most power with the best fuel economy by optimizing spark timing for a certain air/fuel ratio is limited by engine knock. By detecting engine knock and controlling the timing of the ignition to allow an engine to run at the knock threshold provides the best power and fuel economy.

But sometimes the systems may malfunction, and when that happens, there is a need for a way to locate the error in the system, and that is what this thesis is about.

1.1 Objective

The main target of this thesis is to have a method and equipment that enables recording and reproduction of piezoelectric knock sensor signals for Wärtsilä's engine control system, UNIC.

The reason for wanting to record and reproduce the signals is that if it is suspected that there is a faulty sensor. Then the signal from a known functioning sensor is recorded, and that signal is replayed to the engine control system.

The outcome from that is to determine the behaviour of the system. If the system is working correctly with the reproduced signal, then it is proved that there was a malfunctioning knock sensor. However, if the problem still occurs, then there might be a problem with the cylinder control unit.

Equipment from Dewesoft will be used to record and replay the knock sensor signals to Wärtsilä's engine control system.

The signals must also be synchronized to the engine cycles and crank angle. The window where engine knock naturally occurs, the "knock window", is between 10°CA and 50°CA after TDC (Top Dead Center).

This thesis will only include the development of a method of recording and reproducing the knock signals. A tool for field use will not be developed in this thesis nor will the knock signals be analysed deeper.

1.2 Organization

Wärtsilä is a global frontrunner in complete power solutions for marine and energy markets. By emphasizing technological innovation and total efficiency, Wärtsilä maximizes the environmental and economic performance of the vessels and power plants of its customers. In 2016, Wärtsilä's net sales totalled EUR 4.8 billion with about 18,000 employees. The company has operations in more than 200 locations in nearly 70 countries around the world. Wärtsilä is listed on NASDAQ Helsinki. Wärtsilä consists of three main divisions; Marine solutions, Energy Solutions and Services.

1.2.1 Marine solutions

Wärtsilä Marine Solutions enhances the business of its customers by offering the marine industry, integrated system solutions and products that are efficient, economically sound, and ecologically sustainable. The solutions are developed based on customers' needs and include products, systems and services.

1.2.2 Energy Solutions

Wärtsilä Energy Solutions is a leading global system integrator offering a broad range of environmentally sound solutions. The product portfolio includes flexible internal combustion engine based power plants and utility-scale solar PV power plants, as well as LNG terminals and distribution systems.

As of 2017, Wärtsilä has 63 GW of installed power plant capacity in 176 countries worldwide.

1.2.3 Services

Wärtsilä Services supports its customers throughout the life-cycle of their installations by optimizing the efficiency and performance. Wärtsilä Services offers service to both the energy and the marine market. [1]

2 Theory

In this chapter, background theory about the techniques and methods used in this thesis will be explained. Only theory about four-stroke engines will be included in this thesis, as most of Wärtsilä engines are fore-stroke engines and the engines which this thesis touches are four-stroke engines.

2.1 Combustion engines

A four-stroke engine is an internal combustion engine in which the piston completes four separate strokes that turn the crankshaft. A stroke refers to the full travel of the piston along the cylinder, from TC to BC as pictured in *Figure 1*, in either direction.

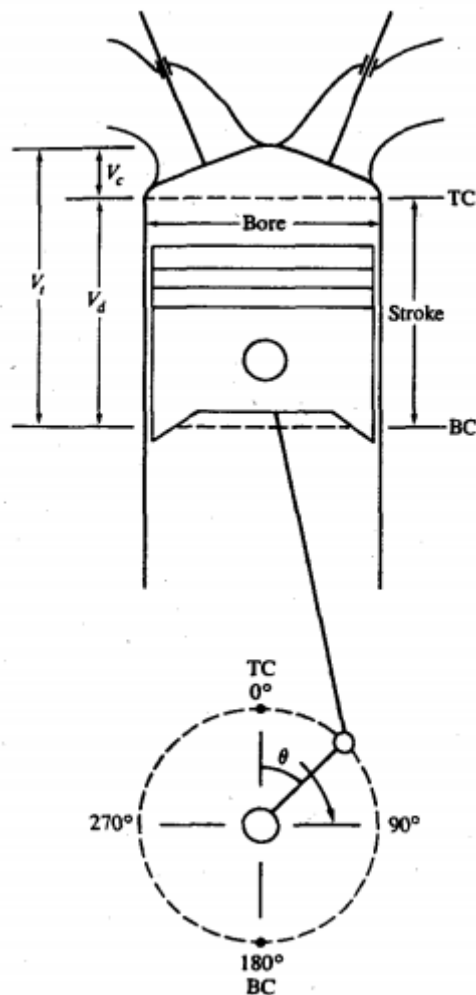


Figure 1. Illustration showing the movement of the piston. [2]

The four separate strokes are termed:

1. **Intake stroke** – The piston starts at TC and ends with the piston at BC, which draws fresh mixture into the cylinder. [2]
2. **Compression stroke** – When the piston has passed BDC, the intake valve closes and the piston moves upward. As the piston moves up, the air-fuel mixture is compressed, to a tenth of its volume that it had in the start of the cycle. [2]

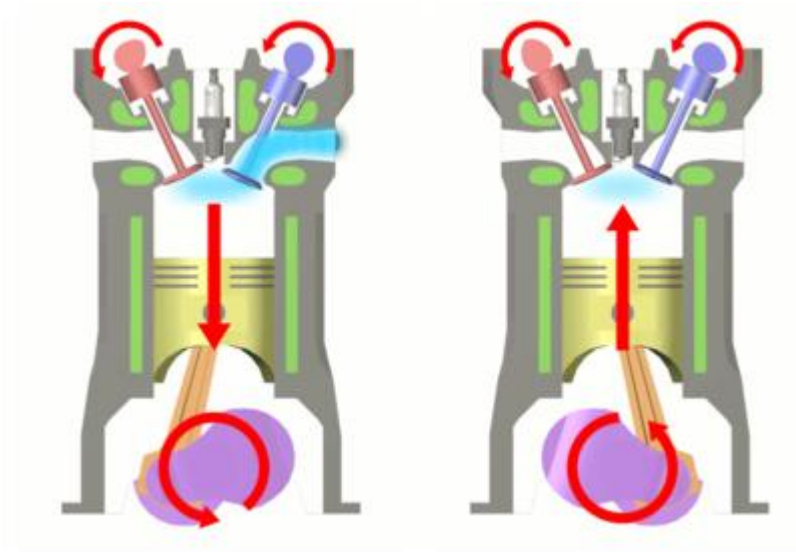


Figure 2. Intake stroke and compression stroke. [3]

3. **Power stroke** – When the piston reaches TC, the air-fuel mixture is ignited by a spark that is generated by a spark plug. The high-temperature, high pressure gases push the piston downwards and forces the crank to rotate. [2]
4. **Exhaust stroke** – As the piston reaches BC during the power stroke combustion is complete and the cylinder is filled with exhaust gases. The exhaust valve opens, and inertia of the flywheel and other moving parts push the piston back to TC, forcing the exhaust gases out through the open exhaust valve. At the end of the exhaust stroke, the piston is at TC and one operating cycle has been completed and the cycle starts again. [2]

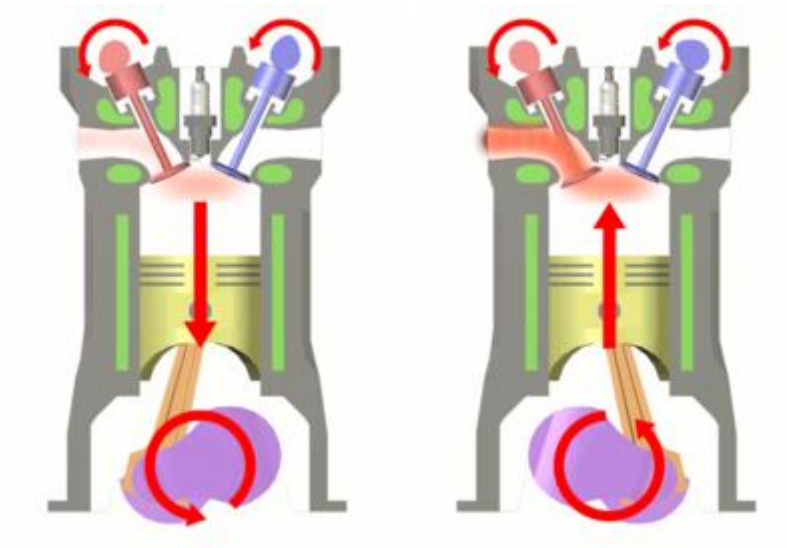


Figure 3. Power stroke and exhaust stroke. [3]

2.1.1 Gas engines

Gas engines operate with lean burn combustion. This means that the air/fuel ratio (λ) in gas engines is around 2, compared to a diesel engine for which the ratio is about 1. If the λ value is over 1, it means that the air/fuel mixture contains more air than fuel, a lean mixture. And vice versa if λ is below 1, the air/fuel mixture contains more fuel, a rich mixture.

Thanks to this the gas engines produce less emissions and they have higher efficiency. *Figure 4* shows the operating window for gas engines. From there the benefits of a gas engine, compared to a diesel engine, can be seen. As the air/fuel ratio increases, the NOx values decrease and the thermal efficiency increases. A higher BMEP (Break Mean Effective Pressure) can also be achieved when the λ value is increased. The challenging part that gas engines bring is a narrower operating window, which requires the actuators and control system to be more precise. [4]

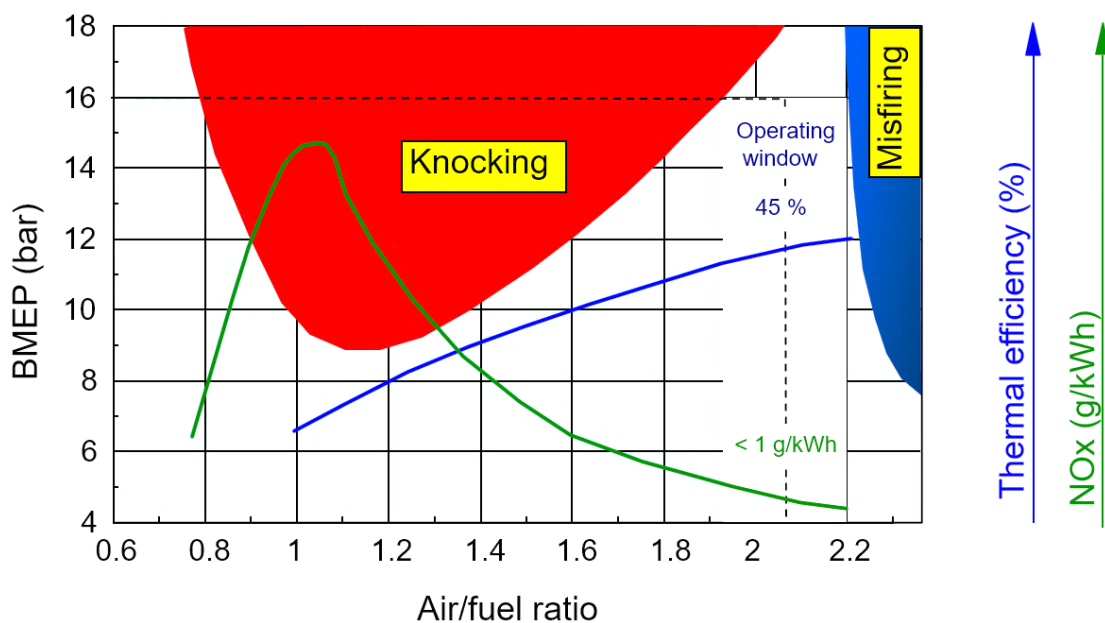


Figure 4. Lean burn combustion. [4]

2.1.2 Wärtsilä gas engines

Wärtsilä has several different types of gas-fuelled engines

- **SG engines** – Runs only on gas.
- **DF and GD** – runs both on liquid fuels or gas.

SG and DF are the most common engine types manufactured by Wärtsilä. [4]

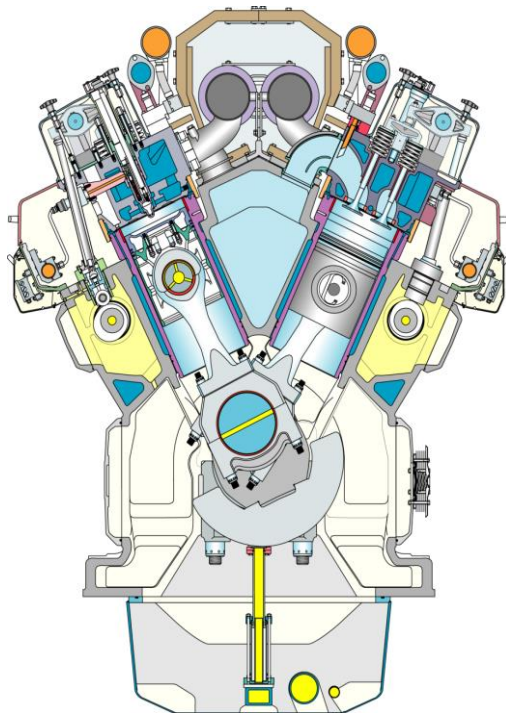


Figure 5. 34SG engine. [4]

2.2 Engine control UNIC

UNIC is an embedded engine management system. The systems design is modular. Some modules and functions in the setup are not always needed, depending on the engine size and type.

The system is constructed and designed for the demanding environment on and around engines. Therefore, temperature and vibration has been taken into consideration to get a robust and enduring design.

The system is mounted directly on the engine. Therefore, the design will be compact and minimalistic, and eliminates the need for external cabinets.

With the modules mounted directly on the engine allows full testing of the engine at the factory. The number of in- and outputs is determined to optimally suit the application, and the galvanic signal isolation is also made to match these needs.

WECS is the predecessor of the UNIC system, and is still in use on older systems. [5]

2.2.1 UNIC C1

Unic C1 is used for common diesel engines, and it manages basic engine safety and control. The system has a hardwired interface for communicating with sensors.

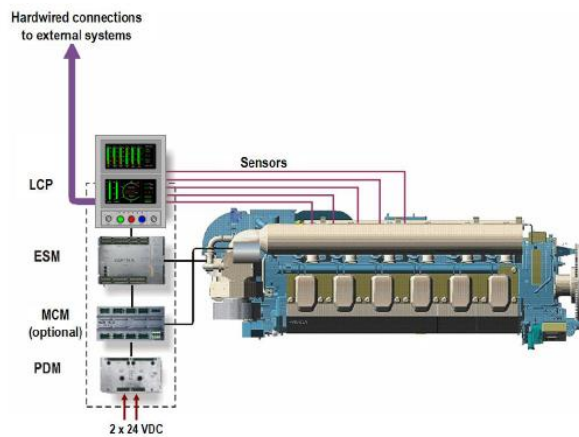


Figure 6. UNIC C1 configuration. [5]

2.2.2 UNIC C2

UNIC C2 is used for common diesel engines with engine management and utilizes modern bus technologies for safe transmission of sensor- and other signals.

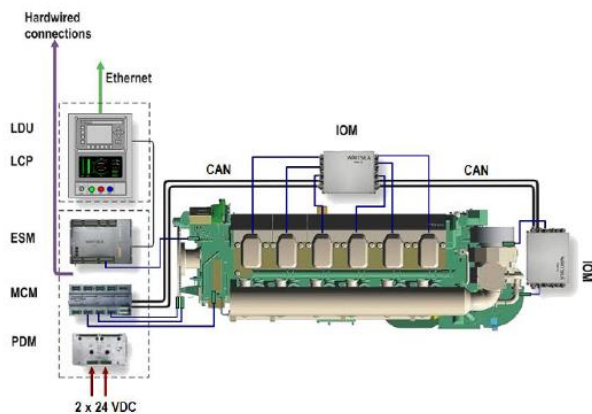


Figure 7. UNIC C2 configuration. [5]

2.2.3 UNIC C3

UNIC C3 is used for diesel, common rail, gas engines with electronic combustion control. The system uses CAN bus for communication between the modules.

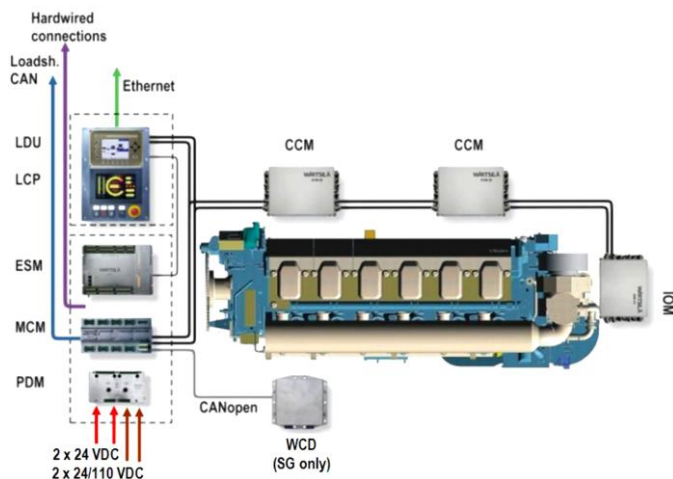


Figure 8. UNIC C3 configuration. [5]

2.2.4 UNIC Modules

The UNIC automation system comprises of different modules that all have their own tasks in the engine management process.

- ESM – Engine Safety Module

- The ESM module handles the fundamental safety features of the engine.
- MCM – Main Control Module
 - The Main Control Module handles the speed and load control of the engine, The MCM also manages all the other modules.
- PDM – Power Distribution Module
 - The Power Distribution Module protects all modules from disturbances in the incoming voltage.
- LCP – Local Control Panel
 - Control panel with backup instruments and for local operation.
- CCM – Cylinder Control Module
 - Handles cylinder specific control, such as knock detection and fuel injection. Every CCM module handles three cylinders.
 - It is to the CCM module that the knock signal will be replayed.
- LDU – Local Display Unit
 - Acts as an operator interface and as a communication interface to other networks
- IOM – Input Output Module
 - Universal, configurable data acquisition unit. It has a variety of flexible analogue and digital measuring channels for different applications.

2.2.5 WECSplorer

WECSplorer is used for tuning parameters, troubleshooting and for loading software into the UNIC engine control system. WECSplorer can also be used to trend process data, such as engine speeds, temperatures and pressures. [5]

2.3 Engine knock

Engine knock, or detonation, occurs when the temperature or pressure in the unburnt air/fuel mixture (end gases) exceeds a critical level, causing autoignition of the end gases. This produces a shock wave that generates a rapid increase in cylinder pressure.

The shock wave causes resonance in the cylinder at a characteristic frequency that is dependent of the cylinder dimensions and the temperatures in the combustion chamber. If heavy knock is left untreated, damage to pistons, rings and exhaust valves can be the result.

[6]

When the pressure wave interacts with the piston face, it acquires the action of a chisel that hits it, inflicting structural damage, rather than exerting pressure to push.

There are different scenarios from which engine knock originates. Among them are detonation and preignition.

The process as seen in *figure 9* is normal combustion, how the process should run when there are no interferences.

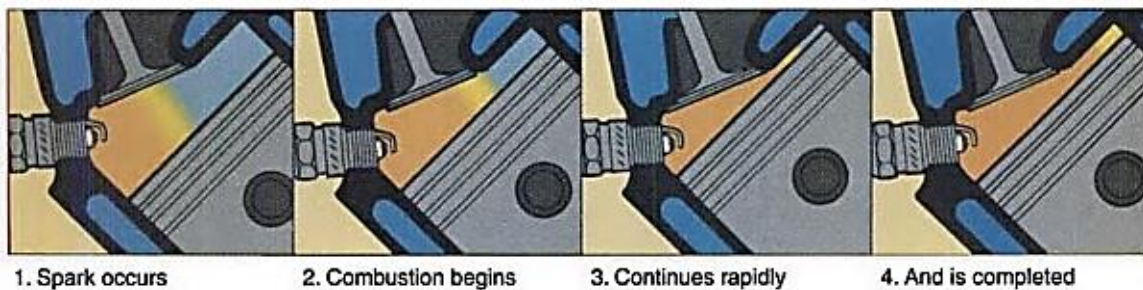


Figure 9. Illustration of normal combustion. [7]

When detonation occurs, as can be seen on *Figure 10*, a pocket of air-fuel mixture is ignited by the heat accumulated from the rising pressure in the combustion chamber, resulting in a secondary flame front which collides with the flame front generated by the spark plug. This collision results in high pressure peaks and dangerous vibrations that could both damage the engine if left untreated and affect the engine performance.

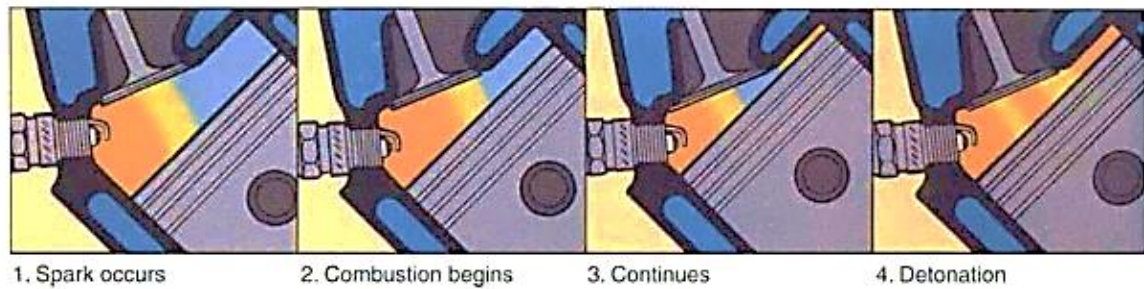


Figure 10. Illustration of detonation. [7]

As seen from *Figure 11*, when the air-fuel mixture is ignited before the spark plug ignites, is called preignition. Any hotspot in the combustion chamber can result in preignition. Preignition usually leads to detonation, although they are two different phenomena's. [7] [2]

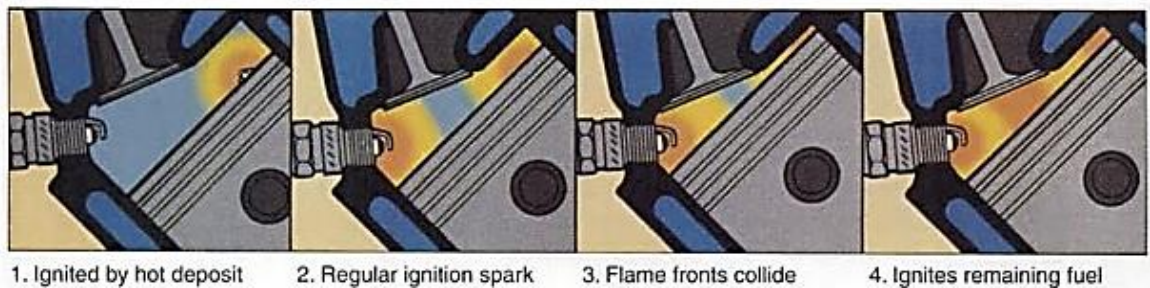


Figure 11. Illustration of preignition. [7]

2.3.1 Causes

There are many possible causes for engine knock:

- Carbon deposits form a heat barrier and can be a contributing factor to pre-ignition.
- High compression of air-fuel mixture.
- An overheated spark plug (too hot a heat range for the application).
- Glowing carbon deposits on for example a hot exhaust valve (which may mean the valve is running too hot due to poor seating, a weak valve spring or insufficient valve lash).
- A lean fuel mixture.
- An engine that is running with a higher working temperature than normal due to a cooling system problem.

- Self-ignition of engine oil droplets.
- Insufficient oil in the engine.
- Ignition timing too far advanced. [8]

2.3.2 Effects

Knocking affects the engine in many ways. Concluding both mechanical wear and actual damage on different parts on the engine if left untreated.

Engine efficiency is also directly affected by knocking. When an engine under normal running conditions is compared with a heavily knocking engine, the result shows that engine efficiency drops by around 13 %.

When engine knock is present, the cylinder pressure is about twice the pressure at the beginning of normal combustion. The same applies for absolute temperature. This leads to the possibility of engine failure will grow higher.

However, this is highly unlikely, because the anti-polishing ring (APR) and the pistons are made of different materials with different thermal expansion coefficients. Different materials that have different coefficients can result in that the piston top will expand more than the APR during knocking. The occurrence of this is very improbable, as the temperature difference would have to be in the range of 500 °C.

Engine parts may suffer from mechanical deformation due to knocking, because of the increased pressures inside the cylinder chamber. The compression and wear on engine components due to knocking all affect the engine life-span in adverse ways. [9]

2.4 Knock detection

There are two major techniques to detect engine knock. Those are to utilize pressure sensors and vibration sensors. [6]

2.4.1 Pressure sensor

Pressure sensors measure the pressure inside the combustion chamber of an engine. The signal provided by the pressure sensor offers the best signal to analyse to detect engine

knock. This kind of sensor is used on some Wärtsilä engines. But every cylinder needs to have its own sensor and the cost of the sensors is high. [9]

2.4.2 Piezo Knock sensor

The piezoelectric knock sensor measures the vibrations in the engine that occur at uncontrolled combustion. Because of the forces produced by the vibrations in the engine the piezo-elements in the sensor generates a voltage that can be measured. [6]



Figure 12. Bosch knock sensor. [9]

During combustion, engine knocking sends vibrations to the silicon rings attached to the piezoelectric crystals in the knock sensor (in the form of mechanical stress), accelerating the silicon ring, forcing the sensor to generate an electrical voltage.

A typical voltage signal generated by the knock sensor can range between 300 mV to approximately 500 mV, depending on the intensity of the knocking. When there is no knock present, the sensor will output a voltage level due to the normal vibrations of the engine.

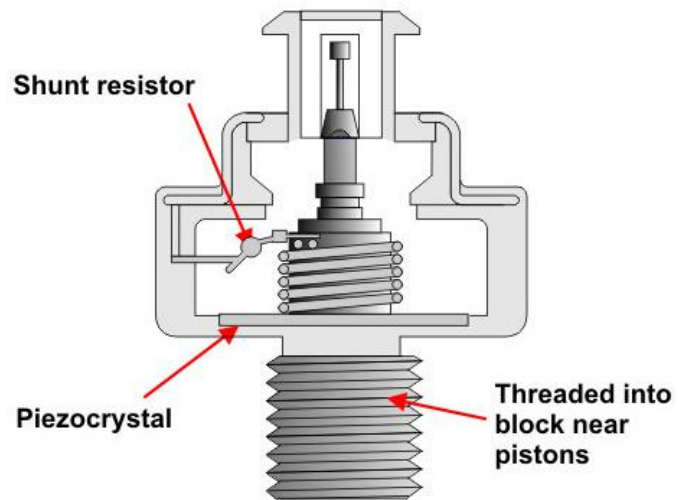


Figure 13. Construction of a knock sensor. [10]

2.4.3 Piezoelectric effect

The word "piezo" is derived from the Greek word for pressure. In 1880 Jacques and Pierre Curie discovered that pressure generates electrical charges in a number of crystals such as quartz and tourmaline. They called this phenomenon the "piezoelectric effect". Later they noticed that electrical fields can deform piezoelectric materials. This effect is called the "inverse piezoelectric effect".

Pressure generates charges on the surface of piezoelectric materials. This **direct piezoelectric effect**, also called generator or sensor effect, converts mechanical energy into electrical energy.

Vice versa, the **inverse piezoelectric effect** causes a change in length in this type of materials when an electrical voltage is applied. This actuator effect converts electrical energy into mechanical energy. This effect is frequently also called the actuator or motor effect.

Today the piezoelectric effect is used in many everyday products such as lighters, loudspeakers and signal transducers. Piezo technology is also being used in automotive technology, because piezo-controlled injection valves in combustion engines reduce the transition times and significantly improve the smoothness and exhaust gas quality. [11]

2.5 Speed and position measurement

Engine speed is one of the most critical measurements on the engine. The measured engine speed is, among many other things, used for speed control, i.e. control of the fuel demand in order to achieve a stable control of the engine speed.

Since an engine working cycle consists of two flywheel revolutions, speed sensors mounted on the flywheel cannot alone be used to measure the engine position. Therefore, phase sensors are also used to measure the engine phase using a half moon shaped disc mounted at the end of the camshaft.

In the cylinder control modules both speed and phase are needed to calculate the injection timing. Therefore, the speed sensors are connected both to the main control- and the cylinder control modules, and the phase sensors are connected only to the cylinder control modules.

For redundancy reasons two speed sensors and two phase sensors are connected to each cylinder control module. *Figure 14* shows the hardwired speed/phase signal distribution.

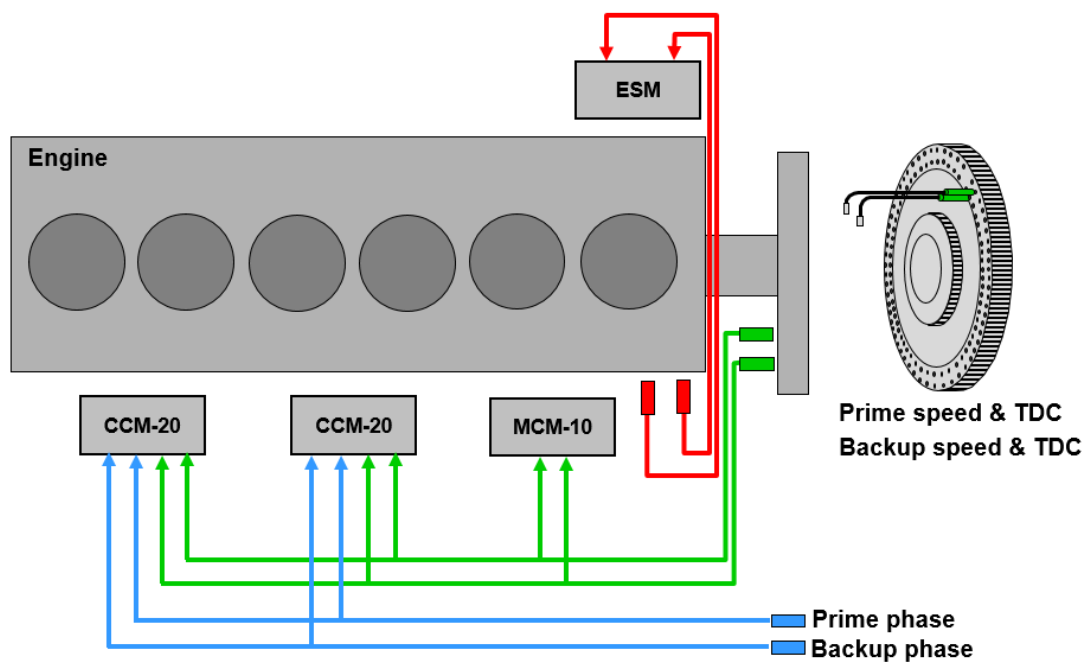


Figure 14. Speed & position measurement overview. [5]

2.5.1 Position measurement

As UNIC C3 must detect the accurate engine angular position, one missing hole is arranged in both speed sensing hole-peripheries on the flywheel, i.e. the pulse train will contain one missing pulse for each engine revolution. The angular locations of the missing holes are such, that the end-edge (= positive electrical flank) of the hole coming after the missing hole, is accurately at TDC (Top Dead Centre) of cylinder (A)1. The speed sensors use separate holes, but the holes are “in parallel”, thus the phase difference between the two signals is negligible. The number of holes is 120 minus the missing one, i.e. $120 - 1$.

The engine speed is measured by dividing the number of holes or teeth on the flywheel into equivalent segments of a certain length. The time of rotation is measured for each segment and because the angle of rotation is known the engine speed can be calculated.

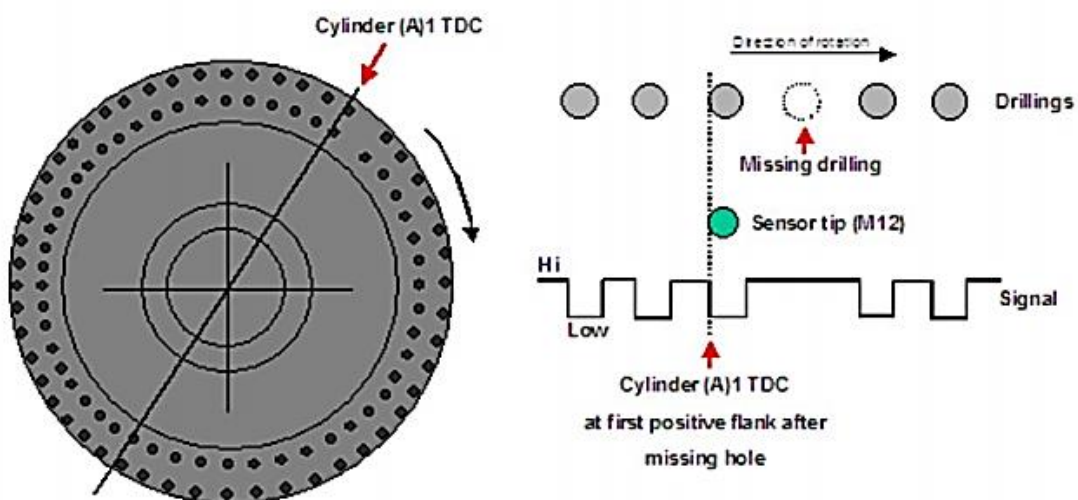


Figure 15. Missing hole measurement. [13]

2.5.2 Phase measurement

On 4-stroke engines, the crankshaft and thereby flywheel will make two revolutions for one complete engine cycle. To detect which TDC marker signal (missing pulse) belongs to the working phase of cylinder A (1), engine phase detection is also needed. Two phase sensors are provided of redundancy reasons. These sensors are mounted at the driving end of the camshaft of the engine. These sensors are PNP-type proximity switches.

The phase sensors are detecting the “phase” of the engine by means of detecting the position of a “half-moon” disc, attached to the driving end of the camshaft. This disc is mounted in such a way, that a positive edge (signal going high) will occur 180° BTDC of cyl. (A)1, and will remain high until 180° ATDC for the same cylinder, see *Figure 16*. Based on whether the phase signal is high (24 VDC) or low (0 VDC) when the missing pulse occurs, UNIC C3 knows that the engine is at A1 TDC. Based on the state of the phase signal at A1 TDC, UNIC C3 can determine which phase the crankshaft is in.

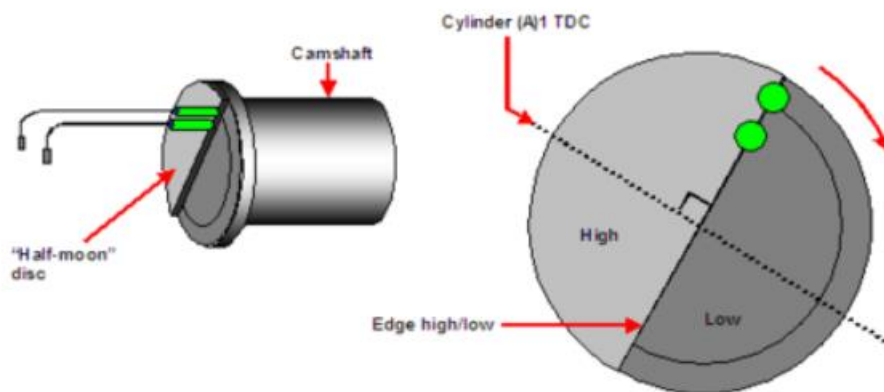


Figure 16. Half-moon disc. [13]

2.6 Measurement equipment

The equipment used in this thesis, both hardware and software, were delivered by Dewesoft, a manufacturer of measurement equipment from Slovenia.

2.6.1 Hardware

SIRIUS® is a data acquisition device, which offers high flexibility for inputs like voltage, current, temperature, strain, vibration, pressure, counters and CAN. The data-transfer to the PC can be done via USB or EtherCAT®. [14]

There is a wide range of applications for which the Sirius can be used;

- Automotive
 - On-road testing
 - Durability testing
 - Power train and E-mobility
 - Extreme environment testing
- Power & Energy
 - Power analysis
 - Power quality analysis
 - Power system testing
- Dynamic signal Analysis
 - Noise & Vibration
 - Rotating machinery
 - Structural analysis
 - Acoustic analysis

2.7 Software

Dewesoft is a measurement software which can acquire data from many different kind of measurement hardware and enables the user to do processing, storage and analysis in a simple way.

Dewesoft has two main versions of the software, Dewesoft 7 and Dewesoft X2. Dewesoft 7 is to be used with third party hardware and Dewesoft X2 is to be used with Dewesoft products. [15]

2.7.1 Counters

Counters are mainly used for measuring RPM and angle of rotating machines.

Typical applications are:

- Event counting (basic, gated, up/down, basic encoder)
- Sensor (encoder, tacho, CDM, 60-2...)
- Waveform timing (period, pulse-width, duty cycle) [16]

2.7.2 Analog out

There are several ways to use the analogue outputs;

- File replay
 - Outputs measured data directly to an analog output.
- Analog/Digital Output
 - Data channel or single value may be assigned to an analog output.
- Function generator
 - Can be set to Signal Output, Trigger Output, Frequency Output and Channel Output.
 - Signal output produces a specific waveform with adjustable amplitude, offset, frequency, and phase.

- Trigger output, outputs a trigger signal during measurement.
- Frequency output, outputs a frequency signal.
- Channel output may be set to any data channel to be outputted to an analog output. [17]

2.7.3 Combustion analysis

The Combustion Analysis system from Dewesoft is used for engine research, development and optimization. The system can also be used for development and testing of engine components, actuators and auxiliary equipment.

Measurements in both the angle and time domain are supported, and with algorithms for online or offline mathematics and statistics – calculating heat release and other thermodynamic parameters.

In this thesis, the Combustion Analysis module was used for recording of the knock sensor signals, as they can be added as “additional channels” in the engine setup. The module also records crucial data for analysing engine performance, if the data collected would be to be used for other purposes also. [18]

The setup of the CA module is divided into six sections:

- **Engine:** Defines the geometry and assigns the channels to the cylinders
- **Angle Sensor:** Assigns the angle sensor, sampling type and the TDC detection
- **Calculations:** Setup of the basic statistics and the pressure correction principles
- **Thermodynamics:** Setup mainly for the thermodynamic calculations
- **Knock detection:** Configuration of the knock detection algorithms
- **Outputs:** Enables/disables the output channels for the CA module

Engine setup

Fuel type defines the fuel of the engine. Depending on the selected fuel type a polytrophic exponent used for thermodynamic calculations is suggested. The defined value must be entered manually into the polytrophic exponent field.

Start of combustion (SOC) and end of combustion (EOC) are provided as results. EOC is defined where integrated heat release reaches 95 %, which is valid for diesel and gasoline fuel types. With gasoline, SOC is defined when the integrated heat release reaches 5 %, and with diesel, SOC is defined when the integrated heat release crosses 0 % (Due to the injection of diesel fuel the integrated heat release goes negative first). [18]

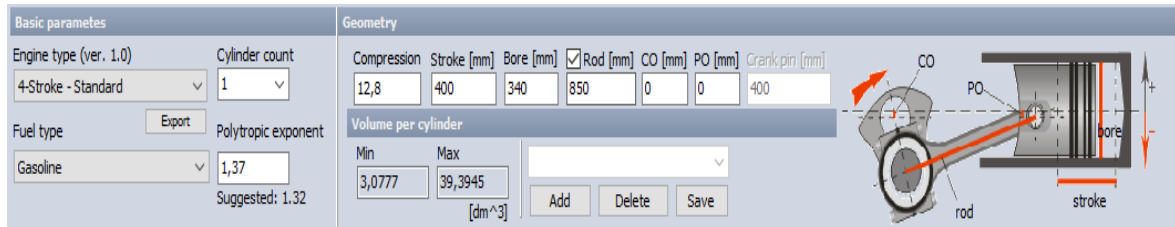


Figure 17. Engine setup.

For each cylinder the corresponding pressure channel needs to be assigned from the channel input list. Also, the ignition misalignment relative to the reference cylinder needs to be entered in degrees. The reference cylinder is indicated with a piston, and could be applied to any cylinder.

The knock sensor channels are added to the module as additional channels, SE6014 indicated that it is cylinder 1A. Cyl1 ACC is an additional accelerometer added to be used as reference. [18]

Cylinders	
Cylinder	Ref. cyl. 1
Pressure channel	Pcyl1
Ignition misalig. [°CA]	0,000
SOI/EOI channel	<Unassigned>
No. of injections	0
SOI trigger level	0
EOI trigger level	0
Additional channels	SE6014
Additional channels	SE6024
Additional channels	SE6034
Additional channels	SE6044
Additional channels	SE6054
Additional channels	SE6064
Additional channels	Cyl1 ACC
Additional channels	<Unassigned>

Figure 18. Engine setup channels.

Angle sensor

The counter channel for the speed signal must be assigned, and what type of sensor is used. On Wärtsilä engines the speed signal is a so called 120-1 Pulse train, i.e. 119 pulses with one missing pulse.

Sensor and sampling						
Sampling	Angle sensor type: Miss teeth-119-1	Connected to	Resolution	Upper frequency [RPM]	Resampling type	
Time domain Ext. clk. not supported	120-1	CNT B-4	0,2 deg.; 1800 p/rev	833	Unfiltered	
Top dead center location		Top dead center detection				
Trigger offset	No. of cycles	Thermo. loss angle	Cylinder for TDC	TDC detection		
3,000 [°CA]	120	0,7	All	Start	Cylinder must not be fired for TDC detection!	

Figure 19. Angle sensor setup.

2.7.4 Sequencer

A sequence is an ordered list of procedural commands which will be executed one after another. It contains building blocks (also called elements).

The execution flow starts with the block marked with a grey dot and follows the path defined by the arrow lines. The same elements can appear multiple times at different positions in the sequence. [19]

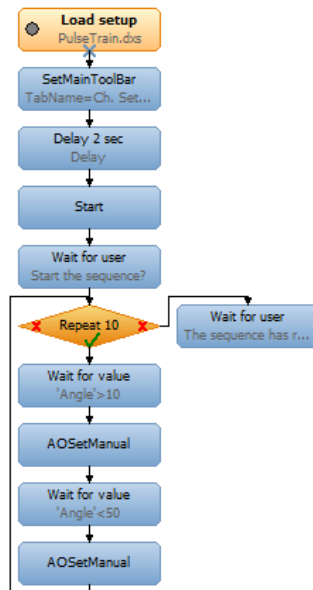


Figure 20. Sequencer.

There are different kinds of blocks with different functionality to be used in the sequencer;

- **Action block**
 - When the sequence needs to perform an action, such as setting the screen to a different mode or activating some settings, an action block is inserted into the sequence.
- **IF block**
 - The IF block is used for making decisions. The decision is based on user choice where the question is asked in the Text info and the user answers with Yes or No. Another option is to base the statement on a value. The statements are built by comparing two values, variables or manually entered values, with

normal logic (>, <, = and !=). Multiple comparisons can be nested together to make logical statements.

- **Repetition block**

- The block is used to repeat a certain action a certain number of times.

- **Delay block**

- The block is used to wait for a certain amount of time defined in the delay field. Time is defined in seconds.

- **Wait block**

- The block is used to wait for a certain event. It can be used to wait for user interaction or to wait for a value.

- **Audio Video block**

- This block is used to play audio, video files, slideshows, or output text-to-speech. Supported file formats are .avi, .mkv, .mov, mp4, mpg, wmv, mp3 or .wav.

- **Macro block**

- This block is used to record and replay a Macro. By starting the macro recording, DEWESoft will execute the sequence up to the point where Macro needs to be recorded. The mouse and keyboard can be operated normally and audio can also be recorded during the recording. When playing it back, DEWESoft will repeat the same actions.

- **Calculation block**

- This block is used to set variables or control channels. Also the data header values, internal variables used in math or control channels which operate analog or digital output.

- **Custom block**
 - This block is used as the black box which can contain more blocks to reduce the complexity of the main sequence.

- **File manager**
 - The file manager block is used to manage the data files, for example delete, copy or rename the files. [19]

2.8 Reference curves

Reference curves can be used to set a wanted level of a signal, and is often used with comparisons and as trigger criteria.

In this project reference curves were used to “import a signal”, as recorded data from another session was needed in the setup.

There are a set of different types of reference curves to be used. But as this project is only in the time domain, only time reference curves are used. [15]

Reference curves
Frequency domain ref. curve
Time reference curve
Vector reference curve
XY reference curve

Figure 21. Reference curves.

The values for time reference curves can be either entered manually or loaded from a data file. When the file is loaded, it is important that the sample rate of the setup being used is at least as high as the sample rate used in the file imported. [15]

Time reference curve setup

Input

Reference curve type: From data file

Start condition channel: <Unused>

Name: SE6014

Units: mV Color:

Max value: 1E004 mV

Min value: -1E004 mV

Templates: Save

Settings

File name: D:\DEWESoft\Data\Sirius_W6L34DF_Knock_25k.dxd ... Loaded

Start if value above: 0 Offset: 0

Name	Use
SE6014	Yes
SE6024	No
SE6034	No
SE6044	No
SE6054	No
SE6064	No
Cyl1 ACC	No
Pcyl1	Yes
CNT B-4/Angle	Yes
CNT B-4/Frequency	No
Engine_load	No
Engine_speed	No
FUPT101	No
SE6014	No
SE6024	No
SE6034	No
SE6044	No

Figure 22. The channels included in the loaded file.

3 Implementation

The implementation and testing part of this thesis has been done in different sections, and different approaches to the problem has been taken. Because there are multiple ways of solving the task, but they all have their pros and cons.

3.1 Test setup

To see how the engine control system would behave with the signal being replayed to it from the Dewewsoft, a small test rig was built.

The test rig consists of a MCM and a CCM module and a speed simulator. The speed simulator outputs a signal like the signal that the MCM gets from the speed sensor. To establish communication between the UNIC modules and the PC, a CAN interface was used.

In *figure 23* it can be seen how the analog output is connected to the CCM module with regular banana clips, as this is just a test setup.

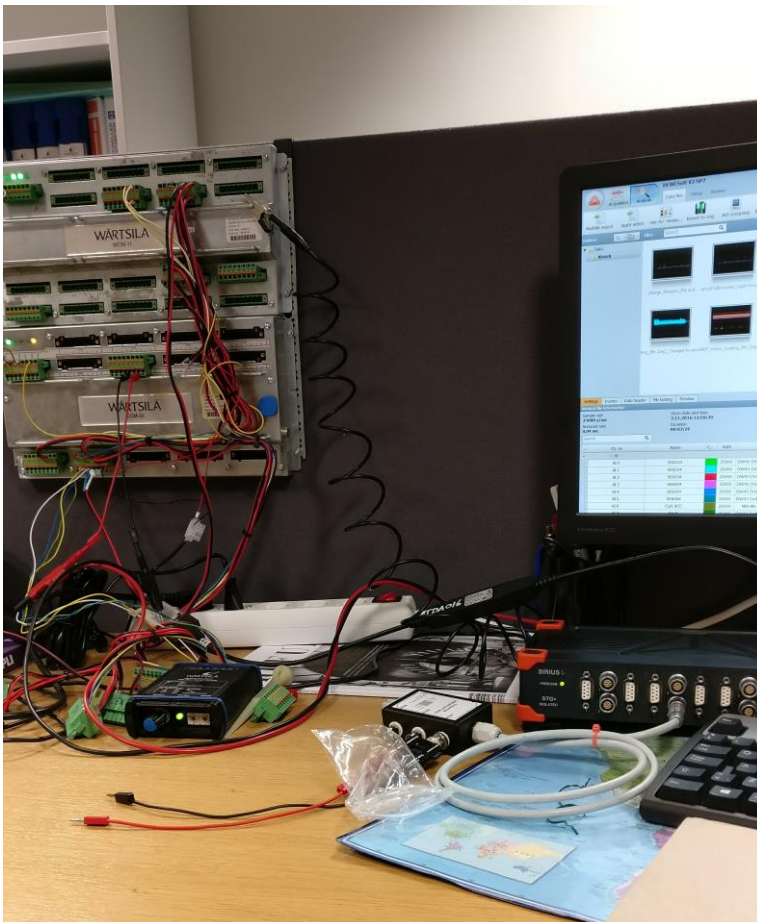


Figure 23. Test setup.

3.2 Recording

The recording of the knock sensor signals took place at the engine laboratory, on a W6L34DF, i.e. a six-cylinder engine with an inline configuration, 340 mm bore and can run on both gas and diesel fuel. The test data used in this thesis was pre-recorded during previous tests.

3.2.1 Combustion analysis

For recording, the CA module was used. The knock sensors from all cylinders were recorded in the same recording.

3.3 Reproduction

For reproduction, there were three different methods for outputting the signal; Function generator, A/D-channel and direct playback.

The tests showed that the function generator was the best alternative for this project, as with the direct playback there were no possibilities for synchronization, and the A/D-channel corrupted the signal so much.

3.3.1 Function generator

AO buffer length - when the Function generator function is used in Dewesoft, the software sends data to the DAQ. The DAQ then waits for the AO buffer to be filled (in this case, 1 second) before it starts to output the signal. This must be done, to prevent data loss. By default, the buffer length is set to 1 second.

⊙ Analog out	
AO buffer length Buffer for "analog out" channels	<input type="text" value="1"/> seconds
Fill samples when buffer is more than	<input type="text" value="50"/> % full.

Figure 24. Buffer settings for AO.

If the buffer is set to 2 seconds. The samples will be sent when the buffer is filled to the defined level (at 2-second buffer, 50 % means that it will wait for 1 second). By default, the value is set to 50 %.

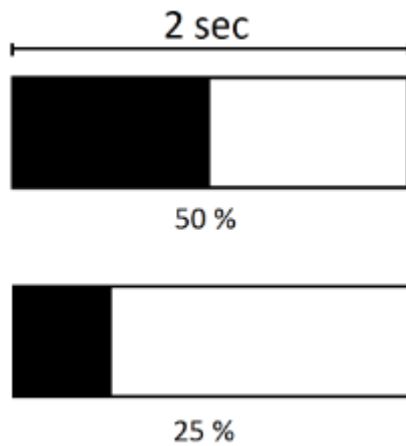


Figure 25. Buffer filled.

When the channel is assigned to the function generator, the signal is not altered as much as with the A/D channel. The problem with the function generator is that it is delayed in the beginning because of the buffer. *Figure 26* shows the signal being delayed by the size of the buffer. It might look like that the signal deviates from the original signal, but as auto-scaling is active, they look different.

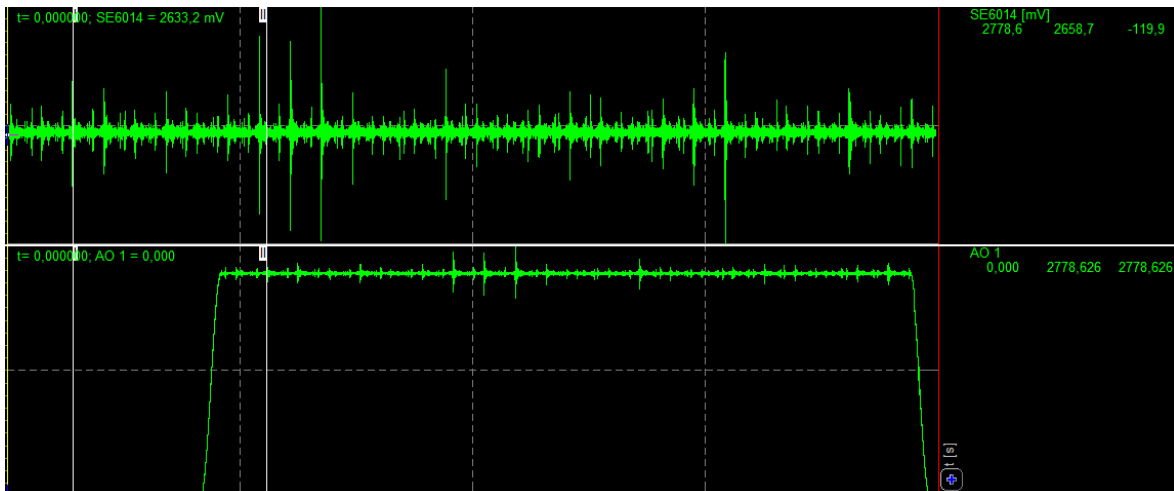


Figure 26. Function generator example.

3.3.2 A/D Channel

By using the A/D channel function, the synchronization would have been easier than with the function generator. But because the A/D channels are asynchronous, the signal gets corrupted too much to be used. When the channels are asynchronous, it means that the data points in the signal appears unevenly.

The A/D channels are mostly used for outputting a voltage level with no rapid changes, for example to drive a shaker or as a trigger signal.

The main reason why the A/D channels was not used, is that the method is too imprecise, as can be seen from *figures 27-29* as the dark green coloured signal trace.

When the period time is set to 1000 ms, or 1 s, the samples are too far apart. The result from that is a changing voltage level.

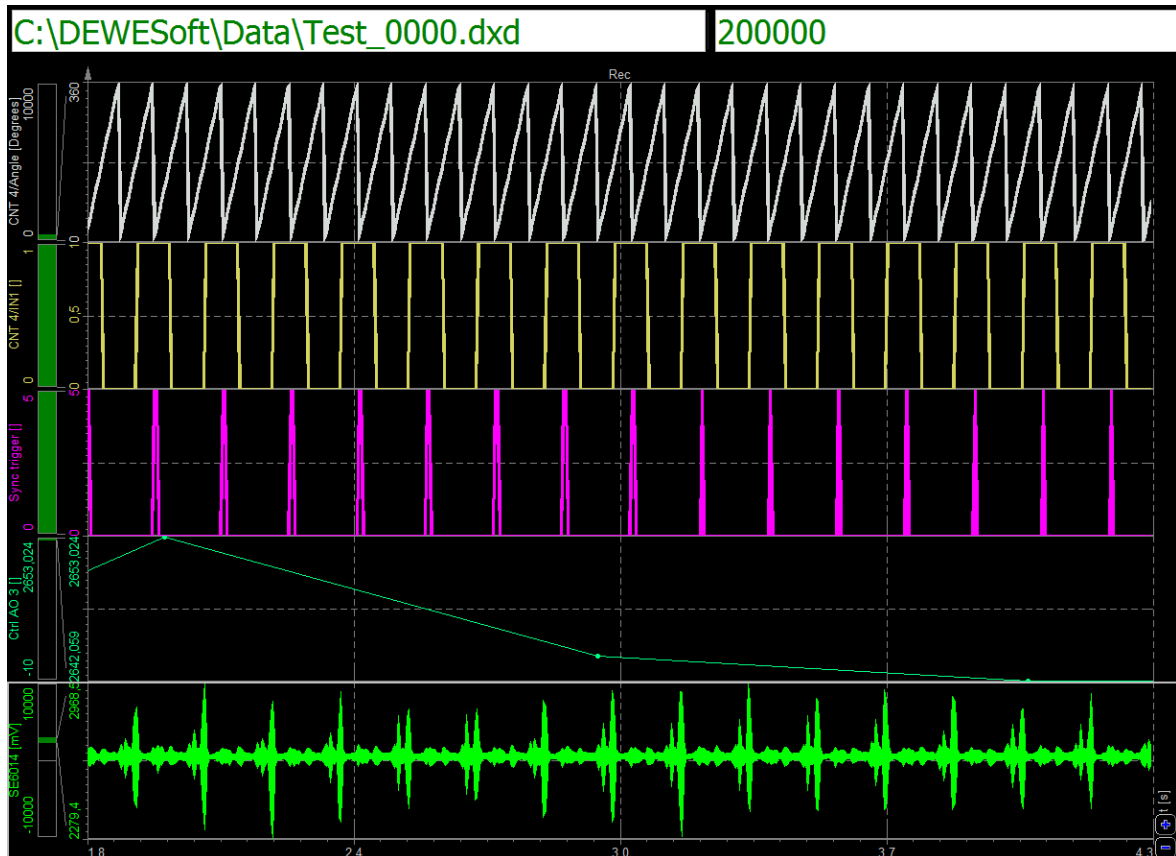


Figure 27. Period time set to 1000 ms (default).

If the period time is set to 1 ms, the samples are a bit closer together, but still not good enough, if compared with the main signal that is attempted to be reproduced.

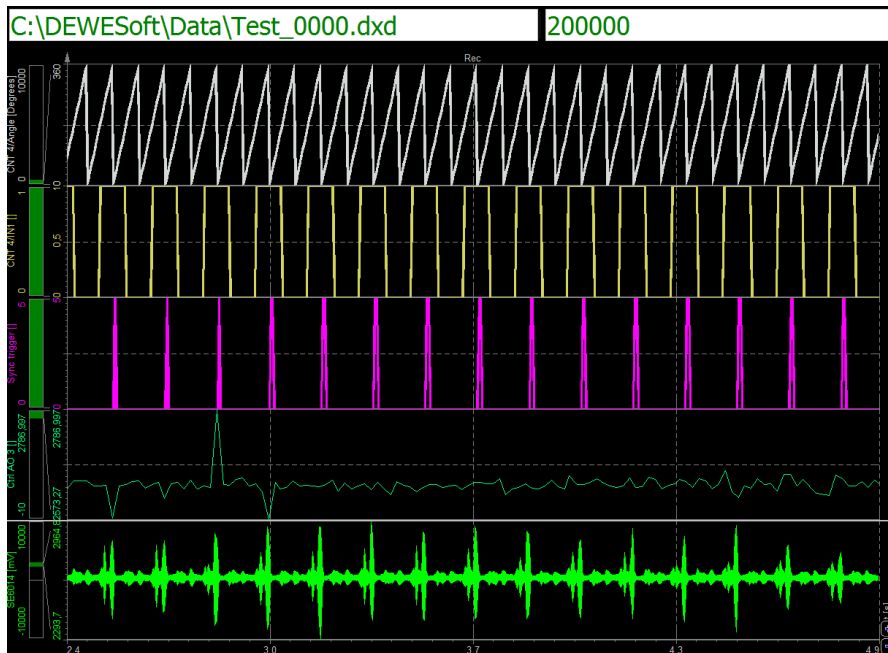


Figure 28. Period time set to 1 ms.

As can be seen when the period time is set extremely low, the result is the same as with the 1 ms period time.

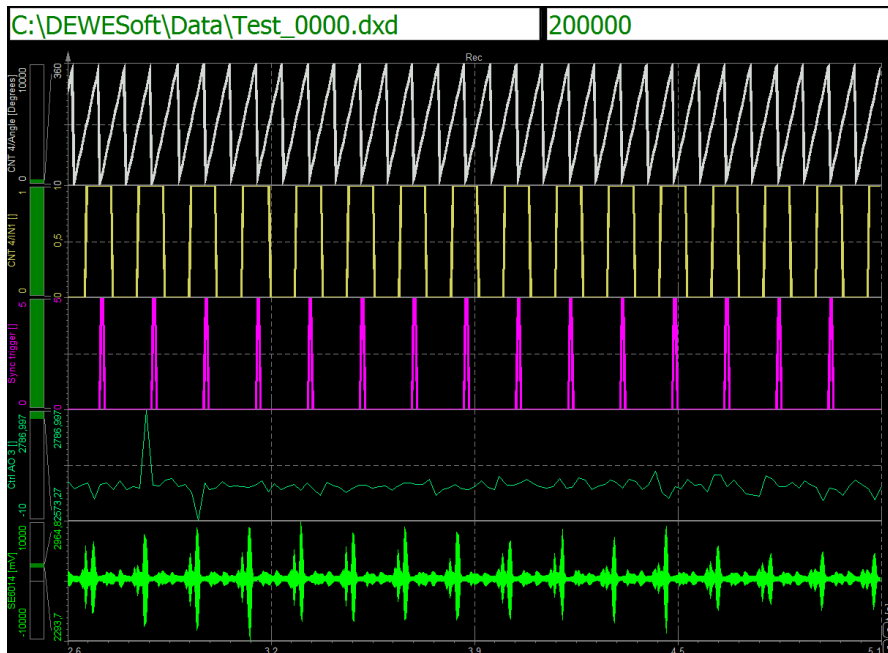


Figure 29. Period time set to 0,000001 ms.

3.4 Importing

As there will be used two different setups for recording and reproducing the signals, the wanted signals must be imported to the actual setup from the recorded data. The original idea was to declare the wanted signal to a global variable and read from the variable in the setup, but global variables can only be used to store a single value. The signals needed were imported as time reference curves

3.5 Synchronization

To only trigger the analog output to start outputting the signal at the right moment was not enough. If the recorded signal had a different speed than what the current running speed was, then the signals would start out looking good, but as time elapsed, they would start drifting apart from each other and become unsynchronized.

Therefore, a method to synchronize the signals in real time was needed. A few different methods were thought of:

- Starting and stopping the function generator manually.
- Adjusting the delay so the signals would stay coordinated.
- Remote controlling Dewesoft through MATLAB as a DCOM object, to read the signals to MATLAB, add padding samples to get the signals coordinated and send the data back to Dewesoft to be outputted.

Starting and stopping the function generator would cause an uneven signal that would not be of much use, and controlling Dewesoft through MATLAB would create unnecessary delays.

As different methods were investigated, Dewesoft released an update which included a new feature called Dynamic Delay Channel. The dynamic delay channel was finally used to manage the synchronization.

3.5.1 Math implementation

There is the functionality to use alarms as triggers in Dewesoft, but to set up the alarms, every alarm needs its own control channel, and the possibilities to customize the trigger conditions in the way needed was not the most sophisticated.

In this project, the triggers were built with math channels, as it provides more freedom to set the criteria's, than with the built-in trigger alarms.

Sync trigger – Is high when the angle on the counter channel (white) is greater than or is equal to 0° and below 50° and the Phase channels (yellow) is not equal to 0. The reason the phase is taken into consideration is to know if it is cylinder A1 or B1 that are doing the power stroke at the moment. The sync trigger can be seen as the purple coloured signal trace.

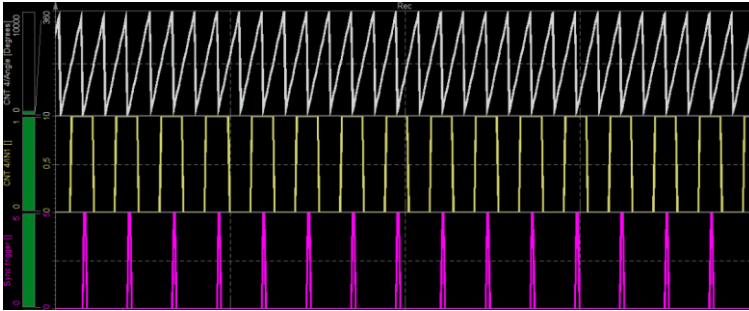


Figure 30. Synchronization trigger.

The trigger was built with an IF statement that checks if the angle value is between 0° and 50° , and that the phase signal is high. If those conditions are met, the signal goes high, else it outputs a 0.

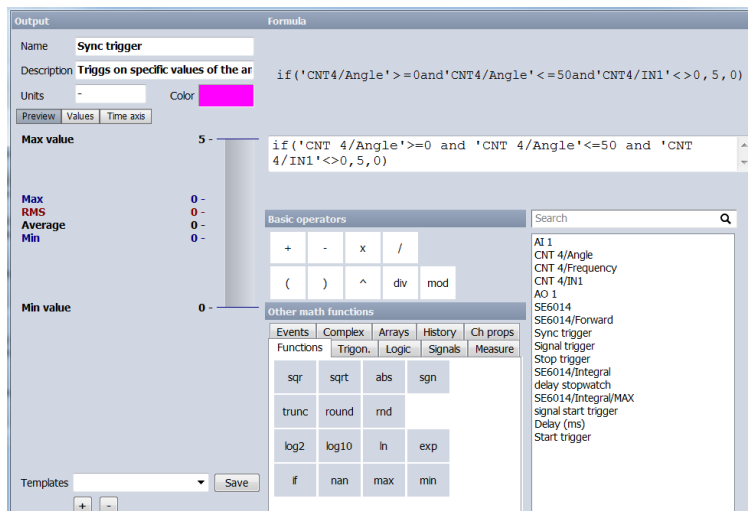


Figure 31. Synchronization trigger setup.

Signal trigger – To get the trigger value for the signal trigger, the integral of the outputted signal is used. When the signal reaches a limit value, the maximum value of the integral, a pulse is given.

As can be seen from *figure 32*, when the aquamarine coloured signal trace peaks, it is compared with the blue signal at the bottom. The blue signal at the bottom is the max value of the integral, and as it calculates the maximum value for set blocks at the time, then the peaks on the integral occur, it compares it with the previous value.

When the value of the integral is greater than or equal to the comparison value, it outputs a high signal, which can be seen as the light blue coloured signal trace on the top. It would be possible to use a filter directly on the knock signal, but as there are interferences, multiple triggers might occur for each cycle.

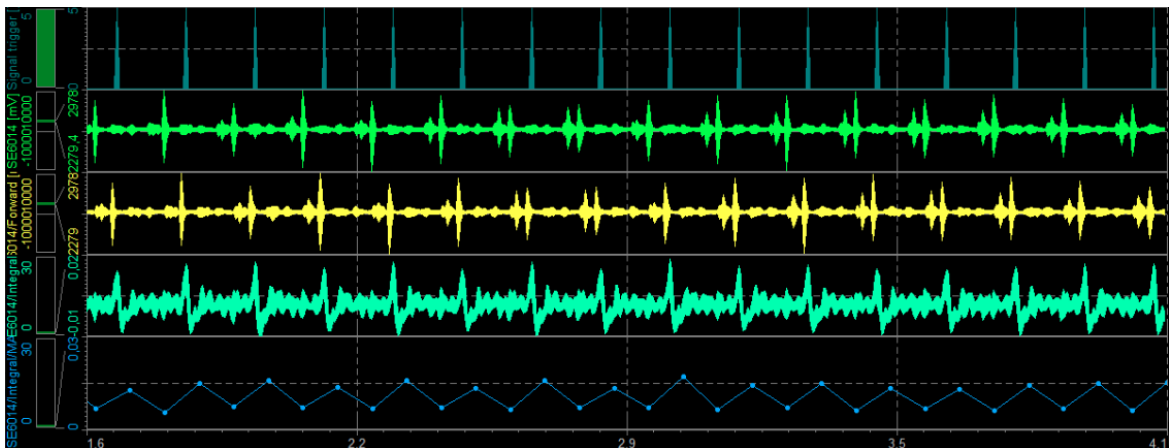


Figure 32. Signal trigger.

The two triggers are then put into a stopwatch function that calculates the time elapsed between the pulses, thus calculating the delay between the signal and the angle. The result is multiplied by 1000 because the result from the stopwatch is in seconds, and the input to the delay function is ms.

Used	Store	Formula	Setup
		stopwatch('Signal trigger','Sync trigger')*1000	
	delay stopwatch	0	1E4

Figure 33. Stopwatch function.

3.5.2 Delay

The delay math channel sports a feature that allows the delay to be dynamic, that it can change during measurement.

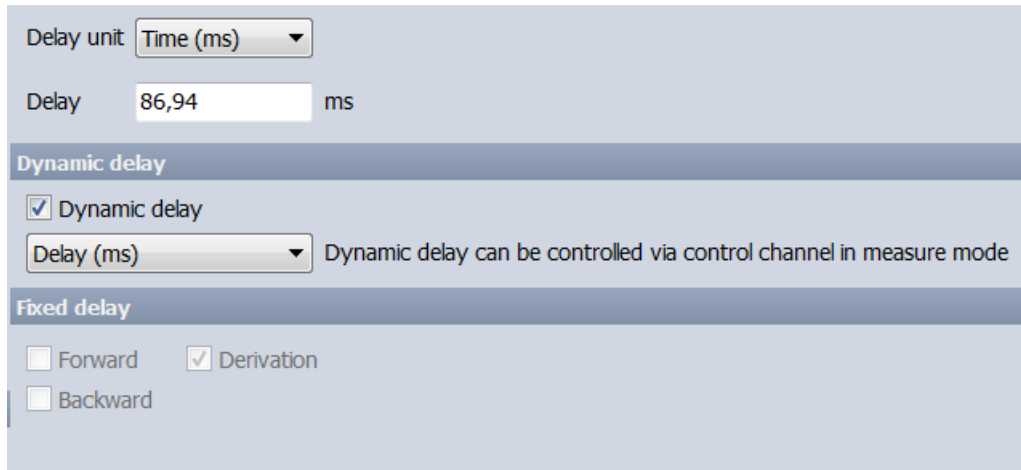


Figure 34. Dynamic Delay Channel setup.

The dynamic delay parameter is taken from a control channel, which can be defined by the user. In this case, a single value channel is used.

#	Used	C...	Name	Description	Timebase	Min	Max	Init value	Value reset	Value	Reset value
0	Used		Delay (ms)	-	Single value	0,00	200,00	42,04	On start measurement	42,04	Reset
1	Used		Start trigger	-	Async	0,00	100,00	0,00	On start measurement	0,00	Reset

Figure 35. User control channels.

3.5.3 Sequencer

To run this continuously, a short sequencer is built. The sequence is basically a loop, that for every iteration declares a new value to the single value control channel that controls the delay channel, and the value is the result of the stopwatch formula.

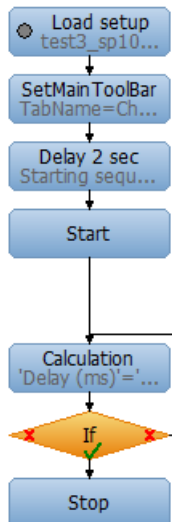


Figure 36. Sequencer used to update the delay in the dynamic delay channel.

The functionality and the setup of the blocks used in the sequence is as follows:

1. Loads the setup created

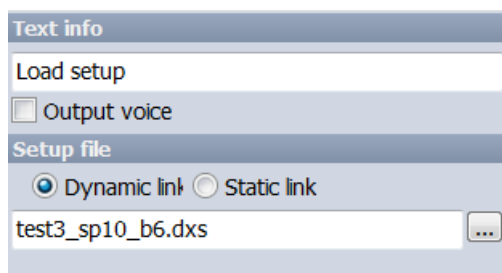


Figure 37. Load setup.

2. Sets what toolbars and menus are visible prior to start

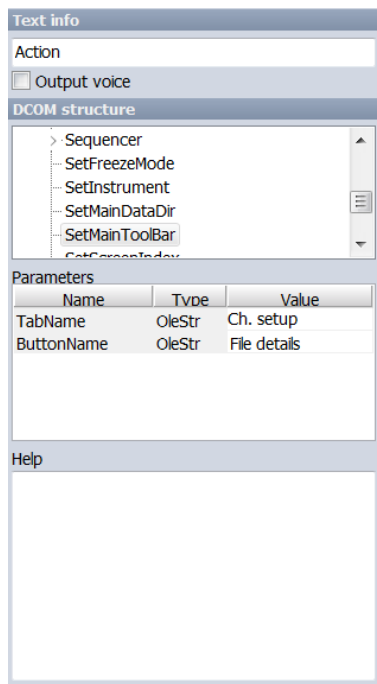


Figure 38. Set toolbars.

3. Waits for the set amount of time before starting the sequence, while waiting the screen shows the screens chosen in the previous step.

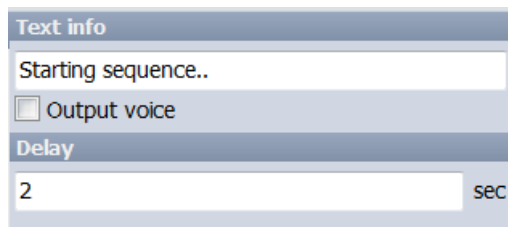


Figure 39. Delay start.

4. Starts the sequence

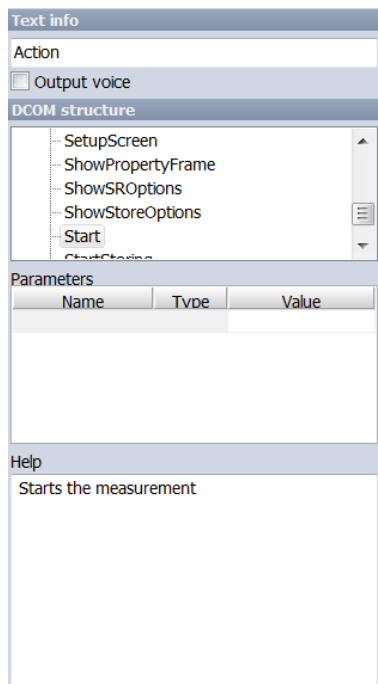


Figure 40. Start measurement.

5. The calculation block declares the value from the math channel 'delay stopwatch' to a control channel named 'Delay (ms)'. 'Delay (ms)' is used as input to the dynamic delay channel.

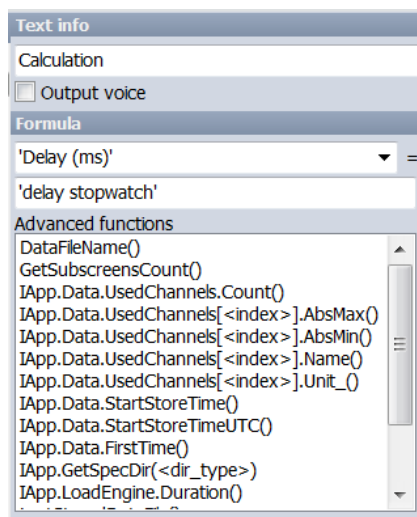


Figure 41. Calculation block.

6. Checks if a certain value is reached. In this case, it checks if a variable named 'Stop trigger' is equal to 3. The stop trigger is a math channel that counts the number of times the spacebar has been pressed. The reason for the need of three presses to quit the program is that you might press it once by mistake, but not three times.

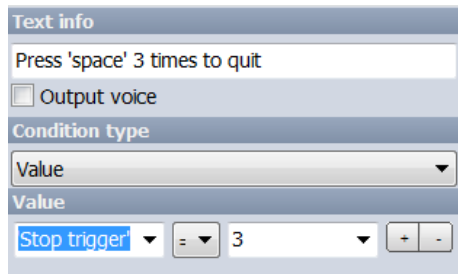


Figure 42. Stop trigger.

7. Stops the program.

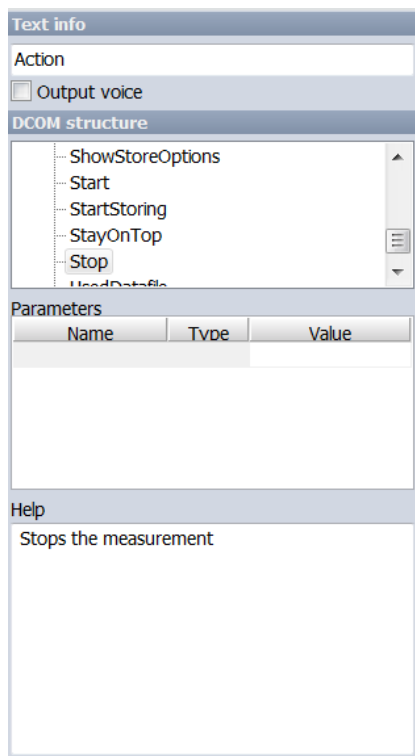


Figure 43. Stop measurement.

Figure 44 shows the signals without the dynamic delay channel. The traces as seen from the top to the bottom:

1. Crank angle, signal from the engine position sensor.
2. Engine phase, tells which cylinders are performing the work cycle at the moment.
3. Analog output, the signal assigned to the analog output channel, in this case the delayed knock sensor signal.
4. Synchronization trigger, the trigger shows at what position the analog output should be positioned.
5. Signal trigger, triggers at the signal peaks.
6. Knock sensor signal, the unaltered knock sensor signal.
7. Dynamic Delay Channel, the knock sensor signal with dynamic delay applied.
8. Filtered integral of the Dynamic Delay Channel, the signal trigger is based on this signal.
9. Max value of the integral, when the integral value is greater than the maximum value, the signal trigger is active.

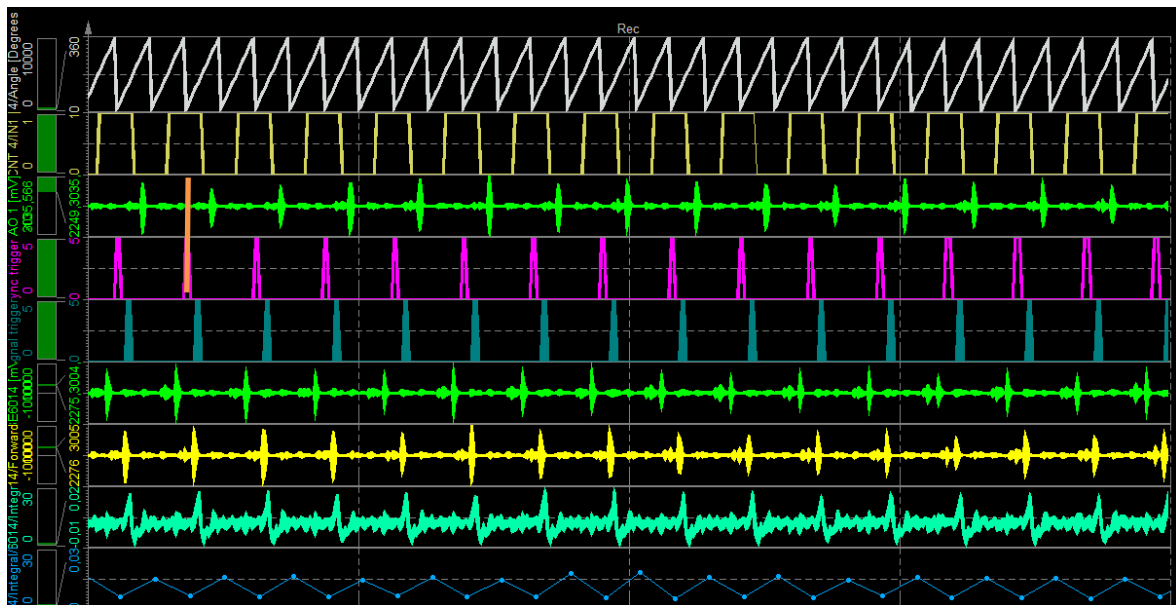


Figure 44. Signals without synchronization.

As can be seen from figure 44, the synchronization of the signals is not active on this test run. If the synchronization would be activated, the traces at rows 3 and 4 would chime in at the same positions, but in this case the peaks are not synced.

Figure 45 shows that the signals that have been synchronized with the synchronization sequencer. From the figure, it can be seen that the analog out signal and the synchronization trigger chime in at the same position, in other words, the signal is synchronized.

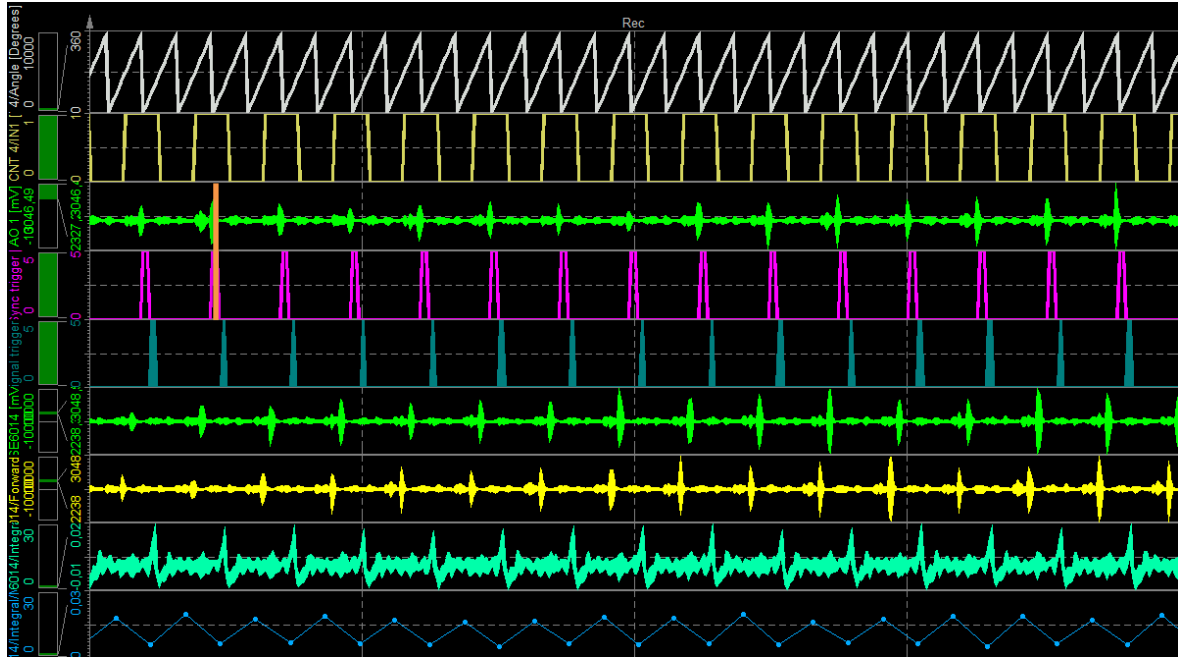


Figure 45. Signals synchronized.

4 Results

The road to the result was long and winding, as many different solutions were tried until the final solution was found.

The challenging part was to get the synchronization working. Many ideas about how to synchronize the imported signal were thought of, among others to shut off the function generator and turn it back on when the signal was to be outputted. However, the result was a lagging signal, with a lot of inactive time.

Another possibility was to create a link between Dewesoft and Matlab, in a Master-Slave like relationship. But it turned out to be more difficult than imagined, and with the additional delays that it caused, it would not be possible to use it as a Real-time application.

The final solution was to use a dynamic delay channel, which was not present in the beginning of this thesis, but was added later. The value for the dynamic delay is inherited from a stopwatch function that measures the delay between two triggers, and then a loop updates the delay value for every iteration.

The result of this thesis was a method to record and reproduce engine knock sensor signals. The knock sensor signals were recorded with the combustion analysis module and the knock sensors added as additional channels, in addition to the needed channels to perform the combustion analysis. Because of the separation of the setups, any recorded data could be used with the setup that was created for outputting and synchronizing signals. It is possible to use the setup with other sensors as well, if needed.

Additional sync conditions as cylinder pressure could be used for syncing the signals. But in this solution, only the engine speed and the phase were used for synchronization. One problem still lingers with the synchronization, when the sync sequence is started, the signals are synchronized, but they are out of phase, so the solution to this was to change the analog out buffer size to 1,01 s instead of 1,00 s. as the phase shift was 10 ms. It may be possible to tweak the setup to eliminate the need for the changing of the buffer size. As the situation is now, the buffer size needs to be changed if the signal to be outputted is changed, due to the differences in sample rate during the recording and the engine speed during recording and how much fluctuations there was in the engine speed.

As there are still many bugs in Dewesofts software, there was no possibility yet to test the final version of the project. Because the function generator, when set to channel output, there

was no signal outputted through the analog output even though the software showed that there was a signal present. If the function generator settings were changed to signal output and a triangular wave was outputted then it worked normally.

5 Discussion

This thesis was time consuming, as I had never worked with any Dewesoft products, hardware, or software. So, the first thing was to study how the device and the software worked, what possibilities it added.

This thesis was also multifaceted, as I needed both my knowledge as an automation engineer, but I also needed good knowledge about combustion engines, what happens in engines when they are knocking? What is an engine knock? So, I have spent a lot of time reading literature about the subjects, and I also attended a course about combustion engines so that I could gather as much knowledge as possible, and get a better understanding about the problem that was presented to me.

One major part of this thesis was to get the signals synchronized with the engine speed, and it proved to be one of the more challenging parts of the thesis.

Basically, this was more of a study if it is possible to use the measurement device from Dewesoft for such purposes. With the software version of Dewesoft X2 available when the work was initiated, this solution would not have been possible, due to the lack of functionality and some stability issues. But I have been in contact with a couple of support engineers at Dewesoft, and together we worked out how to tweak the existing functions to fit the needs of the project. One issue that remained was how to get the signals dynamically synced, as it was, the delays and trigger conditions needed to be manually entered.

Eventually, Dewesoft released a function called Dynamic Delay Channel, so that the delay needed, could be entered as a variable. Thanks to the helpfulness of the engineers at Dewesoft, a dynamic solution was found.

There have been a few other theses works done around the subject engine knock for Wärtsilä in the past, all of them building towards the same goal, more durable systems, and better knock control systems.

If I would start over from scratch now, I am not so sure I would use Dewesoft equipment. National Instruments equipment and software, LabVIEW is more sophisticated when it comes to real time handling of signals and data. But to use National Instruments hardware and software would mean a major investment, as the equipment is rather expensive and the software needs to be bought separately. With Dewesoft equipment, the software license comes bundled with the hardware.

5.1 Future development

This thesis is a part of the goal to get a knock sensor calibrator developed. A handheld shaker would simulate the engine, and the knock sensor would be attached to the shaker, and the signal from the knock sensor would be analysed, and the knock limits needed for the software would be derived from the signal. The method for determining the knock limits now, is that the knock sensor is attached to the engine, the engine is run so that it starts knocking, and the knock sensor values are then entered as knock limits. The procedure must be repeated for every cylinder, so it is rather time consuming.

The part about developing a tool for field use was left out of this thesis, but it may be possible to make it a topic for a thesis in the future. There are already some ideas circulating about how to develop a tool that is robust enough to withstand field use, that is small enough for the field service engineers to carry around and that is rather cheap to manufacture so it might even be mass produced.

There are a few possibilities, to use an already existing SOC (System on a chip) that is a credit card sized computer. There are for example the Raspberry Pi and the Beaglebone black. The positive aspects of the Raspberry Pi are that it is quite cheap, and there are a lot of already existing peripherals to add to it. The positive aspect about Beaglebone is that it has much more means of communication, as it has built in analog inputs, where the Raspberry only has digital inputs. Then only a DAC (Digital-to-analog) card would be needed.

The downside about the Raspberry Pi is that the operating system might not be reliable enough for such applications. And the downsides with the Beaglebone are that it is more expensive than the Raspberry Pi.

Another possibility would be to build it around an Atmel microprocessor. Then there would be much more room for customization and only the functions needed for the tool would be

added, so that the form factor would be kept smaller. The tool could also be configured in the future to be used on different kind of sensors, to get a universal sensor testing tool.

Another function that could be added to the tool would be a web interface, with a “measurement database”. Measurements taken in Vaasa could be stored on the database, and the signals from the measurement could be used in troubleshooting of a similar engine in for example Brazil.

6 Bibliography

- [1] Wärtsilä, "Wärtsilä," [Online]. Available: <http://www.wartsila.com/about>.
- [2] J. B. Heywood, Internal Combustion Engine Fundamentals, vol. 1, Massachusetts: McGraw-Hill Inc., 1988.
- [3] "Four-stroke engine," 2005. [Online]. Available: https://en.wikipedia.org/wiki/Four-stroke_engine. [Accessed 28 Februari 2017].
- [4] Tech. rep. Wärtsilä Finland, "Product Review W34SGD," [Online]. Available: DBAD273877 (Wärtsilä IDM).. [Accessed 4 Mars 2017].
- [5] Tech. rep. Wärtsilä, "Unic presentation," [Online]. Available: DBAA001148 (Wärtsilä IDM).
- [6] T. G. Horner, "Engine Knock Detection Using Spectral Analysis Techniques With a TMS320 DSP," Augusti 1995. [Online]. Available: <http://www.ti.com/lit/an/spra039/spra039.pdf>. [Accessed 4 Mars 2017].
- [7] J. Erjavec, Automotive Technology: A Systems Approach, 4 ed., vol. 4, Clifton Park, NY: Thomson Delmar Learning, 2005.
- [8] D. Hall, Automotive Engineering, Global Media, 2011.
- [9] J. Holtti, "Tuning light knock limits on SG engines. Bachelor's thesis in Automation Technology," Novia University of Applied Sciences, Vasa, 2012.
- [10] A. E. Schwaller, Total Automotive Technology, New York: Thomson/Delmar Learning, 2004.
- [11] PI Piezo Technology, "Fundamentals of Piezo Technology," [Online]. Available: <https://www.piceramic.com/en/piezo-technology/fundamentals/#c2232>. [Accessed 4 Mars 2017].
- [12] T. Kaas, "Knock control," 21 November 2012. [Online]. Available: Wärtsilä IDM (DAAB102405). [Accessed 7 Mars 2017].
- [13] Wärtsilä, "Wärtsilä InfoBoard," [Online]. Available: <http://infoboard.wartsila.com:8080/infoboard/main.jsp?docid=17664>. [Accessed 12 Mars 2017].
- [14] Dewesoft, "Sirius Technical Reference Manual," Dewesoft, Kumberg, 2016.

- [15 Dewesoft, "DEWESoft 7.0.3 User Manual," Dewesoft, Kumberg, 2011.
]
- [16 Dewesoft, "Digital Counters," Dewesoft, U.Å.. [Online]. Available:
] <https://www.dewesoft.com/pro/course/digital-counters-13>. [Accessed 7 Mars 2017].
- [17 Dewesoft, "Analog out configurations," 16 April 2013. [Online]. Available:
] https://www.dewesoft.com/download?file=Analog_out_Configurations.pdf.
[Accessed 7 Mars 2017].
- [18 Dewesoft, "Combustion Analyser Manual," 1 December 2016. [Online].
] Available:
https://www.dewesoft.com/download?file=CombustionAnalyser_ManualV1.1.0.pdf. [Accessed 7 Mars 2017].
- [19 Dewesoft, "Sequencer User Manual," Oktober 2010. [Online]. Available:
] https://www.dewesoft.com/download?file=DEWESoft%20-%20Sequencer%20Manual_V1.pdf. [Accessed 4 Mars 2017].