Gezamenlijke opleiding UHasselt en KU Leuven

2018•2019
Faculteit Industriële ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterthesis

programming a motor testbench with a movi-c controller

PROMOTOR :
ing. Geert LEEN

PROMOTOR :
Dhr. Marc GRAAT

Hendrik Vandervelden

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie, afstudeerrichting automatisering

2018•2019

Faculteit Industriële ingenieurswetenschappen

master in de industriële wetenschappen: energie

Masterthesis

programming a motor testbench with a movi-c controller

PROMOTOR :

ing. Geert LEEN

PROMOTOR :

Dhr. Marc GRAAT

Hendrik Vandervelden

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie, afstudeerrichting automatisering

PREFACE

I began my studies industrial engineering in 2013 at the UHasselt in close collaboration with the KULeuven. In the academic year 2018-2019, I needed a master thesis to finish my master of energy engineering. Vanderlande gave me this opportunity by letting me finish the motor testbench student project that they started in 2015.

I have enjoyed my time at the R&D department of Vanderlande and would like to thank everybody there. I especially want to thank Ing. Marc Graat who trusted me to fulfil this assignment and for being my supervisor, Rolf van Bree for giving me an extra hand when needed and enduring the noise that I made with the testbench and a last special thank you to Christoph Jannis for giving me some new ideas from time to time.

I would also like to thank Ing. Geert Leen and Prof. dr. Jeroen Lievens of KULeuven who have supervised me to make this paper and therefore to help me graduate.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

CFast card	CompactFast card, a variant of CompactFlash cards with higher transfer ratio.
Codesys	Manufacturer-independent development environment for PLC
DOL	Direct-On-Line, coupling the motor directly to the electric grid
Ethercat	Ethernet based fieldbus for automation control
Movisuite	A software program for the MOVI-C drive to set parameters
POU	Program Organization Unit. This can be a program, a function or a function block
PWM	Pulse-Width-Modulation is a technique to modulate a given DC voltage to a lower voltage of a chosen form.
RS-485	A serial communication standard that uses two wires
Stray load loss	A loss in energy that varies with the load and is not accurately determinable.

ABSTRACT

Vanderlande in Veghel uses electric motors for the intern logistic transportation systems that they develop. They need a motor testbench to increase the certainty of a motor choice and to verify the motor characteristics of the manufacturer. This thesis continues to develop on the mechanical design of the motor testbench and describes the further development needed to reach a functional testbench. The mechanical design contains the motor couplings that connect the motor-on-test with the torque sensor and an adjustable load

Firstly, the mechanical design was completed with the necessary electrical components. Mainly these are measuring sensors and the load. After which the testbench was assembled. Next came the decision on and the implementation of the test protocols. To achieve this, a PLC program was developed to control the motors and obtain all necessary data, as well as an Excel program to visualize these data. Finally, a manual was written to explain the start-up of every test.

Today Vanderlande has a motor testbench that can be used for four separate tests. The first test determines the slip, power factor, current, power and efficiency of a motor. The second test can determine the torque-speed characteristics of a motor. With the third test the efficiency of the motor or the efficiency of inverter with motor can be measured when the motor is driven by an inverter. The final test is an endurance test where the motor accelerates and decelerates.

ABSTRACT IN DUTCH

Vanderlande in Veghel maakt gebruik van elektromotoren voor de interne logistieke transportsystemen die het ontwikkelt. Zij hebben een motortestbank nodig om de zekerheid van een correcte motorkeuze te verhogen en de motorkarakteristieken van een fabrikant na te gaan. Deze masterproef gaat verder op het mechanisch ontwerp van deze motortestbank en beschrijft het verdere ontwikkelingsproces tot een functionele testbank. Het mechanisch ontwerp bestaat uit de motor koppelingen die de te testen motor verbind met de koppel sensor en een regelbare last.

Als eerste werd het mechanisch ontwerp aangevuld met de benodigde elektrische componenten. Dit is voornamelijk meetsensoren en de last. Vervolgens werd de testbank opgebouwd. Als derde moesten de testprotocollen gekozen en bepaald worden. Nadien is er een PLC programma ontwikkeld dat de motoren aanstuurt om alle benodigde gegevens te verkrijgen en een Excel programma dat deze gegevens visualiseert. Als laatste is er een handleiding geschreven die uitlegt hoe elke test opgestart moet worden.

Vandaag beschikt Vanderlande over een motortestbank waarbij de motor kan getest worden op slip, powerfactor, stroom, vermogen en efficiëntie met een motorkarakteristiek test. Met de tweede test word de koppel toerental grafiek van de motor opgesteld. De derde test meet de efficiency van de motor of de efficiency van de regelaar met motor wanneer de motor word aangedreven met een regelaar. De laatste test is een duurtest waarbij de motor voor een bepaalde tijd accelereert en decelereert.

1. INTRODUCTION

Vanderlande Industries develops value-added logistics process automation systems for airports, post and parcel and warehousing. In airport systems they do both passenger handling (hand held luggage) and baggage handling (cargo luggage). The Warehousing systems are subdivided in fashion, food and general merchandise. The post and parcel systems does not have subdivisions because they must all handle a wide range of products.

All these systems make use of electrical polyphase motors. Therefore Vanderlande wants to have the possibility to check the motor data given to them by manufactures to check whether the motors meet the requirements for the applications that Vanderlande wants to use them for.

Jaap Kanters [1] started the mechanical development of the testbench in 2015. This mechanical design is visualised in Figure 1. The blue shapes are the devices, the gray shapes are shafts to connect these devices and the black shapes represents transmission belts. This thesis will continue from there on with the first goal being to put the testbench together. The second goal is to select the fixed motors that will generate a load for the motors that have to be tested. The fixed motors need to have more power than the motors that Vanderlande uses in their applications or the best motor that fits inside the testbench. The third goal is to research how to measure motor characteristics. Motor characteristics are the power factor, efficiency, slip, current and output power of the motor depending on the input power. The main characteristics (current, torque and speed) can also be visualized in a torque-speed and current-speed curve. A fourth goal is to program the PLC (it is demanded that this PLC is a SEW controller UHX85A-R) so it can power the motors to get all test data. SEW controllers use Codesys to control the dependent devices. These tests are the motor characteristics, torque-speed characteristics, efficiency test and endurance test. The last two are more specific tests for Vanderlande products. The fifth goal is to program an Excel macro for each test to use the test data and automatically generate a test report. These test reports have two sections, one with the test data and motor data that is necessary to duplicate the test. The second section contains the visualization (graph) of the test results depending on the test. The final goal is to make an instruction manual on how to use the testbench.

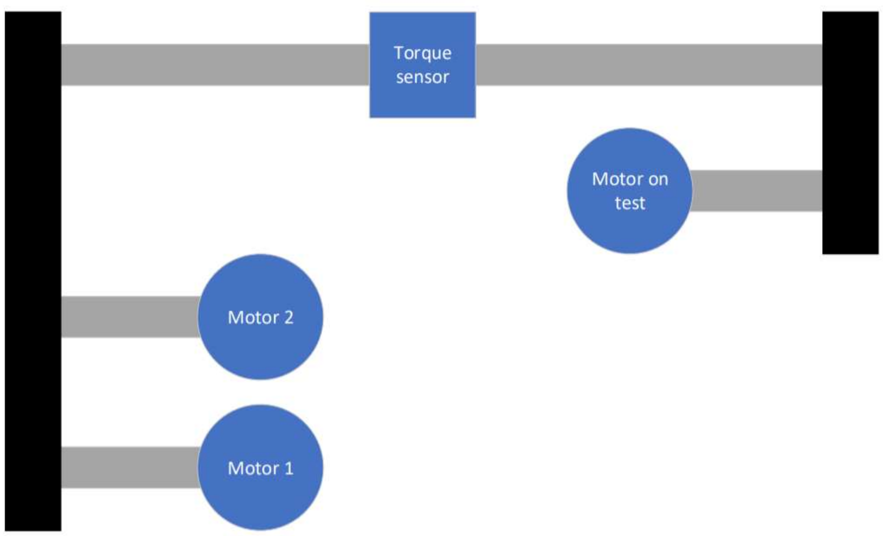


Figure 1: Mechanical representation of the motor testbench

2. LITERATURE STUDY

This Literature study is written to achieve prior knowledge which is necessary for this assignment.

2.1. MOVI-C controller

The MOVI-C controller is a motor controller from SEW and is avaidable in four variants [2]. These can be seen in Figure 2. From left to right there is the MOVI-C standard, MOVIE-C advanced, MOVI-C progressive (MOVI-C power eco [3]) and MOVI-C power.

All MOVI-C controllers can function as a slave on the Profibus and Profinet network while the standard and the advanced can also function as a slave on the sbus and sbusplus (ethercat) network. All controllers can be a master on the SBus and SBusPlus (ethercat) network, the advanced can also be a SNI master. The MOVI-C power can even function as a profibus or profinet master when it has an external gateway. A HMI can only be used by the MOVI-C power in combination with the embedded windows card. This motor testbench uses the MOVI-C power, later mentioned as PLC, because it is the most powerfull motor controller and there is little to no difference with a PLC.

Figure 2: MOVI-C controllers

2.2. Test protocols

This test bench must be able to do several tests. This part of the literature study will provide answer on what kind of tests are performed on other electric motors.

2.2.1. Efficiency test

The standard IEEE 112 is a technical worldwide standard on how to test polyphase induction motors [4]. They separate the motors based on power. Motors with less than 1kW of power can be tested with the “input-output” method. Motors with a power from 1kW up to 300kW must be tested with the “Input-output with segregation of losses and indirect measurement of stray-load loss” method to meet standard [4]. If this is not possible because of the design of the motor, then there are methods for motors with more than 300kW that can be used. These methods are:

- electric power measurement under load with segregation of losses and direct measurement of stray-load loss.
- electric power measurement under load with segregation of losses and assumed value of stray-load loss.

- equivalent circuit with direct measurement of stray-load loss.
- equivalent circuit with assumed value of stray-load loss.

The other tests for motors with more than 300kW do not apply for this test bench.

All these test methods are correcting the losses compared to the value that they would have if the motor would stay at a temperature of 20°C. The difference between them is which losses are corrected, and which loss correction can be neglected. The IEEE standard allows more loss corrections to be neglected at lower powers.

The difference between the tests “electric power measurement under load with segregation of losses and direct measurement/assumed value of stray-load loss” is that one needs a stray-load test where the losses are measured, and the other one calculates these losses by multiplying a constant value (depending on the power of the motor) with the nominal power of the motor. Both tests do not measure the output power while all the input-output tests require this to be measured.

The methods that use equivalent circuits are not interesting for this assignment because the efficiency will be tested, and not determined by calculations. The three "input-output" methods are better suited for this assignment. However, this assignment started from the need of knowing the real efficiency of the motor. That includes the losses of the motor when the temperature is higher than 20°C.

2.2.2. Torque-speed characteristics

There is a school in Croatia where students use an older version of the IEEE 112 standard to measure the torque-speed characteristics from a motor [5]. The method that they have been using before is the “direct measurement” from [4] and the method that they are implementing in [5] is the acceleration method, also from [4]. Figure 3 shows such a characteristic.

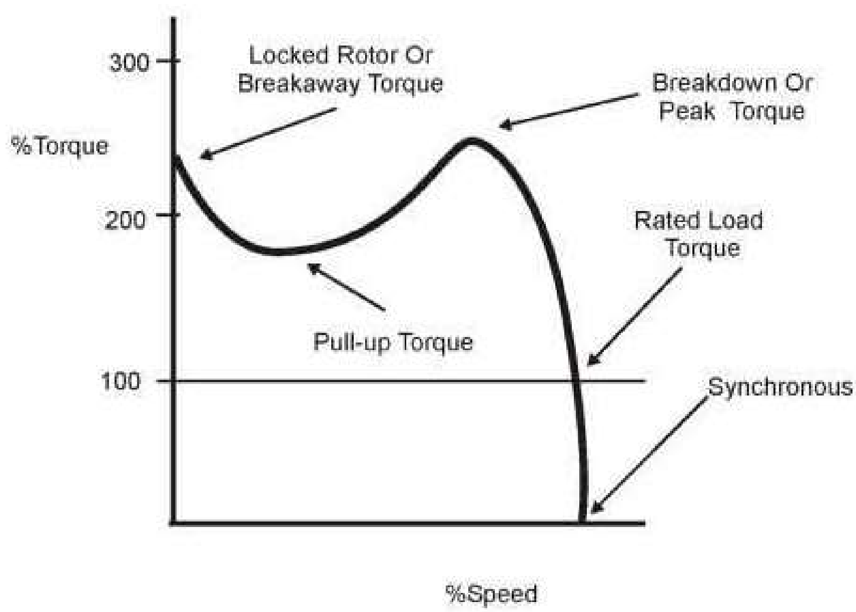


Figure 3: Example Torque-speed curve [6, p. 99]

In the direct measurement they use a dynamometer with a torque sensor and control the speed of the motor for 12 different measuring points by changing the voltage on the dynamometer. All this is done with reduced voltage on the motor. For these 12 points torque, phase, voltage, current, and input power are measured.

When using the acceleration method there is chosen to add a load to the motor. This will increase the mass inertia and decrease the acceleration. The benefit of a lower acceleration is that there will be more time till the motor reaches the rated speed, this gives more measuring points and less effect off the electromagnetic time constant. To even further reduce the effect of the time constant the motor is externally rotated in reverse before starting the test. The benefit of the acceleration method is that it can be done at rated voltage and the motor generates less heat than other methods.

Two other methods to calculate the torque-current characteristics are explained in [4]. These are the measured output method and the input method. For the measured output method, the motor should be coupled to a dc generator. The torque of the motor is calculated from the input power of the generator at each speed and the input power of the generator is calculated from the output power and the previously determined losses of the generator. The torque from the input method is calculated from the input power, subtracting the losses. The benefit of this method is that the load does not have to be disconnected from the motor. The difference between the acceleration method and the input method (both methods can be performed under load) is that for the acceleration method the load must be measured.

The measured output method is not interesting since it requires a DC generator. The input method is possible, but it is not a preferred test in [4]. The direct measurement and acceleration method are more suited for this assignment since we have a torque sensor, speed sensor and an adjustable load.

2.2.3. Motor characteristics

The motor characteristics are included as documentation of a motor. Their layout depends on the manufacturer. Vanderlande has two preferred suppliers for motors. These are NORD DRIVESYSTEMS and SEW-EURODRIVE. Both suppliers measure slip, current, power factor, output power and efficiency in function of the input power at nominal rpm. The difference between NORD (Figure 4) and SEW (Figure 5) is that NORD has rescaled slip, current and power factor to percentages to visualize them on one axis while SEW has them in different scales and uses four axes for them. Both have the Load (or Power output) on an extra axis. In total NORD has two vertical axes and one horizontal axis, while SEW has five vertical axes and one horizontal axis.

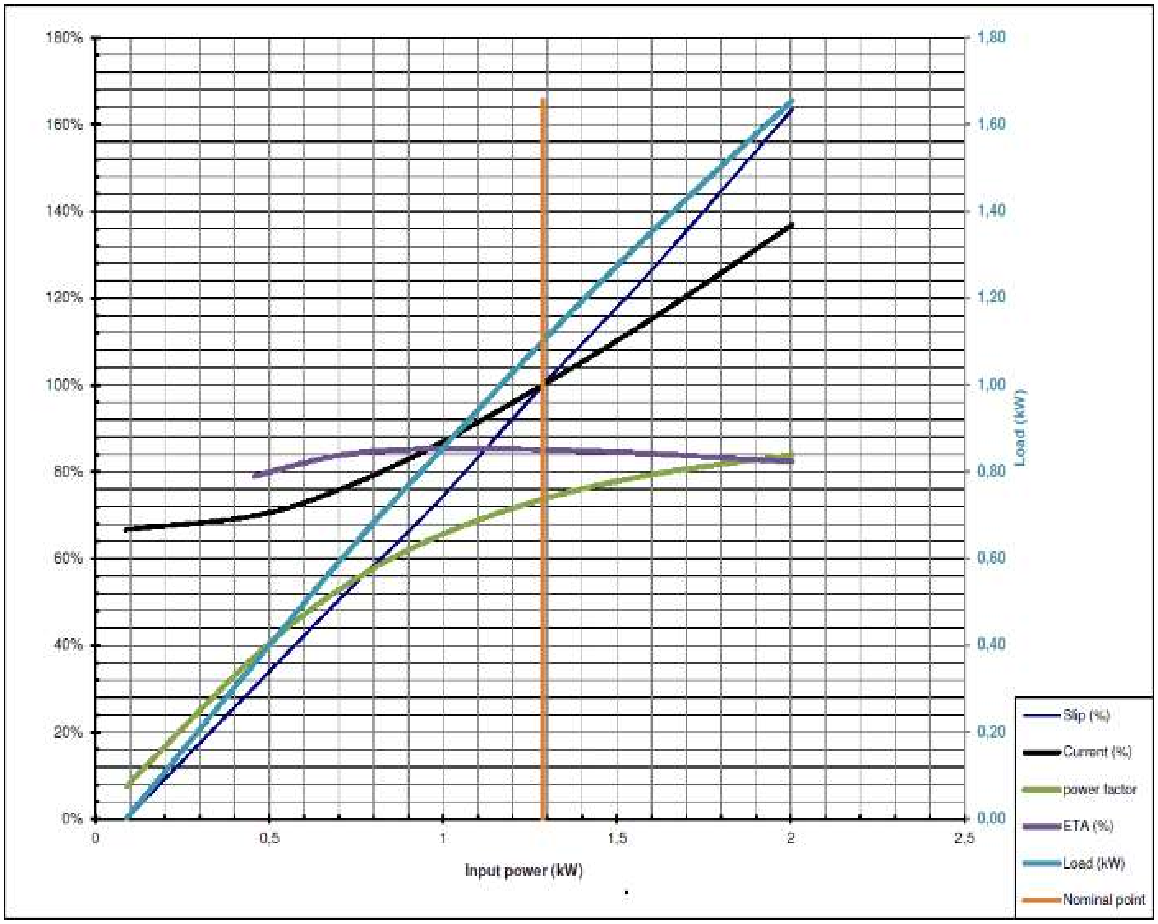


Figure 4: Motor characteristics NORD [7, p. 2]

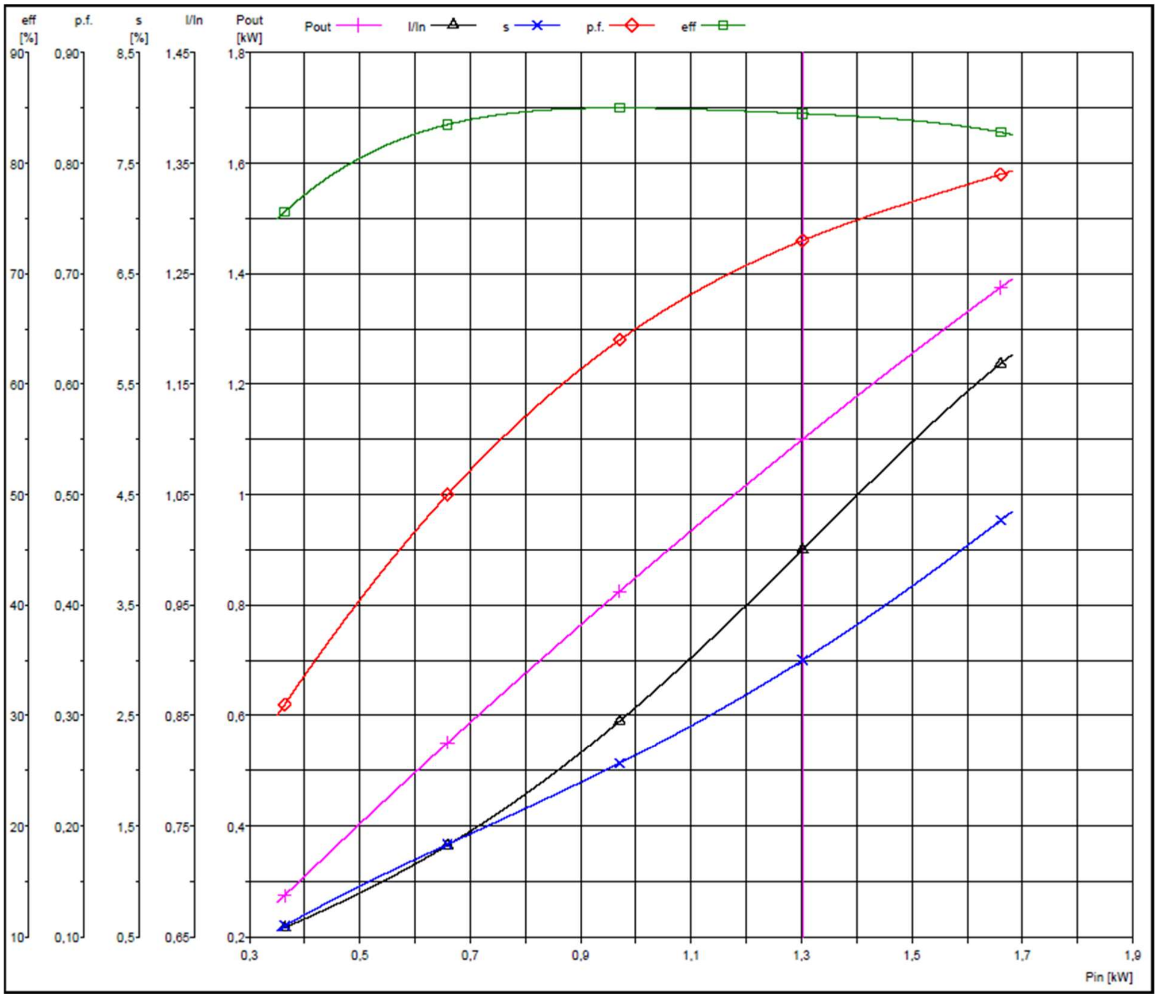


Figure 5: Motor characteristics SEW [8, p. 1]

Excel is the preferred program for the visualisation of the characteristics and can only plot graphs with a maximum of two vertical axis. Therefore, the NORD layout is better than the SEW layout for plotting motor characteristics graphs.

2.3. Exporting data

It is necessary to export the measured or calculated data from the PLC to the data processing program Microsoft Excel. After exporting the data to Excel a template will be used to visualize the data.

2.3.1. Writing to USB

The MOVI-C controller has seven USB ports. Five of them are used for the Windows embedded system (if installed) and the other two are for the PLC [3]. These two USB ports are used to restore the PLC to delivery conditions and cannot be used to write data to [3], [9].

You can however write data to the Cfast card [10]. The Cfast card is the memory card for the controller where the program is located. If the controller is powered of, the Cfast card can be extracted from the controller and connected to the PC with a Cfast to USB converter. The data can also be copied to the PC with Codesys and an ethernet cable.

2.3.2. Codesys Trace

The MOVI-C PLC is developed to be programmed by Codesys. Codesys is hardware-independent software to program controllers. Codesys Trace is a developing tool inside Codesys that tracks variables while the program is running [11]. The values are saved in a buffer, this buffer size can be chosen between 10 and 4 294 967 295. The tracking of these variables can be started manually when the PC is in online modus with the PLC or by the PLC when a chosen variable is set on a chosen value. Running the trace will increase the cycle time of the PLC. Figure 6 shows this developing tool.

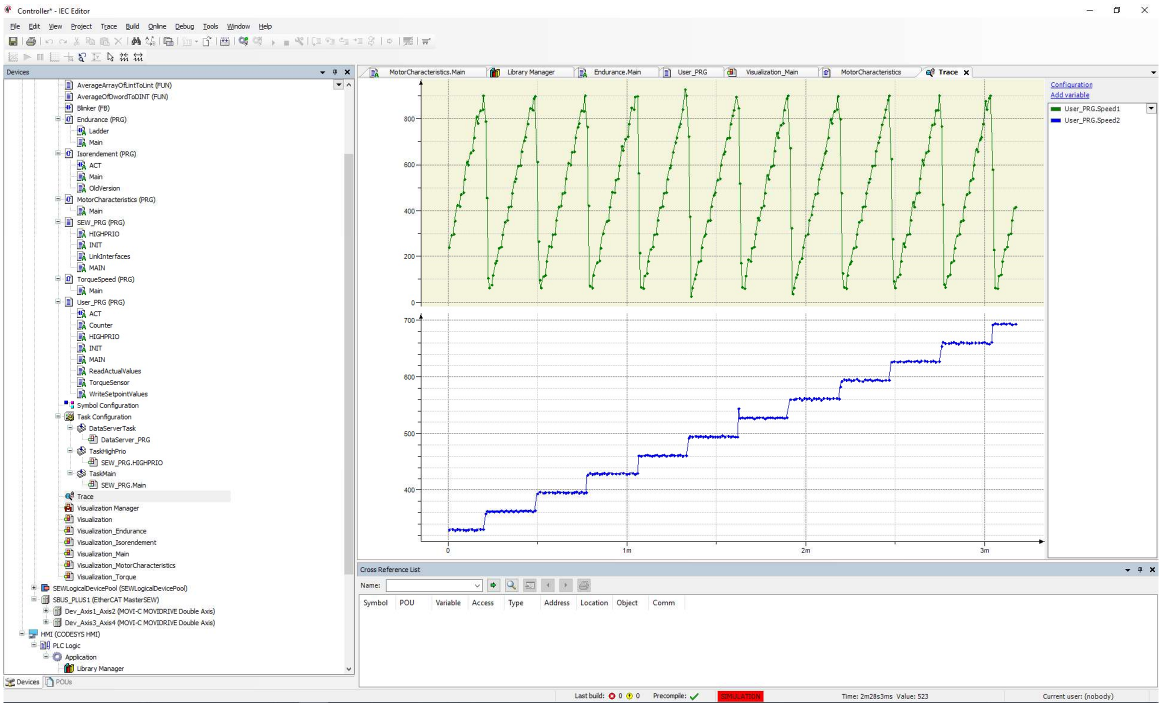


Figure 6: Codesys trace

The trace can be exported in four different file types. The first type is “trace file” and it can be loaded back in Codesys. This file contains the trace configuration and the measured values. The second type is "text file" and contains a path link to the location of the PLC program and all the values in plain text. The "trace dump" type is the third type, this is a CSV file that contains the trace configuration in text and all the values. The last type is “symbolic trace configuration” and does not have the measured values, only the configuration. This file can be imported back in Codesys to trace the same variables. The trace can be buffered on the PLC while the PC is not in online mode, but it is necessary to extract them from the PLC in online mode before the buffer is full. Otherwise, there will be a loss of data.

The disadvantage of this method is that it saves the value of variable every x amount of cycles. Another disadvantage is that, when using “trace dump” to export the values, you are limited to the end of Excel (the end row is 1 048 576) [12] because it plots all the measurements below each other. Here is a theoretical example of a calculation for the number of measurements it will take to reach the end while tracking seven variables (power, voltage, current, speed, torque, set speed, set torque). After using 15 rows for trace settings and 14 rows for each variable only 149 780 rows per variable are left for measurements. With the smallest sample rate (every cycle) and a cycle time of 10 ms (estimated) that is 24 min and 57 sec till Excel is full. With the highest sample rate (every 1000’th cycle) the maximum time of measuring is 17 days 8 hr and 9 min with a cycle time of 10 ms.

2.3.3. OPC server

As a third possibility an OPC server can be used. An OPC server translates PLC parameters to genuine numbers on a windows device. If the PC is equipped with this tool it can read and write a variable in the global variable list. The Codesys OPC server costs 159.5 euro [13]. This can be used in programs such as Excel as the following examples prove.

An Allen-Bradley is used by [14] to export the value list to Excel with the use of a macro. [15] uses RSlinx (a data server for Allen-Bradley plc's) to write and read parameters from and to his PLC.

A second setup is with an industrial plc running on MATLAB. MATLAB has the benefit that an Excel file can be written from the code itself. Making it not necessary to import the test data to Excel and reducing the number of manual interactions. A second benefit is that the results can be visualized in MATLAB while the test is still running. This is especially interesting for a long test like the endurance test.

The second programming language for controllers that can be used to generate Excel files and control an inverter is Visual Studio. [16] has programmed his PC with Visual Studio to control an I/O module from Beckhoff. This PC was a regular PC that was programmed to an industrial PC with the TwinCAT software from Beckhoff. As far as hardware goes these two programming languages can run on the same device.

2.3.4. Codesys scripting

Codesys has a scripting tool that works with IronPython [11]. With the code below, data can be read from Codesys inside IronPython.

```
proj = projects.primary
app = proj.active_application
onlineapp = online.create_online_application(app)
onlineapp.login(OnlineChangeOption.Try, True)
if not onlineapp.application_state == ApplicationState.run:
        onlineapp.start()
Value1 = onlineapp.read_value("PLC_PRG.iVar1") [17]
```

This code makes a connection between Codesys and IronPython. After the connection is made it will check if the PC is online with the PLC, if it is not online, then it will go online. Now that the connection is made and the PC is online, the script will be able to read out a value "PLC_PRG.iVar1" from the program. This variable is called "iVar1" and is a variable inside the POU named "PLC_PRG".

When the value is read from the program it has to be written in a file. The following code can be used to write the value named "value" to a file on the PC.

```
f=open('/tmp/workfile','w')
f.write(value)
f.close() [18]
```

This code opens a file named "workfile" from a folder named "tmp" inside the IronPython directory, if this file does not exist, it will be created. This directory can be changed, for example to "my desktop".

```
F=open('C:\\Users\\nlhvdve\\Desktop\\Testfile.txt','w')
```

This code will open or create a file named "Testfile" with the .txt extension on the desktop of my PC. Other file extensions can be used. The second part of the "f=open" function (where 'w' is written in the example) describes the mode that the file will be opened in. Depending if you use windows or not there are 4 or 8 modes to open a file. The first mode is 'r', this is a read-only mode. If the file is opened in this mode, you can only read data from it. The second mode is ‘w', this mode will delete the existing file (if it exists) and will create a new file. The mode ‘a’ is for adding data, using this mode you will create a new file if the file referred to does not exist. If it does exist, you will be writing additional data at the end of the document. The mode ‘r+’ is for reading and writing. The four additional modes for windows are to open the file in binary mode. The additional syntax for opening the files in these modes are ‘b’, ‘rb’, ‘wb’ and ‘r+b’.

When this method is chosen it is necessary to calculate how many measuring points are necessary to plot the graphs. When the amount of points is known, the scale of the points must be chosen automatically depending on the motor. If all this is done there will always be a fixed number of variables for each test to be extracted.

2.3.5. Movisuite scope

The Movisuite software has a scope tool that tracks variables over time [19]. 10 variables can be tracked on the scope. Long-time tracking is available in “Continuous trace mode”. With this method every 2500 values (the drive itself can hold 2500 measurements on its memory) are transferred to the PC. With “Single trace mode” all traced values are stored on the drive and the trace will stop automatically as soon as the memory of the drive is full. The possible sample rates for this scope are: 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000 or 10 000 ms. Figure 7 shows the layout of the program Movisuite scope. Movisuite scope exports all the values from the same time in the same row. Therefore the number of rows is not dependant on the number of variables. However, it is still possible to reach the end of Excel. With a sample every 0.01 ms the end is reached after 10 sec and 485 ms. With a sample rate of 10 ms it takes 2 hr 54 min and 45 sec and at a sample rate of 10 000 ms it will take 121 days 8 hr and 40 min till the last row of Excel is reached. The end the internal memory in “Single trace mode” is reached after 6hr 56 min and a sample rate of 10 000 ms and after 1 sec 200 ms with a sample rate of 0.01 ms.

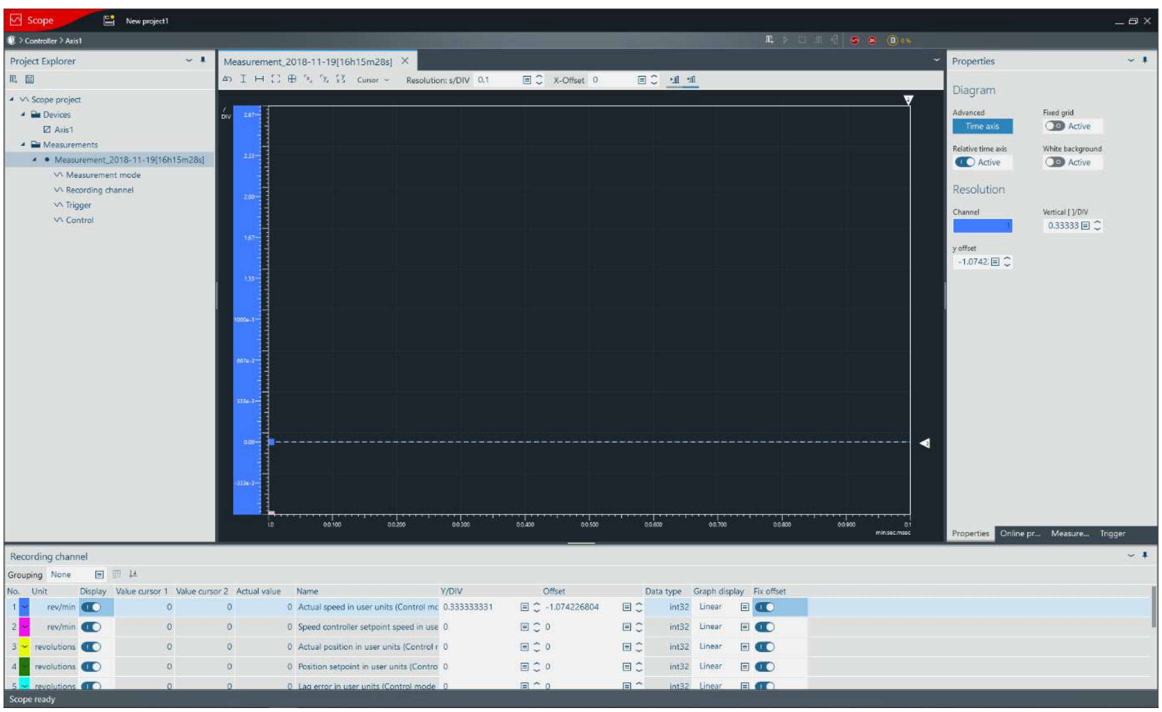


Figure 7: Movisuite scope

A benefit of this tracker it that it has the most samples per second, this is because it is measured inside the drive while Codesys trace measures inside the PLC. The variables it can track however are limited to the variables of that axis. To use this tool, it would be necessary to measure the torque and the speed on this drive. When all the data are extracted, calculations should be done by Excel since raw measured data are exported. The Movisuite scope function is a new and specific tool for automation developing, therefore there is no manual available today. Because of this, it might only be interesting to use this tool if the sample rate is necessary for the chosen test protocols.

2.4. Inserting motor nameplate in Movisuite

If it is necessary to control the motor by the drive, then the motor must be added inside the Movisuite software. When a motor is chosen with an electronic nameplate then all the values will be automatically displayed in Movisuite and one will only have to confirm with the button “Use electronic nameplate” [19]. When no electronic nameplate is implemented in the motor, one will need to type the name in the drive train tab of the configuration window and this will adjust all the settings that can be derived from the name. After controlling and accepting these settings, the motor is ready to be used.

To insert these variables from the visualisation screen inside the PLC and from the PLC inside Movisuite are adjustments necessary by SEW [9]. These adjustments are temporary and will not work anymore when a new version of Movisuite is released. Therefore [9] does not recommend this for a project of this size but rather to make a detailed manual on how to change the parameters of the drive train inside Movisuite.

3. HARDWARE

This chapter will discuss the choice of hardware necessary for automating the testbench as discussed in [1]. This will also discuss the components discussed in [1] that where not ordered. The construction (after adjustments and installation of all the parts) is shown in Figure 8.

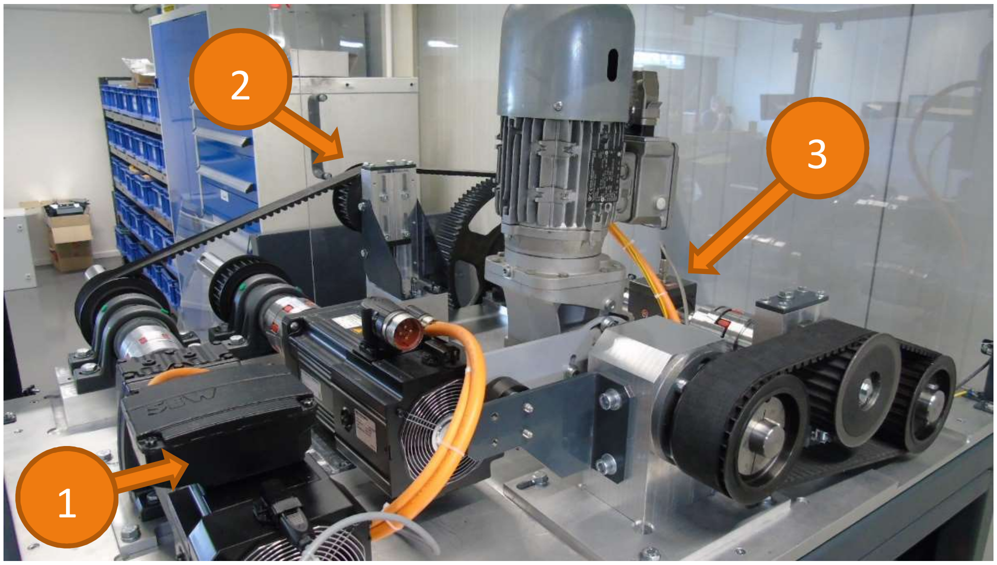


Figure 8: Testbench

The motors that function as a load (fixed motors) are marked with (1) and will be discussed in 3.1. These are connected to a torque sensor (3) with a transmission belt (2). This transmission belt has a gear-ratio of 90/29. This was done by [1] to allow for high torque motors to be tested. Due to the design the fixed motors will be limited to a shaft height of 90mm. one has an available length of 635mm and a width of 259mm while the other one has an available length of 375mm and a width of 247mm.

The four tests that will be explained in chapter 5 can be separated in DOL-tests and Drive-tests. A DOL-test is a test where the motor is on or off where with a Drive-test the motor can be controlled. The necessary devices for a DOL-test and a Drive-test are different. The structure of a DOL-test can be seen in Figure 9 and for a Drive-test can be seen in Figure 10. In these diagrams data wires are coloured blue, electrical energy is coloured yellow and mechanical energy is coloured orange.

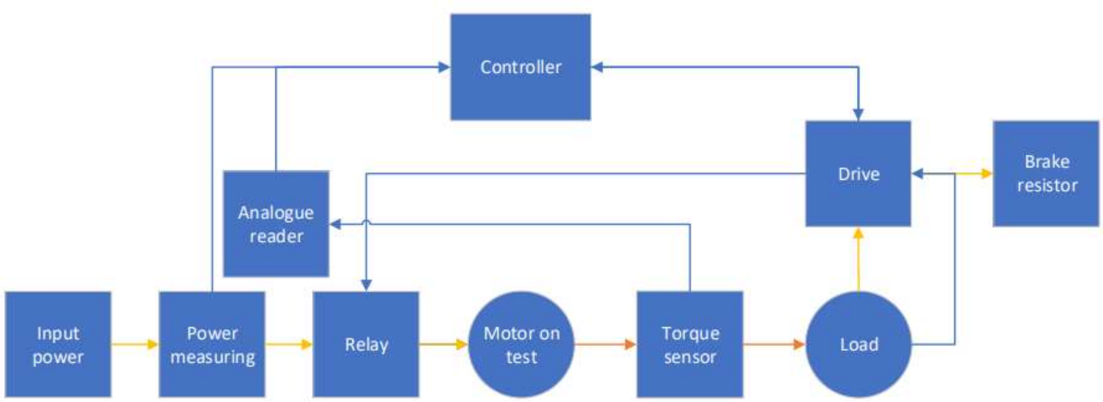


Figure 9: Device structure DOL-test

The motor-on-test in a DOL-test is powered from a power measuring device and controlled by a relay. This is done to know the input power and the input current for the motor characteristics test (5.1) and the torque-speed test (5.2). The load (discussed in 3.1) only generates power in these tests. This energy is discarded in a brake resistor but could be reused with another power module (which is not visualized in Figure 9 but will be discussed in 3.1.3).

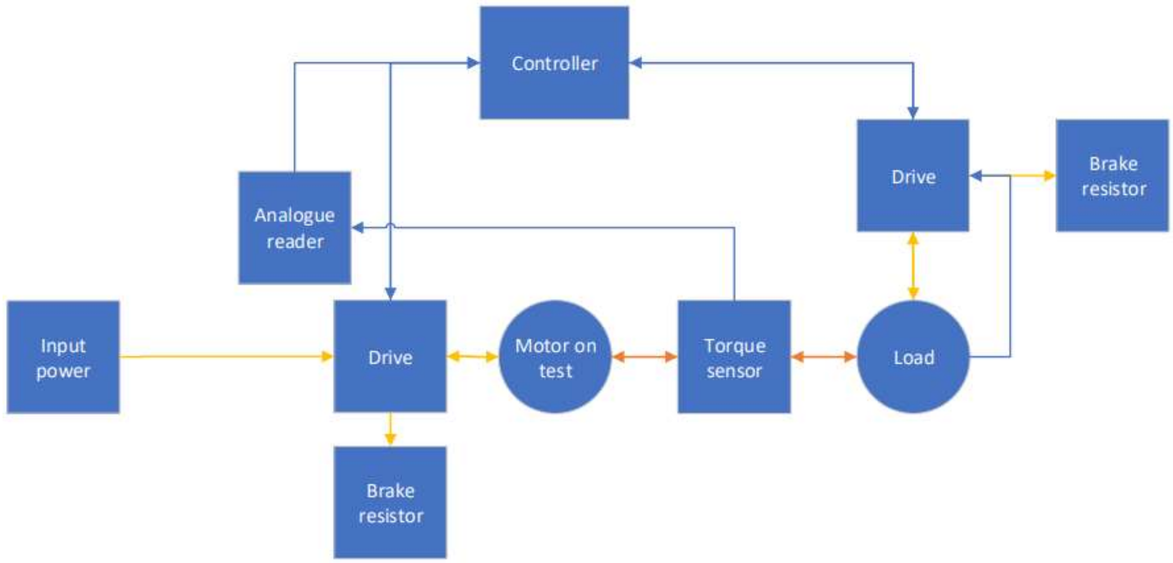


Figure 10: Device structure drive-test

The motor-on-test is powered from a drive with the efficiency test (5.3) and the endurance test (5.4). The drive of the motor-on-test has to be equipped with a brake resistor because the load will power the motor-on-test during the deceleration of the endurance test. This setup does not need a power measuring device because these parameters can be read from the drive. To test drives in the future for THD's, a power analyzer could be mounted temporary for these tests.

3.1. Motor choice

Vanderlande has 139 different motors (Appendix A) and would like this testbench to be able to test all these motors. This list is visualised in Figure 11. To be able to test the motors, the fixed motor has to have a higher output torque and the same speed as all the motors on the list. The highest torque and highest speed achievement by a motor on this list are 500Nm and 615rpm. The output power of a fixed motor with 500Nm torque and 615rpm would be

$$\mathrm{P} = \frac{T*n}{9.55} = \frac{500*615}{9.55} = 32\ 199Watt \ = \ 32.2kW \tag{1}$$

Because a motor with this amount of power is too big for the testbench [1] has chosen to provide two places for fixed motors in the testbench.

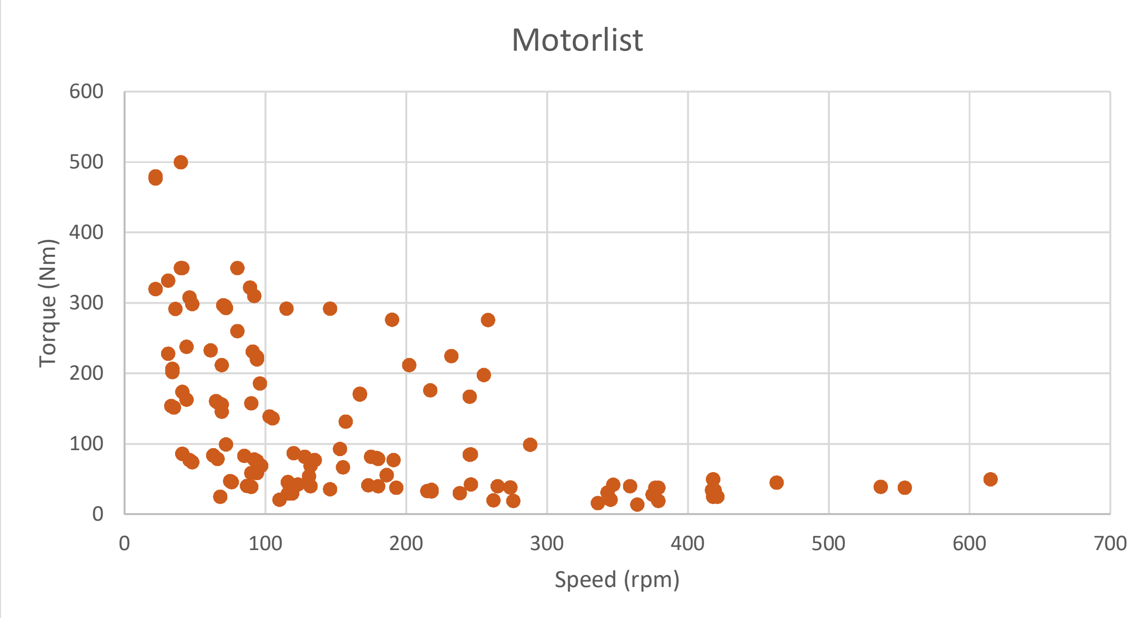


Figure 11: Graphical representation of motors

Because the motors in Figure 11 are easily separated in a group higher than 300rpm and lower than 300rpm there is chosen to have one motor for each of these groups. Because there is a belt transmission ratio of 90/29 on the testbench between the fixed motors and the test motor, our fixed motor for slow speeds will need an output speed of at least 900rpm while our fixed motor for high speeds needs an output speed of at least 1800rpm. The output torque for the fixed motors is lowered by a factor three because of this transmission ratio to 161Nm (500 / (90/29)) for slow speeds and 16Nm (50/ (90/29)) for high speeds.

Because of the high power and low space that is available on the testbench, servo gearmotors are chosen to be used as fixed motors.

3.1.1. Fixed motor for slow speeds

To achieve the high torque (161Nm) and low speed (900rpm) this motor will need a gearbox. The available gearboxes SEW has to offer are:

- helical gear unit,
- helical-worm gear unit,
- spiroplan gear unit,
- parallel shaft helical gear unit,
- helical-bevel gear unit,
- servo helical-bevel gear unit,
- servo planetary gear unit (solid shaft),
- servo planetary gear unit (flange block shaft),
- “low invest” servo planetary gear unit (solid shaft).

The servo gear units are not designed for use at high speeds or for extended periods, this is because the cooling would be insufficient. The helical gear units are developed for use in a rough environment and are therefore larger than normal gearboxes. The spiroplan gear units are limited to 3kW and 180Nm and have a different mounting position than the one provided on the testbench (the testbench is developed for a parallel shaft gearbox). A worm gear unit is a gear unit that has a high output torque and a low output speed. The helical gear unit is the preferred gear unit because of its simplicity, low friction and low heat production.

Of the helical gear units, the RX variant is the preferred one because it has a single stage gear transmission. Gearboxes with single stage gear transmissions have lower heat production then multi-stage gear transmissions. The width of an RX gearbox however exceeds the maximal width available in the testbench (The maximal width is 160mm). Therefore a multi-stage helical gearbox is the best gearbox for this application.

The exact motor gearbox combination is also limited to the height of the motor axle. The mechanical components are calculated for a shaft height of maximum 90mm. The motor gearbox combination with the highest torque output combination at 900rpm is an R37CMP71L. A larger motor (CMP80 and above) will result in exceeding the shaft height because another gear unit will be needed (R47).

This motor will be limited in output torque because of heat development in the gearbox. The torque will be limited to 74Nm for long duration tests and to 126Nm for short tests. Figure 12 shows a graphical interpretation of the fixed motors compared to all the motors with the high and low torque limits of both fixed.

3.1.2. Fixed motor for high speeds

For this motor, there is no gearbox necessary. The motor for the high-speed tests will be a CMP80L. this is the strongest motor with a shaft height lower than 90mm.

This motor has, like the R37CMP71L, a different maximum torque for long and short tests. These torques are 23Nm for long tests and 50Nm for short test. This is because the limited heat dispersion that the motor has due to its build. Due to the fixed motors overlapping at low speed and low torque one of them will be chosen as the preferred motor. This motor will be the fixed motor for high speeds because it does not use a gearbox. These maximum torque values are also represented in Figure 12.

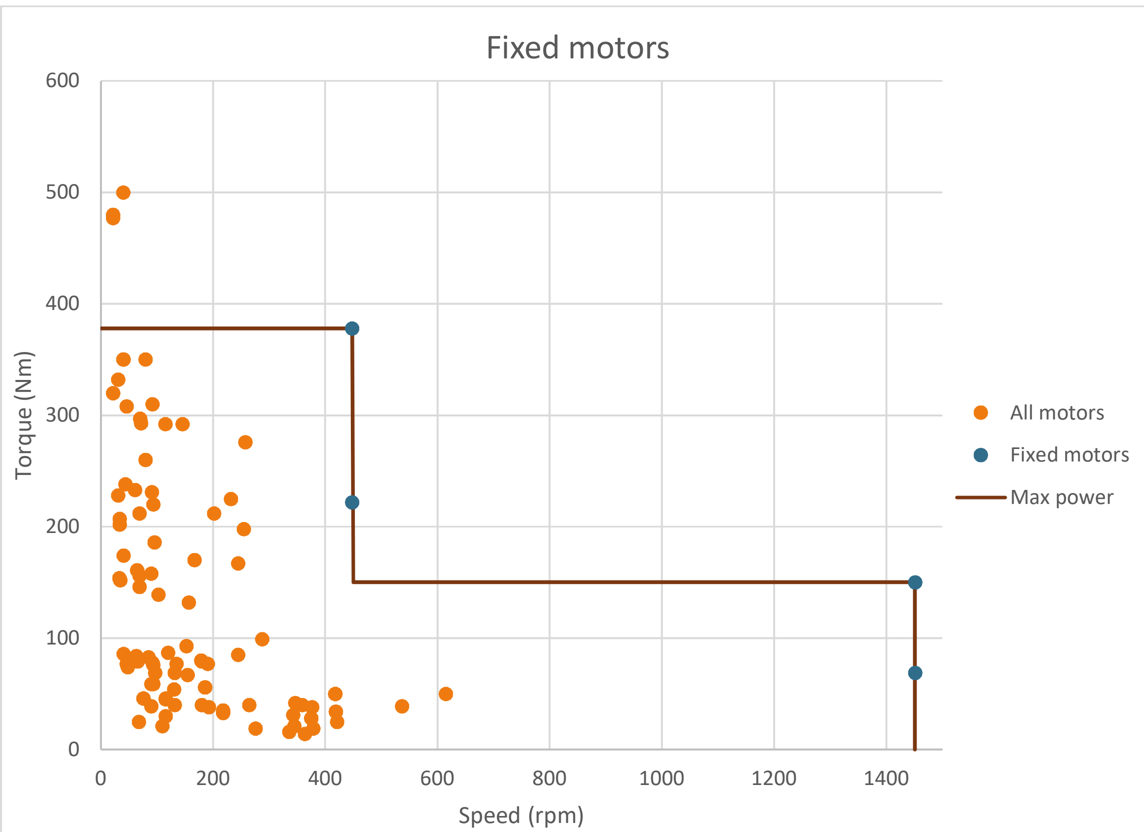


Figure 12: Graphical representation of fixed motors compared to all motors

3.1.3. Related motor parts

Other parts that depend on the choice of motor are the power supply module, the axis module the braking resistor and the encoder cable. The power supply generates DC power for the axis modules and they make AC power for their motor using PWM. And the encoder cable tells the position of the motor. Depending on the type of encoder and the connection another cable is necessary.

There are 2 different power supplies that power the drives. A regenerative power supply and a standard power supply. The regenerative unit can transfer energy back to the grid. This is interesting with tests in DOL and motors with different drives. The standard power supply dissipates the energy of the fixed motors in a braking resister when the regenerative unit would transfer it back to the grid.

The regenerative power supply with related parts would cost 3149.2 euro excluding taxes. The standard power supply and related parts would cost 2406.04 euro excluding taxes. The cheaper power supply uses a more expensive braking resistor to discard all the generated energy for all the motors for all the tests. Therefore the difference between the two is 743.16 euro instead of 871.63 when only the power supplies are compared.

Because of this high cost (minimal 2406 euro that probably will be upgraded to the 3149 euro regenerative power supply) the first generation of this testbench will have the standard power supply of 10kW to lower the costs. This power supply can only be used with a braking resistor with a minimum resistance of 26Ω, what equals to 2400Watt of nominal braking power, and limits the motors for this testbench even more. But ideal for a provisional testbench because these parts are already purchased for R&D projects.

The axis module is available in different sizes that can be seen in Table 1. The PWM frequency of all single axle inverters can be set at 4kHz, 8kHz and 16kHz [20]. A higher PWM frequency lowers the nominal output current as can be seen in table 1. A high PWM frequency is preferred for a higher accuracy of the output current and for lower noise levels of the inverter.

Table 1: Device data inverter module [20, p. 14]

Type designation	Nominal output cur-rent PWM = 4 kHz	Nominal output cur-rent PWM = 8 kHz	Nominal output cur-rent PWM = 16 kHz	Maximum output current
	A	A	A	A
MDA90A-0020-503-X-S00	2	1.5	1	5
MDA90A-0040-503-X-S00	4	3	2	10
MDA90A-0080-503-X-S00	8	6	4	20
MDA90A-0120-503-X-S00	12	9	6	30
MDA90A-0160-503-X-S00	16	12	8	40
MDA90A-0240-503-X-S00	24	18	12	60
MDA90A-0320-503-X-S00	32	24	16	80
MDA90A-0480-503-X-S00	48	36	21.6	120
MDA90A-0640-503-X-S01	64	48	32	160
MDA90A-1000-503-X-S00	100	75	50	250
MDA90A-1400-503-X-S00	140	105	-	280
MDA90A-1800-503-X-S00	180	125	-	360

Since the maximal power of the motors is lowered because of the limited space, the axis modules are chosen to the maximum power of the motors. The MDA90A-0640-503-X-S01 is chosen because this module is able to provide the maximum current of the motor (103A and 115A). And will provide the static current (29A and 30A) easily at 8kHz.

The chosen motors are equipped with a sin-cos encoder (RH1M). One motor has its encoder connected on terminal blocks while the other has a M23 circular plug. these have to be connected to the drives which have 15 pin D-sub connectors. The encoder cables on the terminal blocks are connected as shown in Figure 13. While the encoder cable with the M23 circular plug is connected as shown in Figure 14. This information was delivered with the motors. How this should be connected to a drive is written in [21], and visualised in Table 2. These encoder cables also allow for a temperature sensor of type KTY, TF or PK (different two wire temperature sensors).

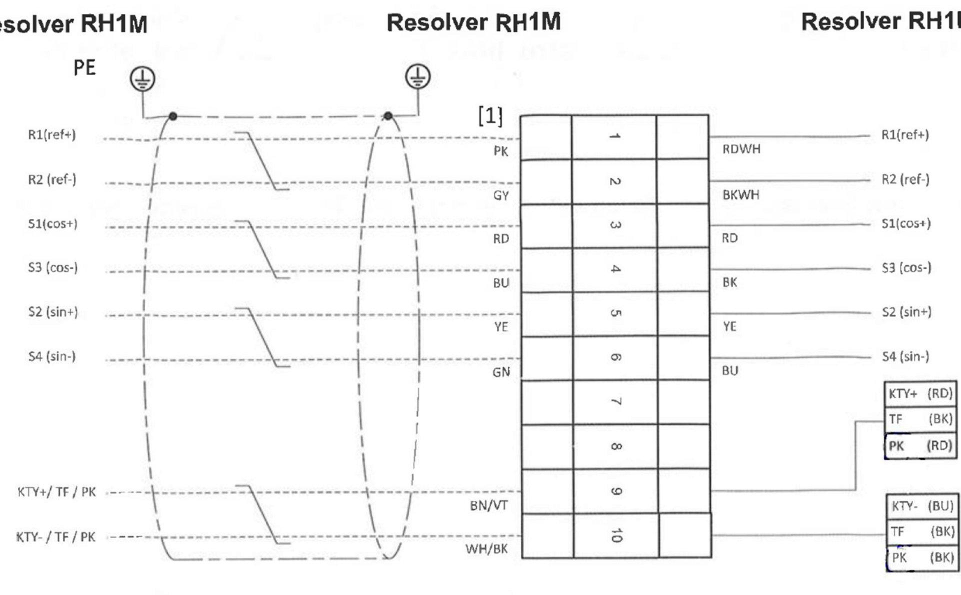
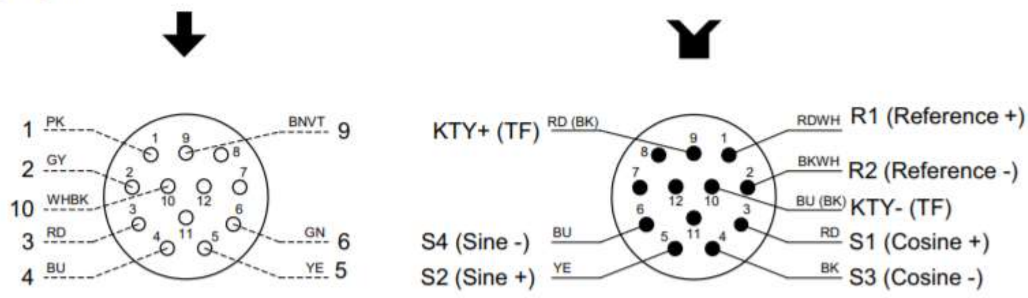


Figure 13: Encoder cable terminal blocks [22, p. 2]

RH1M resolver signal plug connector

Wiring diagram



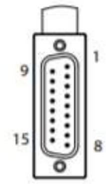
64627axx

Contact assignment of the plug connector lower part

Pin	Color code	Connection
1	RD / WH	R1 (reference +)
2	BK / WH	R2 (reference -)
3	RD	S1 (cosine +)
4	BK	S3 (cosine -)
5	YE	S2 (sine +)
6	BU	S4 (sine -)
7	-	-
8	-	-
9	RD	KTY +
10	BU	KTY -
11	-	-
12	-	-

Figure 14: Encoder cable M23 circulair plug [23, p. 193]

Table 2: Encoder pin assignement [21, p. 140]

Motor connection side					Connection MOVIAXIS® MX	
Plug connector view X	Pin no.	Description	Cable core color	Description	Pin no.	Plug connector view Y
ASTA 021FR	1	R1 (reference +)	(PK) Pink	R1 (reference +)	5	D-sub
0198 6732	2	R2 (reference -)	(GY) Gray	R2 (reference -)	13	15-pin
12-pin with socket contacts	3	S1 (cosine +)	(RD) Red	S1 (cosine +)	2	
	4	S3 (cosine -)	(BU) Blue	S3 (cosine -)	10	
	5	S2 (sine +)	(YE) Yellow	S2 (sine +)	1	
	6	S4 (sine -)	(GN) Green	S4 (sine -)	9	
	7	n. c.	-	n. c.	3	
	8	n. c.	-	n. c.	4	
	9	TF/KTY +	(BN) Brown/(VT) Violet[1)]	TF/KTY +	14	
	10	TF/ KTY -	(WH) White/(BK) Black[1)]	TF/ KTY -	6	
	11	n. c.	-	n. c.	7	
	12	n. c.	-	n. c.	8	
		-	-	n. c.	11	
		-	-	n. c.	12	
		-	-	n. c.	15	

1) Double assignment to increase cross section

All connectors are shown with view onto the pins.

3.2. Power meter

A power measurement tool is needed for DOL-tests. This measures the electric input power of the motor-on-test. This power meter should be able to measure at least the voltage till 400V, the current till 20Amp and the cosϕ. This paper looked into three different methods of measuring the electric power and how to read the values with the MOVI-C PLC. The power meter and the torque sensor voltage reader (3.3) are dependent of each other and therefore the chosen power meter will be decided in 3.3.

3.2.1. Janitza

Janitza is the only preferred supplier for power meters at Vanderlande. But it is not possible to directly connect a Janitza device to the MOVI-C controller because the controller does only support Ethercat slaves, and Janitza does not have a device that communicates over Ethercat [3], [24]. There is a device that can be used to couple the MOVI-C controller with other devices that are not Ethercat compatible. Such a device is called a gateway. There are multiple types of gateways but here we will be limiting it to the MOVI-PLC I/O System [25] of SEW. This device will also be discussed in 3.3.2 as a possible use for connecting the torque sensor. The I/O system will need an option card to read an RS485 supported device. This option card is the ORS11C. RS485 Modbus is one of the supported interfaces of Janitza power measurement devices. A positive thing about the Janitiza is that it can measure the harmonic distortion and that it can therefore be used to also measure the motor input power in a drive-test. The device structure , for a DOL-test, with a Janitza would look like Figure 15.

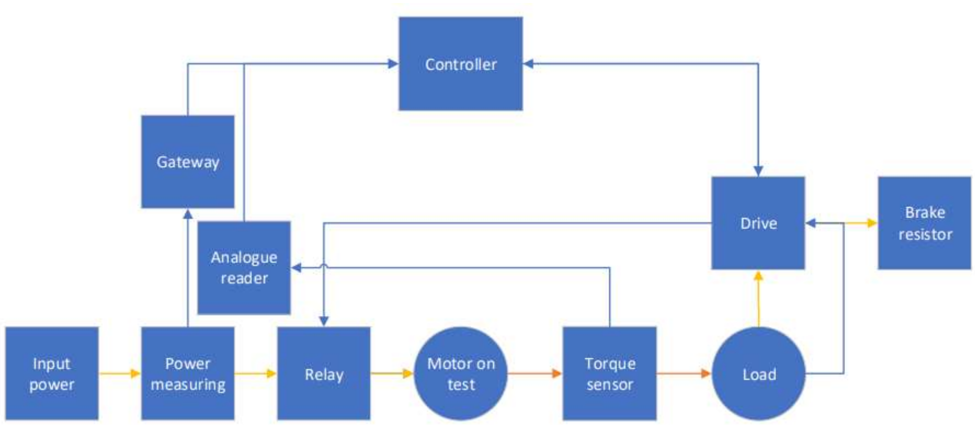


Figure 15: Device structure with Janitza

3.2.2. Measurement transducer

A second possible method to measure the electric power is by measuring the current and voltage and reading the separate values with an analogue input module. With multiple voltage measurements transducers, the cosϕ can be calculated as shown in Figure 17 and Figure 16 [26]. This method measures the voltage over the load, over a resistor and over the load and the resistor so that the cosϕ can be calculated with the equation (2) [26].

$$\cos\phi = \frac{U1^2 - U2^2 - U3^2}{2 * u2 * u3} \qquad (2)$$

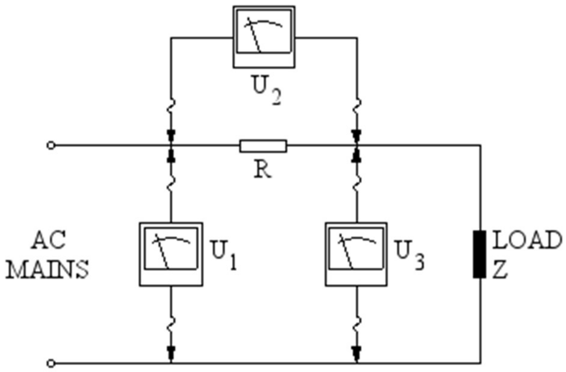


Figure 17: Wiring diagram three voltage method [26]

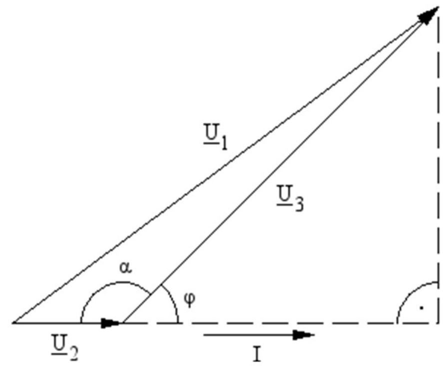


Figure 16: Voltage vector diagram three voltage method [26]

The reduced voltage and current can be digitalised by either an analogue input card on the inverter or with an I/O system. The inverter can only have one input card and every input card can only read two analogue signals limiting the amount of voltage/current transducers to four when using the analogue input cards on the inverter off the fixed motors. With four transducers only a two wattmeter method or a one wattmeter method with cosϕ measuring can be performed.

The analogue input card (CIO21A) for the drive [20] has a resolution of 11 bits, the MOVI-PLC I/O system [25] has a 16-bit resolution (OAI42C) and Beckhoff I/O modules are available with a resolution of 12, 16 and 24 bits. This limits the application to only voltage, current, power and cosϕ, which is sufficient if only the motor has to be tested. When the testbench will be upgraded for testing a motor inverter combination it will need to measure the THD. The device structure when using measurement transducers is visualized in Figure 18, where "power measuring" is the transducers and the analogue input cards are in the drive of the fixed motors.

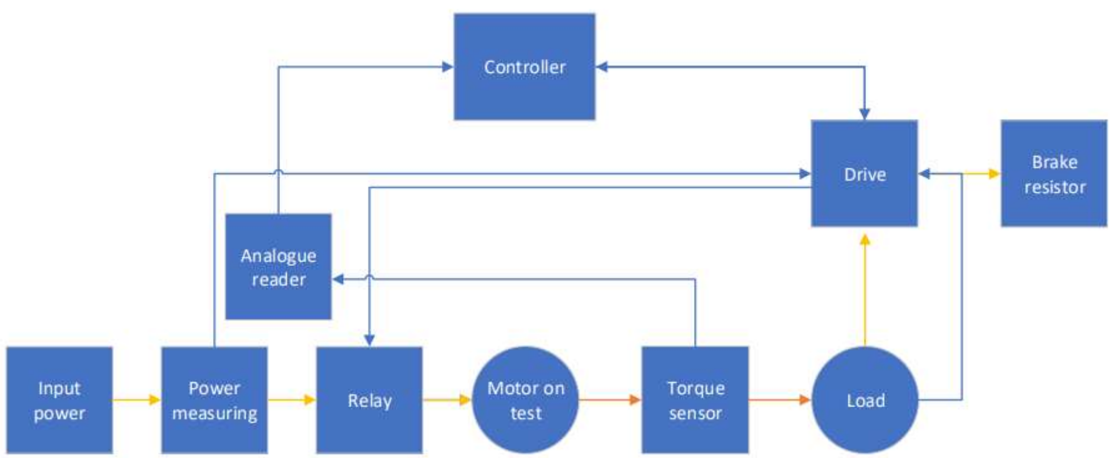


Figure 18: Device structure measurement transducers

3.2.3. Beckhoff

Beckhoff has invented Ethercat and therefore has lots of Ethercat modules. The EL3403 is a 3-phase power measurement terminal that can be read with an EK1100 coupler to be read over Ethercat [27]. The EL3403 can measure current, voltage, effective power, reactive power, apparent power, energy, cosϕ and frequency every 20ms. There are other power measurement modules that Beckhoff already has or is developing but these require the TwinCAT library that is not available for third party controllers. The Beckhoff module cannot correctly measure the output of the drive, this is because the EL3403 cannot handle the high frequency PWM signal. When to motor-on-test is powered with the drive, the input power and the current can be measured there. The device structure from Figure 9 does not change with the Beckhoff module.

3.3. Torque sensor

The torque sensor used in this application will be a FUTEK DRFL-VI-500 or a FUTEK TRS600 as written in [1]. The DRFL-VI-500 can measure a maximum torque of 500Nm at ±10V and the TRS600 can measure a maximum torque of 100Nm at ±5V. This is because these sensors have an interior differential amplifier. This voltage has to be digitalised by an analogue input reader. Multiple methods are available for this purpose, but this paper will only cover the most relevant three.

3.3.1. Analogue input card

An analogue input card can be mounted on the drive as in 3.2.2 to measure the torque from the sensor. The benefit of this method is that when the analogue input card for the drive is used, an I/O module would not be necessary. However, the drives of the fixed motors have not enough analogue input channels to measure the power and the torque. The device structure would look like Figure 19 if this method was used. The analogue reader (the input card) would physically be inside the drive of the load.

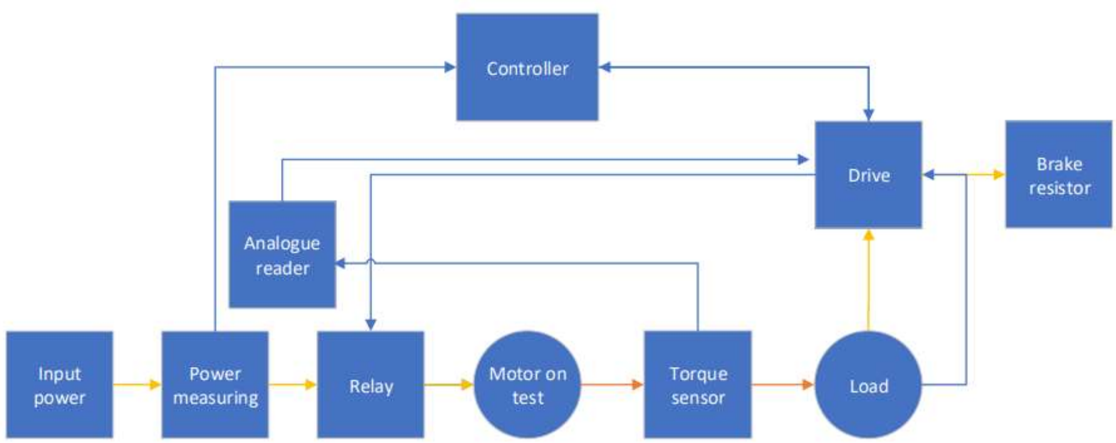


Figure 19: Device structure analogue input card

3.3.2. MOVI-PLC I/O system

The analogue input module (OAI41C) for the MOVI-PLC I/O system is suitable for reading an analogue voltage in the range of -10V to +10V and has a resolution of 16 bits in this area. This means that for the range of -5V to +5V the resolution will be 15 bits. This method uses an Ethercat to RS485 gateway making it more complex to program. Therefore it will be more difficult for other people inside Vanderlande to read and improve the testbench. The device structure would be the same as in Figure 9 and Figure 10 when the SEW I/O system would be used.

3.3.3. Beckhoff

The analogue input module EL3602 from Beckhoff can measure the signal voltage from the torque sensor (-10V till +10V) with a resolution of 24 bits and 23 bits for the torque sensor with an output range of -5V till +5V. this with an adjustable sample rate from every millisecond till every 400 milliseconds. The benefit of this method compared to the previous methods are the higher resolution and the excluding of a gateway. For these reasons was the EK1100 implemented in this application. Therefor the device structure is the one shown in Figure 9 and Figure 10. The sample rate was set to every millisecond but when looking at the raw data of an endurance test it was established that this was 40 milliseconds.

3.4. User interface

Vanderlande does not have a user interface for the SEW controllers. This is because the SEW controller is being used by Vanderlande as a standalone controller for the first time. All other uses of the controller is with a master PLC of a different brand. SEW recommends two touch screen devices in [3] to be used by the Windows operating system as user interface. A second method for a user interface is in the developing tool. A user interface can be implemented in the developing tool. This is to test the interface on your PC before you use an external touchscreen device. This will be used in this testbench to lower the costs but a HMI screen might be added with future updates. The controller can be disconnected from the PC after the test starts by logging of, the controller will continue with its test but will not have a HMI anymore. This is very important for the endurance test because during a test the controller does not need inputs from the HMI and the operator can take his PC elsewhere while a long during test is executed.

3.5. Safety

The testbench should come to a standstill in unsafe situations. Drives have STO contacts (Save Torque Off) which stop the drive from powering the motor. The motor will coast till standstill after the STO input is switched of (and STO is active). Power should also be cut off from the motor-on-test because, with an DOL-test, only using STO contacts will not create a save environment.

A safety relays (SIRIUS 3TK2821) is used to power off the STO inputs of the drive of the fixed motors, to power off two relays (that are mandatory to use in DOL-tests) and to power off the safety of the drive of the motor-on-test. An unsafe situation is created when the doors are open or when the emergency button is pressed. These are connected in series on the power in of the safety relays. A set button is connected to the safety relays to be pressed when the testbench is safe to use. This can all be seen in Appendix B.

Safety features in the software are discussed in the Instruction manual (7.4)

3.6. Electric diagram

The electric diagram (Appendix B) is drawn with Eplan and is as build (on 31/05/2019). There are multiple devices that are not necessary for every test, for instance the drive is not necessary when doing a DOL test (Motor characteristics or torque-speed test). This results in many devices that are only connected with terminal blocks. Also the designated space in the testbench for the electrical cabinet is too small to allow enough terminal blocks for an correct electrical cabinet as can be seen in Figure 20.

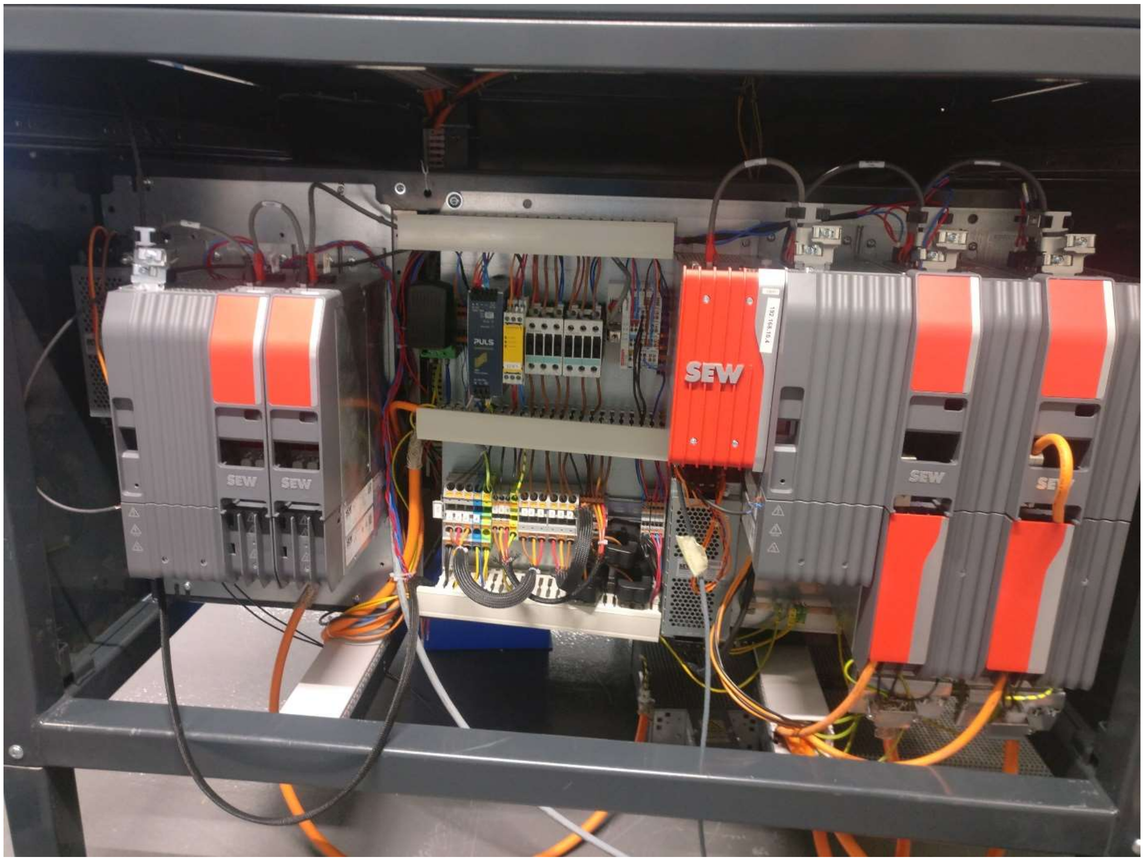


Figure 20: Cabinet layout

4. MODIFICATIONS

The mechanical design of the motor testbench contained some faults with delivery. In this paragraph these faults and shortcomings will be discussed. These can be seen in Figure 21 and are the:

- Frame,
- Tension unit (1),
- Doors (2),
- Fixed motor axle block (3),
- Transmission belt (4).

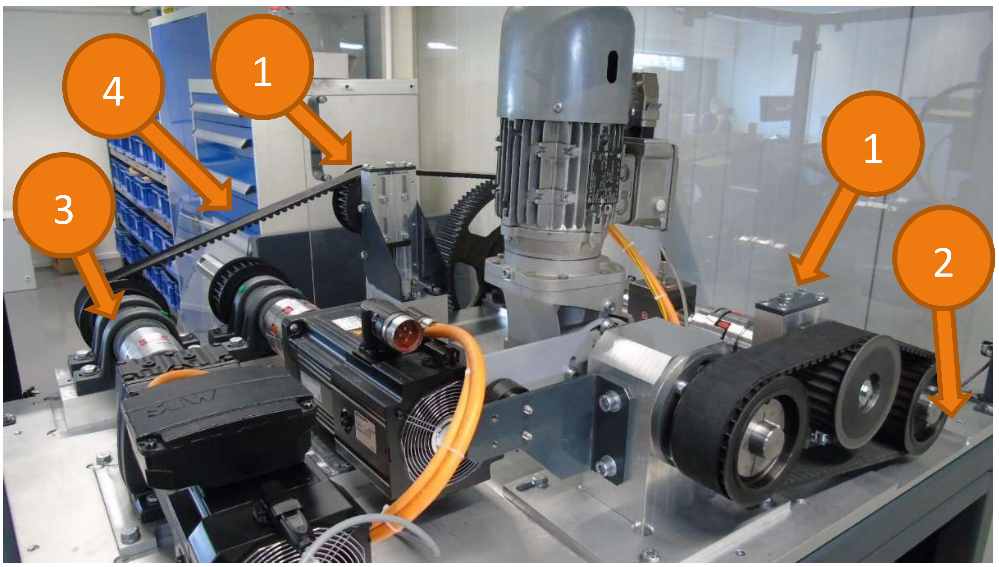


Figure 21: Modifications on testbench

4.1. Frame

The original dimension of the lower frame of the testbench where 111cm by 126cm by 130cm. The door of the room where the testbench is located is 100cm. To get the testbench in the room more easily the legs where made detachable, as can be seen in Figure 22. The legs where cut of and equipped with inner tube as a guidance with the upper frame. With this adjustment the lower frame only measures 86cm by 126 by 130cm and can easily fit through the door.

Figure 22: Testbench leg

4.2. Tension unit

The Tension unit sets a tension on the transmission belt by going up and extend the distance between pulleys. This upwards motion is achieved by turning a bolt as shown in Figure 23. The bolt (marked with 1) will attract the plate (number 2) and the pulley (attached to number 3) will follow due to gravity. The problem here is that if one lowers the pulley to much the plate will drop of the bolt and the Tension unit will have to be opened to reattach the bolt with the plate.

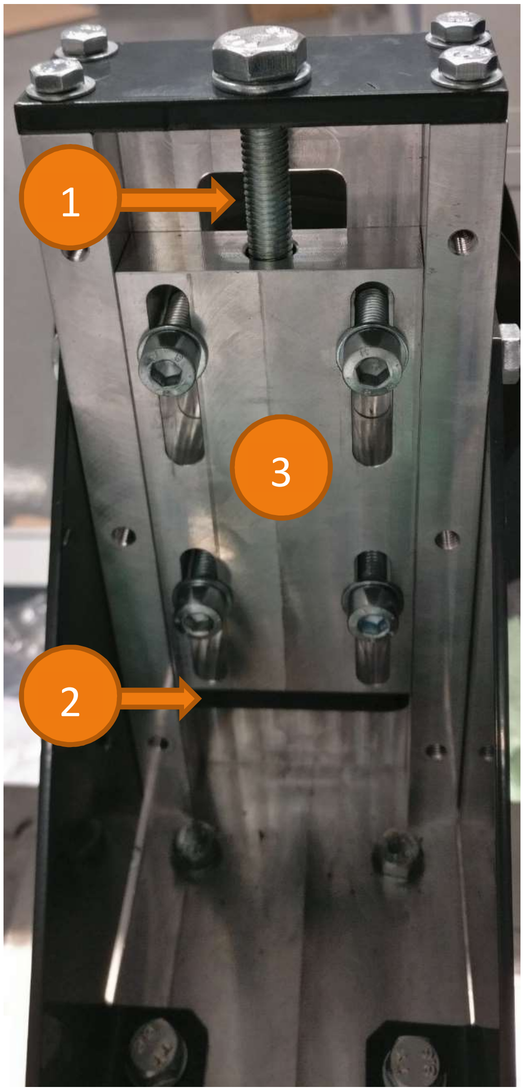


Figure 23: Tension unit

Possible solutions for this problem are a longer bolt or a nut down the plate. With a longer bolt it is still possible to lose the plate, but this will only be if one keeps turning the bolt when the pulley is already at the lowest possible position. With a nut at the end of the bolt it is not possible anymore to lose the plate. The nut lowers the useful distance of the bolt and makes it impossible to lower the pulley to the full possibilities with the delivered bolt. A longer bolt will be needed in combination with the nut.

4.3. Doors

The testbench has two sets of doors. Neither of the doors have a doorstop to prevent the doors touching the moving parts or are equipped with sensor to stop the test if someone opens the doors. On one of the door set, the holes for the hinges did not align with the corresponding holes on the frame. And the last modification that was necessary was on the other door set, where the height of the doors allowed one to touch a moving part of the testbench (as can be seen in Figure 24).

The doorstops are added in the form of available plate steel with door sensors mounted on them. The door sensors are safety proximity switches from SICK. The alignment for the door hinges is adjusted by drilling new holes in the doors. The other set of doors was lowered to make sure it is not possible to touch the transmission belt without opening the doors.

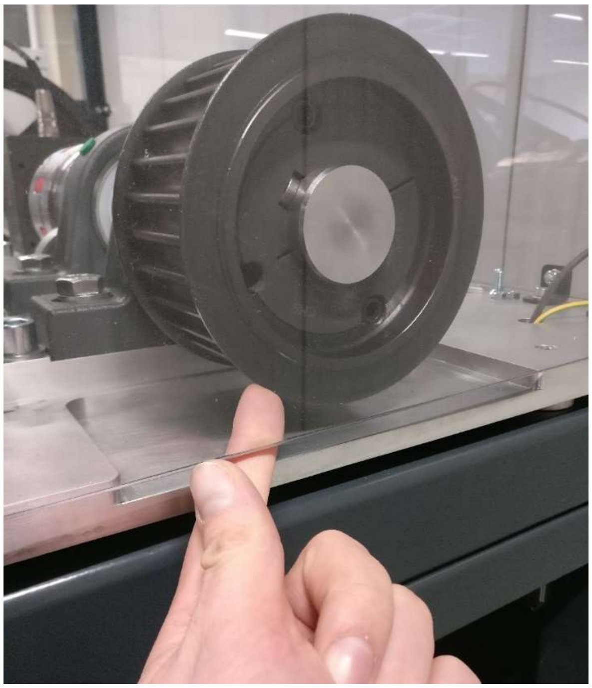


Figure 24: Unsafe door

4.4. Fixed motor axle block

The axle block of the fixed motor is movable to allow changing the transmission belt. This is necessary if the other fixed motor has to be used. Changing the belt is done by first lowering the tension from the tension unit. Secondly loosening the three screws of the fixed motor axle block. Afterwards the shaft coupling has to be loosened. The first hinderance is when one pushes the axle block to loosen al tension on the transmission belt. When this is done, the washer of the middle bolt and the left bolt will hit an edge, that can be seen in Figure 25. This problem can easily be solved by milling the surface flat.

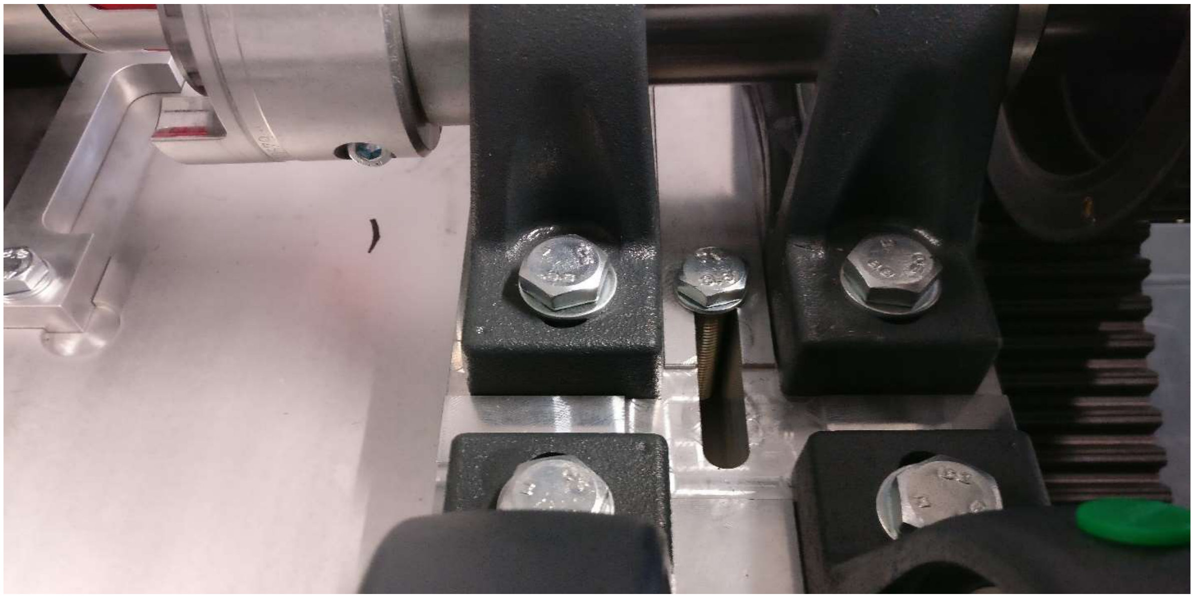


Figure 25: Fixed motor axle block height difference

4.5. Transmission belt

To transfer the power from the fixed motors to the torque sensor a transmission belt is used (Figure 26). The fixed motors will be attached to pulley3 and to pulley4. the transmission from 1 to 3 is done with a belt with length of 2100mm and for the transmission from 1 to 4 this is a belt with a length of 2520mm.

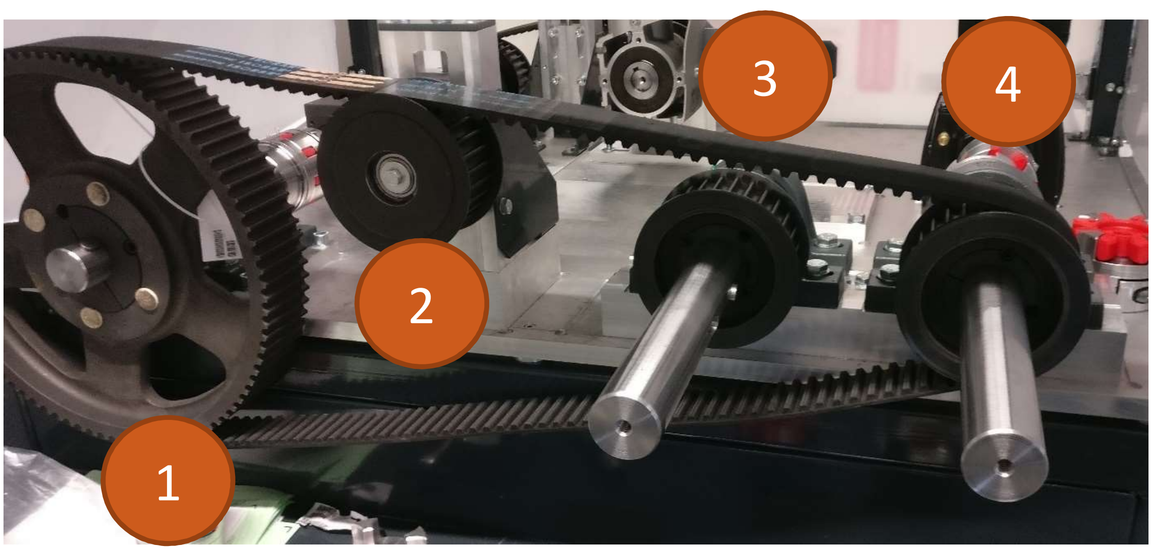


Figure 26: Transmission belt fixed motors

How to loosen the tension on the belt is explained in 4.4. Afterwards the axles and the tension should be put on again. To do this the axle block for fixed motors should be pushed to the right while the tension unit is at its lowest position. This however is not possible with human force due to the tension that will be put on the belt. The explanation of this error is found in the distance between the pulleys. For two pulleys with 90 an 29 teeth and belts of 2520mm and 2100mm with a pitch of 14mm the heart distance is 618.6mm and 832.4mm respectively as calculated with equation (3). The measured heart distances however are 625mm and 845mm.

This can be solved in three possible ways. The first method is using a longer belt. This can only be done when both belts can be replaced with belts equally longer. After looking at Table 3 the larger belts would be 2240 and 2660 (not 2590 because the replacements should both be equally longer). After calculating the new heart distance with equation (3) the extra clearance would be 70mm [28].

Table 3: Possible belt lengths [28, p. 65]

SILVER 2 14M	
Code	**Pitch length [mm]**
1960 SLV2 14M	1960
2100 SLV2 14M	2100
2240 SLV2 14M	2240
2310 SLV2 14M	2310
2380 SLV2 14M	2380
2450 SLV2 14M	2450
2520 SLV2 14M	2520
2590 SLV2 14M	2590
2660 SLV2 14M	2660
2800 SLV2 14M	2800

Another possibility is to change the pulleys to a smaller diameter. Pulleys for the same belt have a different number of teeth. The smallest possible pulley for a belt with a pitch of 14mm is a pulley with 28 teeth [29]. Using a pulley with 28 teeth instead of the 29 gives 3mm more margin.

$$C = \frac{1}{4}(L - \frac{P}{2}(Z1 + Z2) + \sqrt{(L - \frac{P}{2}(Z1 + Z2))^2 - 2 * (\frac{p}{\pi}(Z2 - Z1))^2} \qquad (3)$$

With C = Heart distance (in mm)
L = Belt length (in mm)
P = Pitch (in mm)
Z1 = number of teeth pulley1
Z2 = number of teeth pulley2

When looking at [29] for a smaller pulley to replace the pulley with 90 teeth with the closest pulley is one with 80 teeth. After comparing the new heart distances (from equation (3)) with the old an extra clearance of 40mm is provided for changing the pulley.

Changing pulleys is highly discouraged since it effects the gear ratio and the fixed motors where chosen for that gear ratio.

The third solution does not make it possible to move the fixed axle block to its reference position to be connected with the fixed motors. This solution is to change the mounting plates of the fixed motors to fit with the axle block while it is not in its original position. This option is chosen for its simplicity and because it does not influence other choses. It does influence the tension unit because it lowers the useful stroke, which was solved by increasing the height of the tension unit.

5. TEST PROTOCOLS

5.1. Motor characteristics

The motor characteristics test is in DOL. The motor, physically attached to one of the fixed motors (depending on the power of the motor as described in 3.1) will be switched on with a relay when all the information is added and the start button is pressed. When the motor has reached a stable speed the first measuring point will be taken and the fixed motor will generate a load by increasing the torque. This will decelerate the motor and every time the speed is stable all necessary values will be logged (current, electrical apparent power, electrical real power, mechanical power and speed) to make the graph (Figure 27 shows a possible motor characteristics graph. This will continue until the overload on mechanical power is reached. After this, the relay and the fixed motor stops (in that order). Due to the mechanical proportions of this testbench, the motor characteristics at low output power are not reliable because the fixed motor does not function as a brake at that moment but helps to deliver power to compensate for losses.

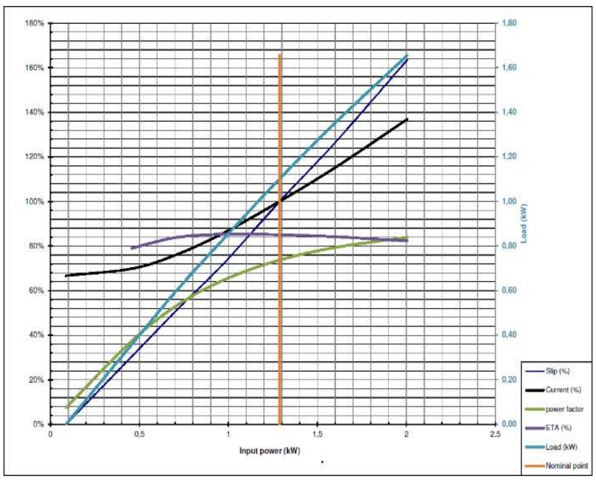


Figure 27: Expected motor characteristics graph [7, p. 2]

5.2. Torque-speed characteristics

There are two possible methods that could be used for this test. These are the direct measurement method and the acceleration method. With the direct measurement method, the motor will be held back to every desired speed to do a measurement of torque. This can be done by setting the fixed motor in speed control. The peak torque that the fixed motor delivers is a multiplication of the rated torque. Since the torque is limited to (126Nm x 3,1=) 390Nm and the highest torque in the list of motors (Appendix A) is 500Nm. Hence, it is not possible to do a torque-speed test for high torque motors. The acceleration method is the preferred method for high torque motors because this minimizes the torque on the belt. This method lets the motor accelerate and measure the speed, torque and current. To have enough measuring points the motor should have a load to increase mass inertia as can be seen in equation (4).

$$T = I * \alpha = I * \frac{s}{t} \tag{4}$$

T = Torque
I = Moment of inertia
α = Rotational acceleration
s = Rotation speed
t = time

The torque of the motor at a given speed is fixed. If the moment of inertia is larger, then the rotational acceleration will be slower and there will be more time for the next measurement.
The total torque that the motor delivers is the sum of the rotational torque and the torque on the torque sensor. The moment of inertia is then calculated from all the parts before the torque sensor. With the current testbench, with the high amount of weight between the motor-on-test and the torque sensor, is it necessary to measure this moment of inertia.

This thesis will use the direct measurement method. This method takes more time which means there are more measurements for the torque-speed curve. With the acceleration method, one inaccurate measurement is visible on the curve which is not with the direct measurement method.

5.3. Efficiency test

The motor is powered by the drive for this test. The motor runs at a set speed, while the fixed motor increases in torque. With every increase of torque the electrical apparent power, electrical real power and the mechanical power will be measured. When the maxed overload mechanical power is reached the fixed motor will idle and the tested motor will increase his speed to do the same at a higher speed. This till the maximum speed is reached. This test will return the efficiency of the motor when it is powered by the drive for every possible speed and torque. The maximum permitted torque that the drive of the motor-on-test delivers is set at 10 times the nominal torque, this is done because it was not possible for a 25Nm motor to get everything in motion due to the high moment of inertia and friction of the total testbench.

5.4. Endurance test

For the endurance test, the motor will accelerate and decelerate for multiple minutes, hours or days following the speed curve in Figure 28. The motor will accelerate for a time (t_a) untill it reaches the set speed (V_{max}), stay at this speed for a set time (t_c) and afterwards decelerate for a set time (t_d). This will repeat after time (t_s) untill the total time has elapsed. From t_a till t_d there is a load set on the fixed motor which can be chosen in Nm. The Temperature, current, torque and speed are logged for this test.

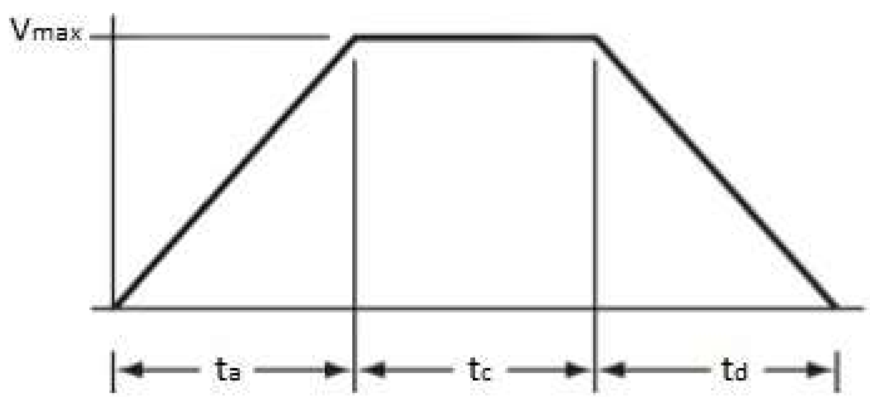


Figure 28: Speed curve endurance test

6. PROGRAM

This chapter will discuss the software for the PLC. The structure off the program can be seen in Figure 29 and will be explained in 6.1.1 till 6.1.5 and the last part (starting from “Excel file”) will be discussed 6.2.5.

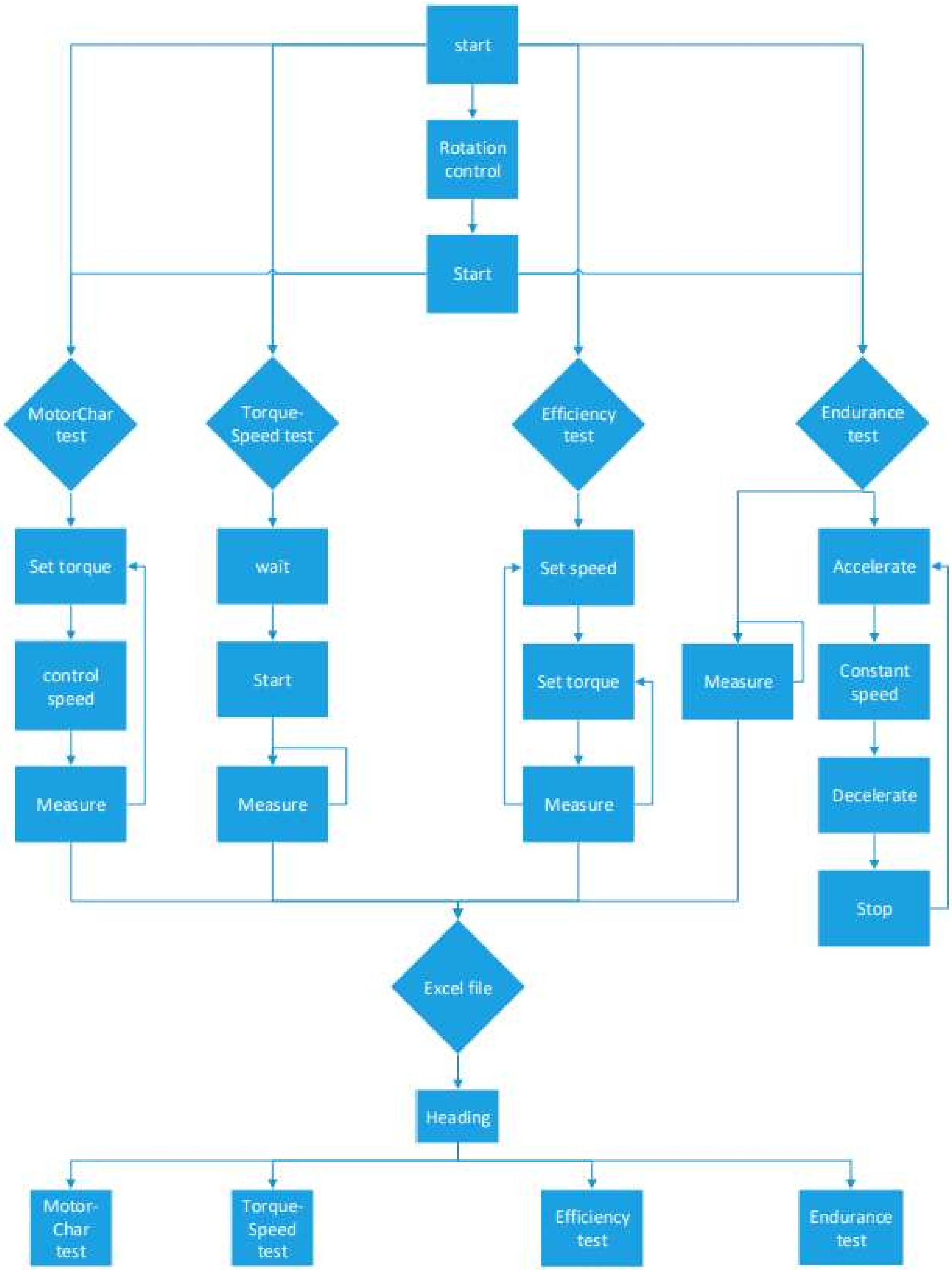


Figure 29: Program structure

6.1. Codesys

The PLC software for a MOVI-C controller has to be written in Codesys. The program will be divided into multiple POU's. these POU's are the main POU and one POU for each test. The program code is added in Appendix C. POU’s start with an action “Init”. The “init” action of the test protocols will reset all the variables for that protocol. After initialising, the main action of that POU will be called and the action “Init” will be bypassed for the next program cycle. All test programs end with printing the measurements and stopping the test.

All variables that can be used to tune a test can be found in the global variable list “GVL_Test”. With these variables you can:

- Increase the number of measuring points,
- Change the minimum torque of the fixed motor to help getting the testbench in motion,
- The overload factor for the motor characteristics test,
- Change the torque sensor.

6.1.1. Main

The main program will read all the variables from the user interface. And use to start the right test. A case setup is used to call the POU for the test that is to be performed. The main program contains a start-stop function, a function to test the rotation direction of the motor, a part to refactor the speed and a main function block with the case structure. The HMI (PC) can have a bad connection with the PLC, this is noticeable due to the parameter “Gvl_test.TestSetup” that sometimes changes value. This is solved in the main program as an if-statement that checks if the parameter still has a value that is within the used range (0-4).

The test to check the direction of rotation (rotation control in Figure 29: Program structure) the motor will be powered at 10RPM with a positive speed. If the motor is not powered by the drive it will also start since the relays will be switched on to turn off the save torque but this will be at nominal speed. The speed and direction of the motor can be measured on the encoder of the fast or slow fixed motor, and should be positive. If it is not positive the speed should be inverted and the direction of the load should be inverted. The torque is also measured and inverted if the value is not positive, this is so the torque sensor does not have mounting directions.

A torque-protection method for the fixed motors and the belt is implemented in the high prior. If the maximum allowed torque for the used fixed motor is measured on the torque sensor (155Nm or 390Nm), then the program will be stopped. This is further discussed in 7.4.1.

6.1.2. Motor characteristics

The main action uses a case structure to set the load, measure speeds and write the measurements to their arrays. This Case structure can be considered a loop where the parameter “Measuringpoints” is the times the loop is executed and the parameter “MotorCharLastRow” is the number of times the loop should be executed. The first case sets the torque and the second case copies the speed. The third case waits for a certain time to go to the fourth case. At that case the speed before and after the time are compared with each other. If those speeds are the same (with a certain margin) then the motor is not accelerating anymore and the measurements can be done. These measurements happen in case five. After the measurements, the loop will start again or the test will be over when all measurements are done.

6.1.3. Torque-Speed characteristics

Inside the program “TorqueSpeed” there is a case structure. This is the only test where the case structure is only run once. The test begins with initiating the fixed motor. One cycle later the motor-on-test is switched in DOL and a timer is started. When the timer is over the test begins. This timer is necessary for the fixed motor to allow for the torque to build up. After this time the fixed motor will accelerate at an acceleration of 5 rotations x min $^{-1}$ x s^{-s}. This gives a number of measurements that is equal to five times the nominal speed. The test is done when the nominal speed is achieved, or the array to log all the data is full.

6.1.4. Efficiency test

The Main action in the Efficiency test has a double loop in the form of a case structure. In the first case, the speed is set on the motor-on-test. In the second case, the torque is set on the fixed motor. Because of this, the load for the motor-on-test can be adjusted while it keeps running at the same speed. The third case looks if the setpoints are reached. When the setpoints are reached the measurements will be done after a certain time and the measurements itself are an average over a certain time. This loop will continue for the total range in speed and torque of the motor that has to be tested.

6.1.5. Endurance test

In the endurance test cases one to four represent the four stages in 5.4. in every case the setpoints of acceleration and speed are set to follow the curve in Figure 28. The load is turned off when the motor-on-test is not running, otherwise the motor would turn in the opposite direction and the breaking resistor of the drive for the motor-on-test would be used unnecessarily.

6.1.6. Codesys HMI

The HMI is used to start the program and change data of the motor, this data is used in the program and in the test report. All HMI screens can be seen in 7.4. The HMI contains labels, text fields, buttons and lamps which all can be changed in location and size. The only other variables for the labels that are used are the font and the text value
The lamps were coupled to a Boolean value (Start, stop and reset) and the background image was changed for different colours.

The buttons are matched with their variables in the input configuration of their properties. With “Tap” a selected Boolean variable will be set TRUE when it is clicked and will set FALSE when it is not clicked with the mouse. This is inverted if “Tap FALSE” is TRUE. Another method that was used to write parameters with buttons is “OnMouseDown” with this method the parameters will keep their values, even if the buttons is not pressed anymore. Multiple input possibilities are used for this program. Figure 30 shows all possible input configurations. “Execute ST-Code” is used for the buttons that chose the test. This is because these buttons hange more than one parameter. “Change shown visualisation” is used for all the buttons that open other screens. There are other methods available for programming an HMI that have the same results, there is no why these where not implemented other than that the implemented method also works.

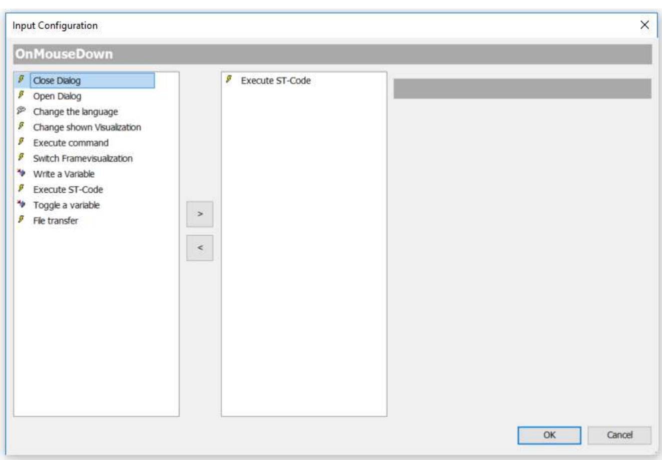


Figure 30: Codesys HMI OnMouseDown configuration

Just as with the button and the lamp is it necessary that the text field is coupled to a variable. This variable can be linked to the field under the tab “Text variables”. To show the text behind the variable in the HMI with formatting sequences that are listed in [30]. This formatting is done at the parameter “text” under the tab “Texts” in the properties of the text field. The formatting that is used is “%s” to show a string, “%d” for an integer, “%2.2f” for a real number with the decimal number and the whole number existing out of two numbers.

6.2. Coupling

The best ways of exporting data from 2.3 will be discussed in full in this chapter. After testing them the best method was selected.

6.2.1. Movisuite scope

Movisuite scope is a developing tool running on the drive from a different program. It has only access to parameters on the drive and cannot communicate with the Codesys program.

Two tests where done to determine the possibilities of the Movisuite scope tool. The first one was with the standard measuring values (Appendix D). The second one with three current which were expected to be sinusoidal (Appendix E). To limit the pages of the appendices they only contain the interesting measuring values.

The first test was running the motor at certain rpm's for 6 minutes with a sample rate of 1ms. The speeds were set at 66, 99, 132, 165, 198, 231 and 264rpm. Table 4 shows the acceleration from 0 to 66 rpm. In this table, "nr" is the number and "x" is the time corresponding to the measuring points. The time contains the day and the hour. The hour corresponding with test value nr 7762 is 34min, 45s 794ms and 999,4µs. The time difference between measuring points nr 7764 and 7765 is 4 minutes 1 second and 994 ms which means that this part of the logging is not accurate.

Table 4: Appendix D Movisuite test 1 acceleration 1

Nr	X	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9
7762	0001-01-01T00:34:45.7949994	0	0	0	0	0	578.393	1	10	1545	0
7763	0001-01-01T00:34:45.7959999	0	0	0	0	0	578.393	1	10	1545	0
7764	0001-01-01T00:34:45.7969993	0	0	0	0	0	578.393	1	10	1545	0
7765	0001-01-01T00:38:47.7909997	63	66	0	0	0	570.067	1	14	1039	0
7766	0001-01-01T00:38:47.7920001	63	66	0	0	0	575.039	1	14	1039	0
7767	0001-01-01T00:38:47.7929995	63	66	0	0	0	575.552	1	14	1039	0

A similar error has happened 17 more times in the first test. Three of them are added to Appendix D. The total time not registered is 1 minute 11 seconds 954 ms. This error is due to fact that the measuring was in continuous trace mode and the motor was controlled by the PC instead of the PLC. This methods sends and receives more data than the ethernet connection can handle, therefore the scope function has to pause sometimes. This error also happens with the motor being controlled by the controller.

The acceleration from 66rpm to 99rpm is shown in Figure 31. The overshoot is maximum at a setpoint of 98rpm where the speed is 116rpm. It also takes 405 samples to reach the setpoint speed. How this actual speed is acquired is not sure since the motor did not have an encoder. Here you can see that the speed is adjusted as a PID controller.

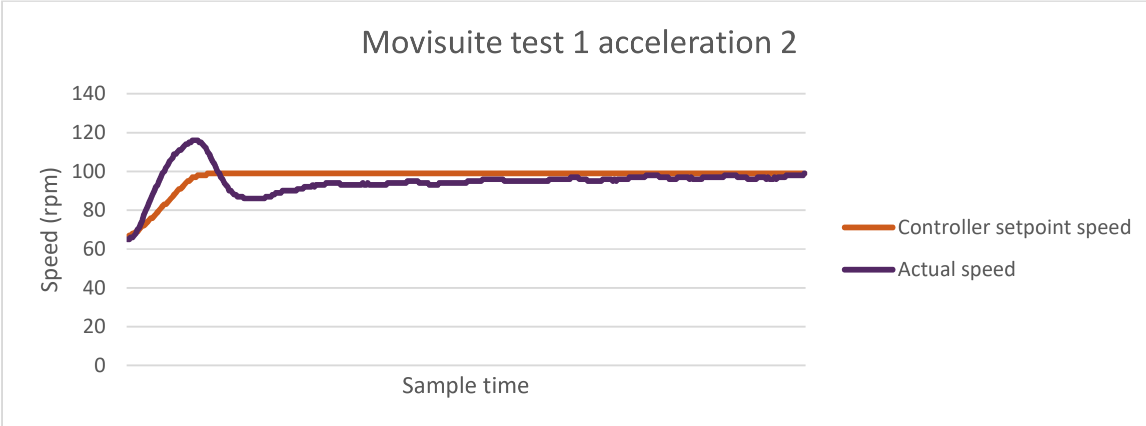


Figure 31: Appendix D Movisuite test 1 acceleration 2

For the second test Movisuite generated much higher values. The maximum registered value in Figure 32 is 21587 %NominalCurrent. What this value means is not known since it is not yet in the parameter description [31]. Derived from the name and other parameters that use percentages it seems that the unit is 10^{-3} or that 21587 equals to 21.587%. Which is an expected value since the motor is running without a load.

The problem that occurred in this test is that not all values are correctly exported to Excel. The values of the first 50 samples were manually corrected to the correct value. With the second half of the values, no correction was implemented. The necessary correction is multiplying with 1000 when there is a decimal point.

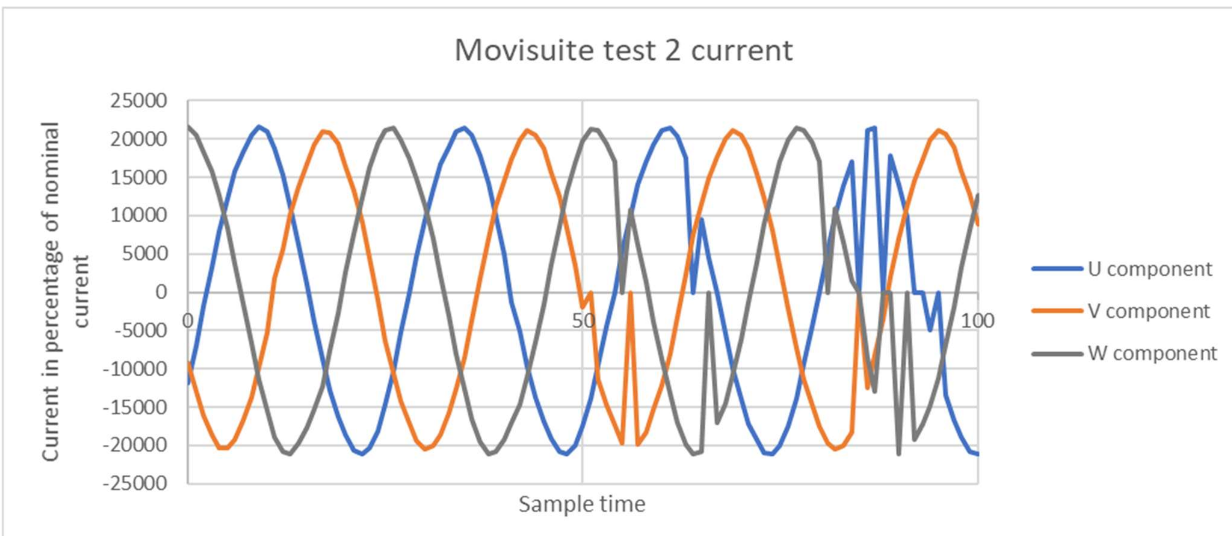


Figure 32: Appendix E Movisuite test 2 current

This error can be avoided if you set the decimal separator on System instead of MOVISUITE language (in Figure 33) as turns out by [9]. There is nothing different, except that all values have a decimal point instead of some values.

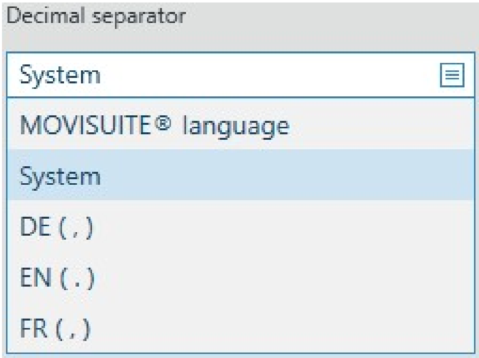


Figure 33: Movisuite scope settings

This method is deemed not sufficient for multiple reasons. The first reason is the missing parameters. The missing parameters can only be avoided with 100% certainty when “Single trace mode” is used. The second reason is that some parameters are exported badly if the wrong decimal separator is used. Reason three is that another method would be needed to export the other parameters since not all necessary parameters are stored inside the drive. The last reason is that this is a time-based logging system and some test protocols need a measuring at unpredictable moments. The first two reasons make this method unreliable, the other reasons make this method not suited for this application.

6.2.2. Codesys trace

This method is like Movisuite a time-based logging system. Unlike Movisuite it can be started from the code and can export every parameter that the plc has access to. There are three reasons that make this not ideal. The first one is that it takes some handling to get the Excel file. The second reason is that it can only extract numbers from the plc and not text. The last one is that it is also time-based. Excel could identify the correct times for not time-based tests but this would result in a more difficult Excel macro.

6.2.3. Scripting

With scripting, every parameter can be written to Excel, even parameters that contain text. Therefore all parameters can be filled in the visualisation screen of the PLC even do they are only used for the test rapport. Like the Codesys trace tool to extract parameters external code would be needed to use the data. Python is easier to program in then Excel because Excel has different standard formulas for each language. There are even different formulas for use when writing macros compared to plain Excel formulas. An example is the function TEXT() in Excel. When you are using Excel in English it is used to change a number saved as text to a number. When you want to use this in a macro, the function is named FORMAT, and in Dutch the function is called TEKST. This makes it difficult to start writing macro's in Excel.

Every function has to be defined In a script before it is used. Therefore you will be programming from the bottom up or with one long function. Multiple functions are preferred for an easier understanding of the code. Therefore the script will start with the functions to extract the values (depending on the test), afterwards the function to make a header will be defined. And the script ends with making a connection to the Codesys program. This can be seen in Appendix F. This script, however, takes a lot of time to make this csv file.

6.2.4. OPC server

The OPC server has advantages and disadvantages as every methode. The Advantiges are already discussed in 2.3.3. The main disadvantage is that the controller has to stay connected to a PC over ethernet. This ethernet connection can be overloaded which results in a loss of data. It was also not recommended by [10] to use this methode.

6.2.5. Writing to Cfast

This topic will explain how to write to the Cfast card and what value it has over the other methods. The "SysFile" library is necessary to write the documents. With "SysFileOpen()" an Excel document can be opened, this function needs three inputs. These inputs are the filename, the access type (read, write or append) and a pointer to the error code. The output is a DWord that handles the file. Writing text to an Excel file can be done with another function named "SysFileWrite". The write function needs the file handler, the text to write as a pointer, the length of the text and the pointer to the error code. With "SysFileClose" the Excel document can be closed again and is available on the Cfast card. The separation of the cells is separated with " ; " and the next row can be started with " RN ".

This is almost the same as the scripting. The differences are that with scripting the Excel is created from the PC where this method the Excel is created by the controller. To get the file on a PC, one will still need their PC the connect with the PLC to extract the file. For scripting, this has to be done the entire time while for writing the Excel file on the Cfast card you only have to connect when you need the file. The advantages of this method over scripting is that it is much faster and can be done by the controller. How this is implemented in the Codesys program can be seen in Appendix C (Write_To_Cfast (PRG)). This POU has 5 actions. One for the header and one for each test.

6.3. Excel

When the controller has made the Excel file containing all the data it has to be imported in Excel for visualisation.

The data will be imported with the use of a macro (This macro is written in Appendix G). Actions can be recorded or programmed in a macro to be used multiple times. One cannot use the macro recordings immediately when using this to make a graph or to import a csv document. This is because the macro refers to the name of the connection with the csv document or refers to the name of the graph.

The program (visualised in Figure 34) starts with opening a pop up to select the Excel file that was created by the controller. All Excel sheets except the one called "Protected Sheet" will be deleted, otherwise an error will occur when Excel tries to create a sheet that already exists. Because importing the test data in Excel is done by queries, and there can only be one query of each name, the function "DelQuery" is called to deleted the name of the query that will be used. This function will check if there is a query with the name of the file and delete that query. The program continues with loading the data from the file. Next, the sheet for the visualisation of the test data will be made. At last, the name of the imported test (printed in cell A5) will determine what last function will be executed. This last function rearranges the values (from mA to A and mWatt to Watt for example) checks what cells to use for the graphs, changes the layout where it is necessary and makes the graph.

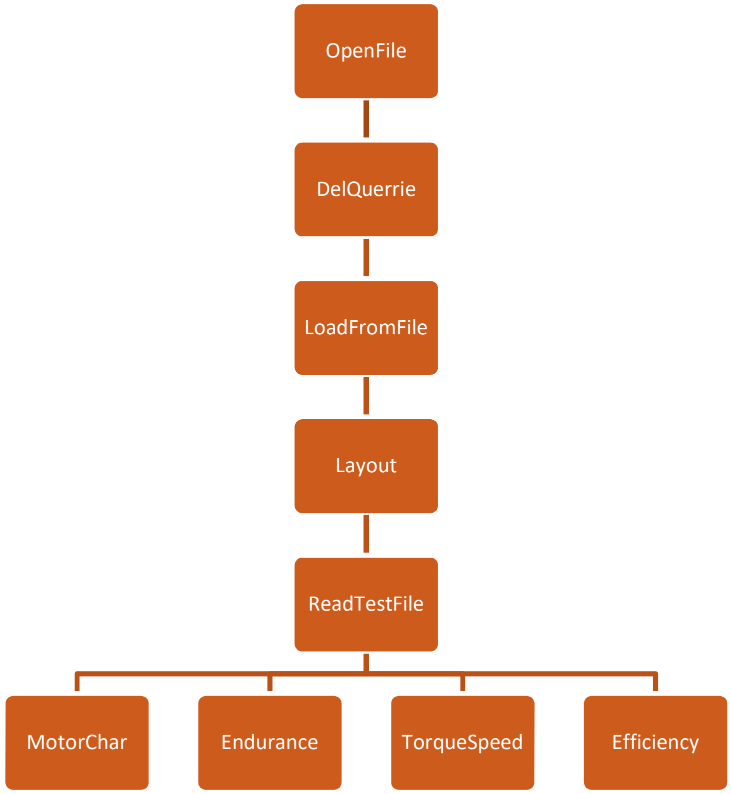


Figure 34: Program layout Excel

7. INSTRUCTION MANUAL

Be sure you have Movisuite SP10 installed on your PC before the start of the test. An installation file of SP10 can be found on the CD-ROM. The program can be found on the CD-ROM and should be imported in Movisuite. Importing the program inside Movisuite is done by clicking in the upper left corner when Movisuite is opened (as seen in).

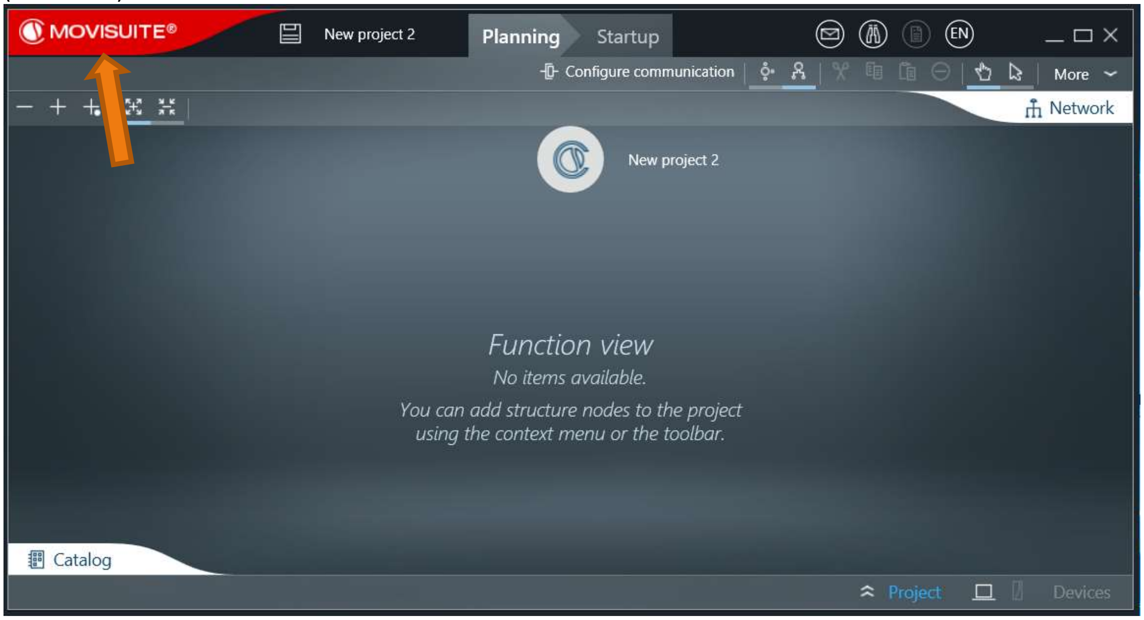


Figure 35: Importing program step 1

Afterwards go to the tap "import" (as seen in Figure 36) and select the import file that is located on the CD-ROM.

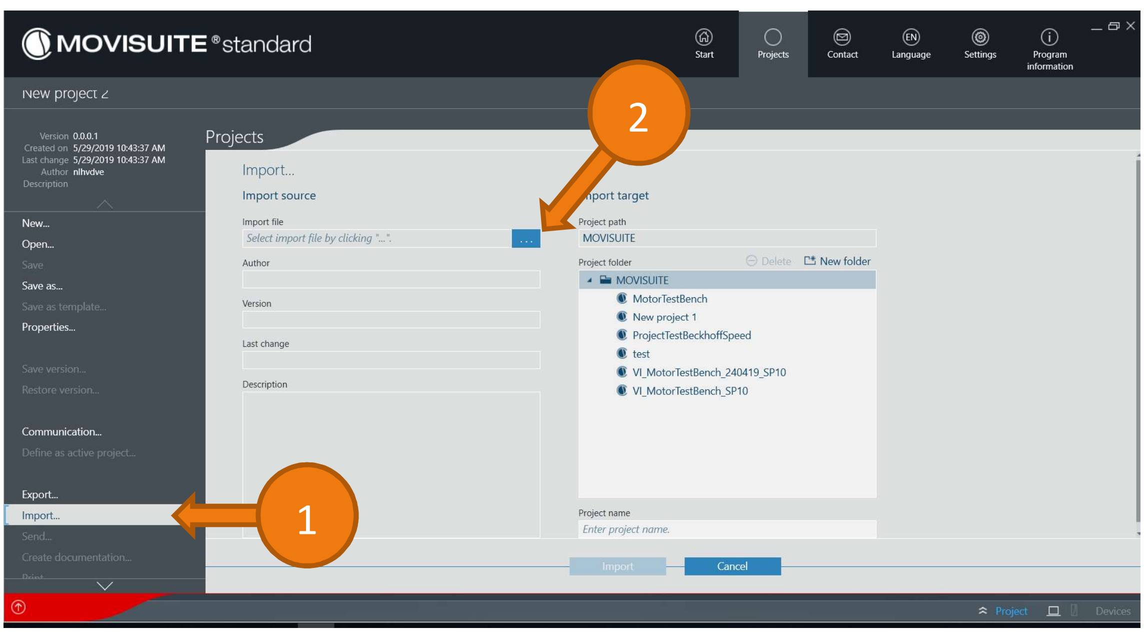


Figure 36: Importing program step 2

7.1. Switching motors

The first step to switch motors is to secure the main switch for safe working and disconnect the power cables. The second step is loosening the tension on the belt of the motor-on-test. This is done by turning the screw above the tension unit counter-clockwise and loosening the bolts of the motor block/plate of the motor-on-test. When the motor is slid to the right the transmission belt can be taken off the pulleys. To get the motor of the motor block it is necessary to place them both on the ground (this is a two-man job). Install the motor on the motor plate/block and place it back on the testbench (this is a two-man job). The two yellow wires to the motor are for lifting the brake.

Depending on the new motor-on-test, a fixed motor has to be coupled to serve as a brake. When the new motor-on-test (1 in Figure 37) has a torque below 155Nm and a speed below 1900 Rpm it has to be connected to the fixed motor for high speeds (2 in Figure 37). When the torque is higher as 155Nm but below 390Nm it should be connected to the fixed motor for slow speeds (3 in Figure 37) if the speed does not exceed 450 Rpm.

Switching the fixed motors is done by loosening the bolt on the tension unit of the transmission belt for the fixed motors. Afterwards disconnect the shaft coupling of both motors and then loosening the screws of the axle block (4 in Figure 37) and pushing the axle block to the left. Now the transmission belt can be changed and the motor can be coupled again to the axle. First place the axle block in line with the fixed motor, connect the axle coupling and bolt down the axle block. End with putting tension on the belt with the tension unit.

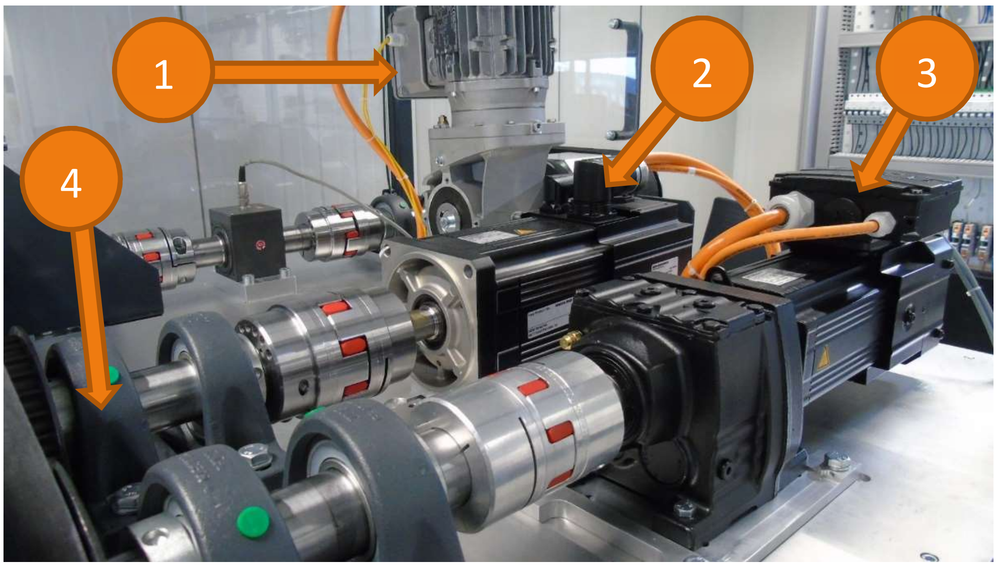


Figure 37: Motors on the testbench

7.2. Installing drive parameters

Installing the drive parameters is only necessary when the motor will be tested driven by the drive. This can be done by opening the configuration settings of the motor called “Nord motor” and going to “Drive train DT1”. The motor and gear unit should be changed. Implementing the settings of a third party motor can only be done by creating the drivetrain manually by catalogue. This can be done by right-clicking the motor and opening the catalogue (Figure 38) or clicking “open menu” and “create drivetrain manually” (Figure 39). There is a subdivision between synchronous and asynchronous motors. Search for third-party motor at the end of the row that belongs to the type of motor that will be tested. After filling in all the data from the motor rating plate and the gear ratio (if needed) then the drive will be ready.
Implementing drive parameters for motors from the same manufacturer (SEW) then the drives is more easily done by using Nameplate/motor label instead of “Create drive train manually” in Figure 39. Here you can type the name of the motor. Motors from SEW can also be used while the settings in the drive are implemented as a third party motor but this can limit the torque or speed of the motor.

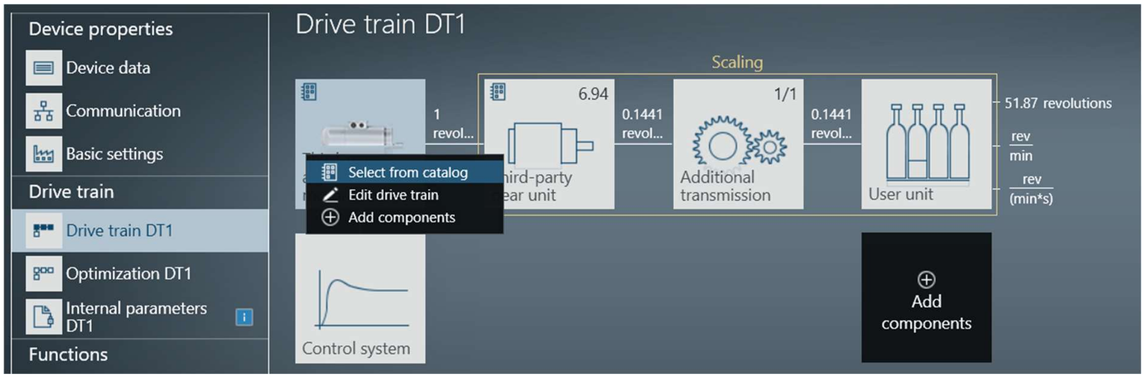


Figure 38: Changing motor drive

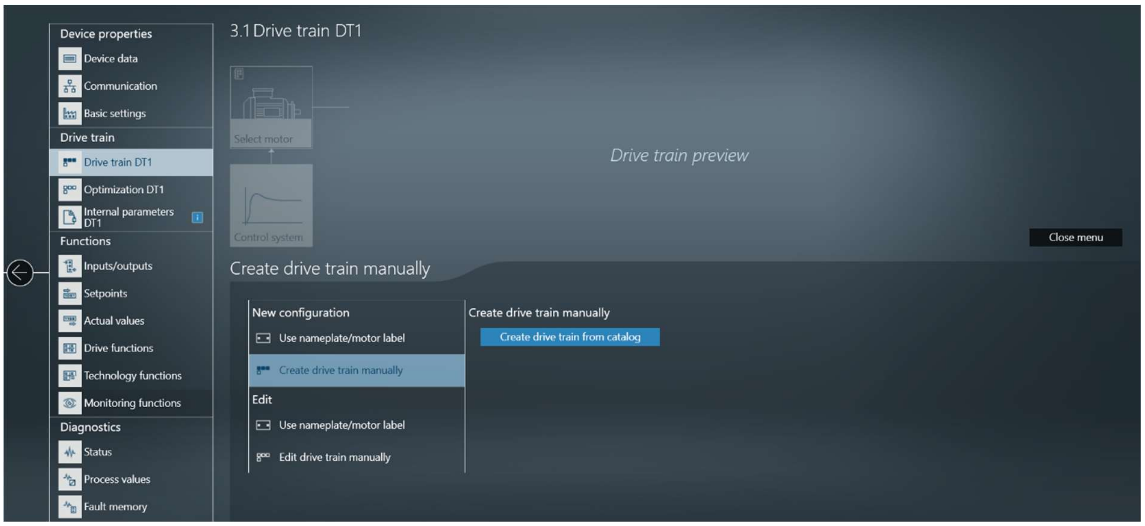


Figure 39: Changing motor drive

7.3. Electrical wiring

The electrical wiring has to be checked to make sure all equipment is connected. This is done by checking the wiring for DOL and with a drive. The numbers refer to Figure 40.

7.3.1. In DOL

For testing the motor in DOL (Torque-Speed test or motor characteristics test) the Beckhoff power module has to be connected. The input (2.1) of the power module should be connected directly to the electrical grid (1). The output of the power sensor (2.2) should always be connected to the relay input (3.1) so that the motor does not start the second there is power. The final connection is the output of the relay (3.2) with the terminal blocks of the motor (4).

7.3.2. With a drive

For testing the motor in combination with a drive (efficiency test or endurance test) the power input of the drive. should be connected with the electrical grid (1). The output of the drive should be connected to the motor (4). The relays or Beckhoff power module are not necessary.

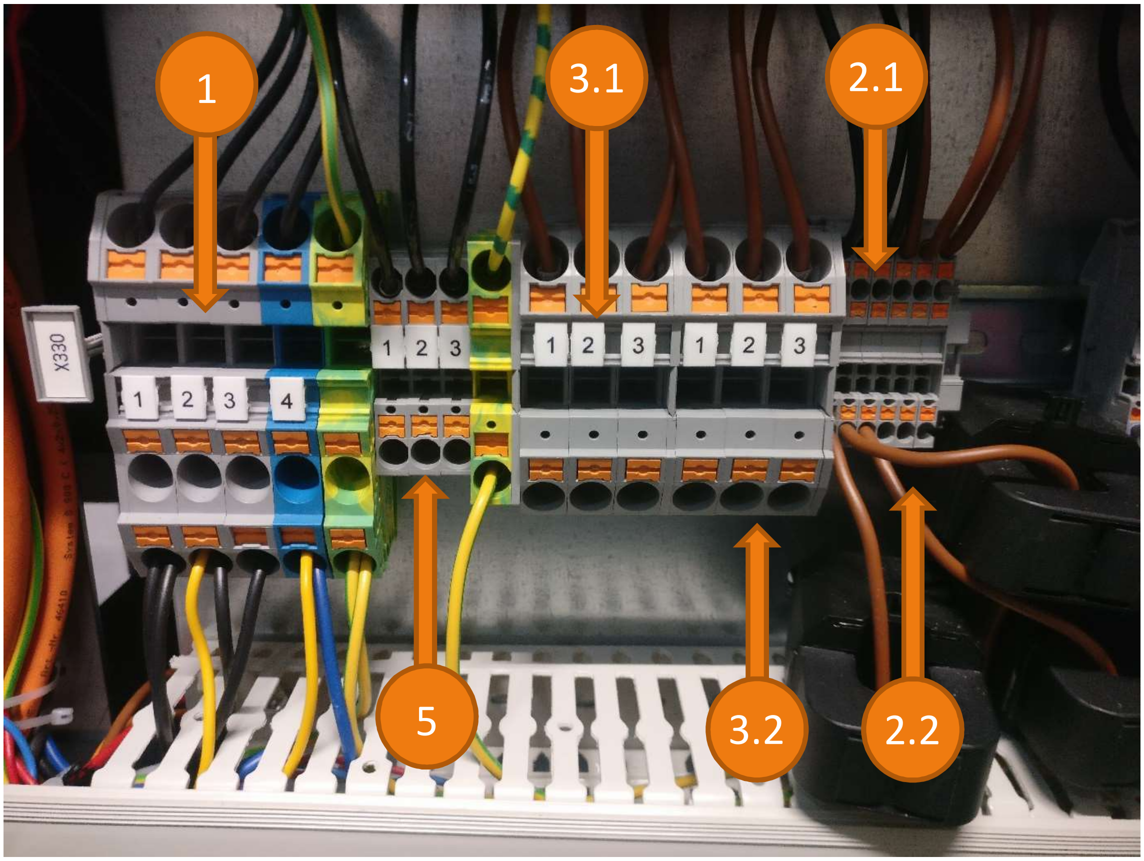


Figure 40: Terminal blocks for electrical wiring

7.4. Starting program

To start the program you will have to open the IEC editor first. This can be done by right-clicking on the controller (The IEC editor can found below "Tools") as seen in Figure 41.

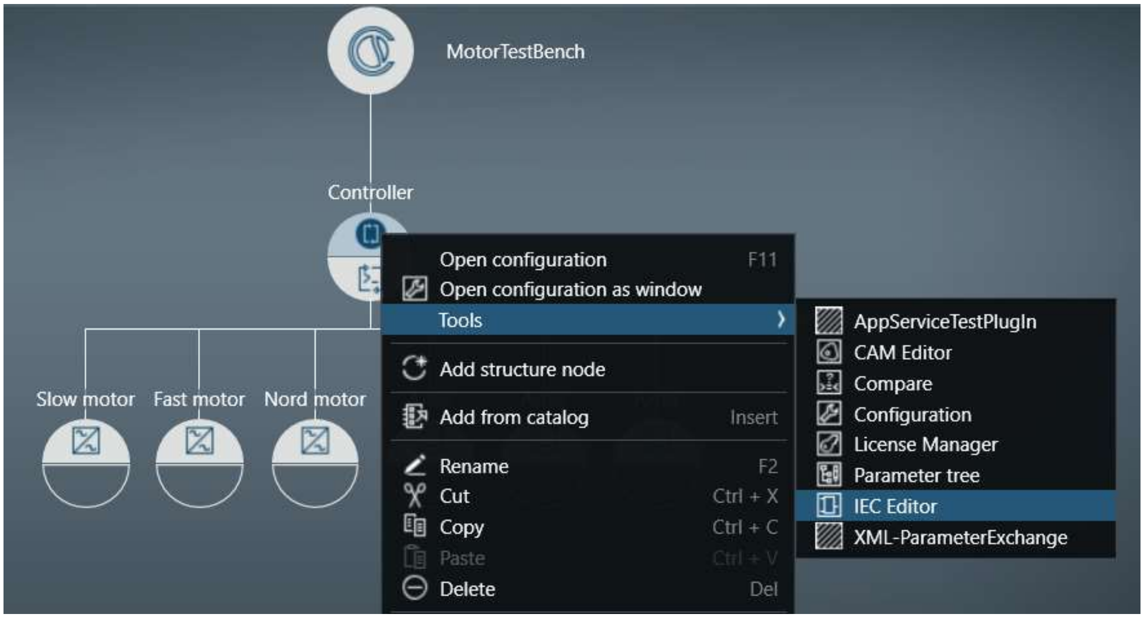


Figure 41: Opening Codesys

Logging in on the controller (Alt+f8 or in the ribbon under "online") is necessary to use the HMI. The HMI screen "Login" (as can be seen in Figure 42) is the first screen where test data has to be implemented. The gear ratio, rpm (without gearbox), and power are necessary for testing, the names frequency and nominal voltage are only used in the test report.

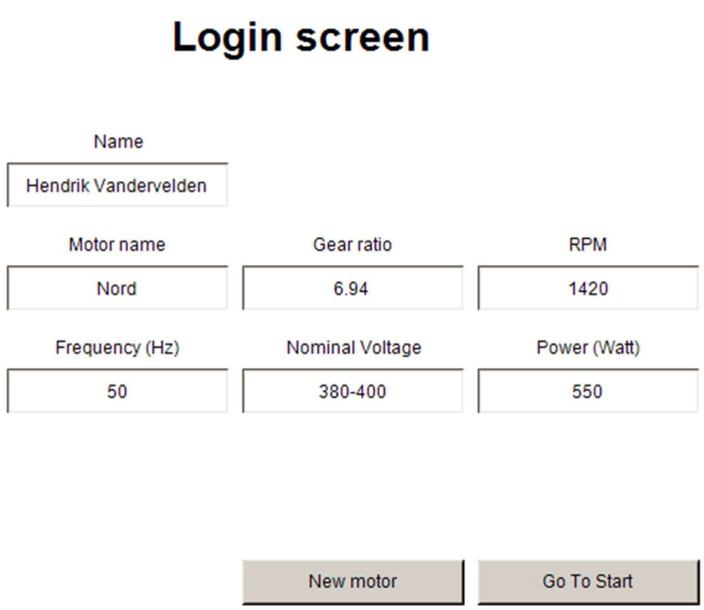


Figure 42: HMI login

With the button "New motor" all parameters about the motor are deleted, these are the parameters in the screen above, but also the rotational direction, torque direction and the connected fixed motor. This has to be done every time a new motor is attached to the testbench. After all the data is implemented you go to the start screen by pressing the button “Go To Start”.

The start screen (as can be seen in Figure 43) is the place to start and stop the test and where a certain test can be chosen. But before you can start a test, the safety relay must be on. The stop lamp (if it is red) indicates that the safety is off or that the stop button is pressed. Close all doors, release all emergency buttons and press the set knop (black button at the back of the emergency button). If there is a new motor mounted on the testbench and the rotational direction, connected fixed motor and torque direction are not yet measured you can do so by clicking start while no test is selected, or selecting the test first and pressing start afterwards, the start button will have to be pressed again after the rotational direction, torque direction and connected fixed motor are measured. The endurance test is the only test that will need more inputs to start the test, if you select that button you will automatically be redirected to the “endurance test screen” which is shown in 7.4.2.

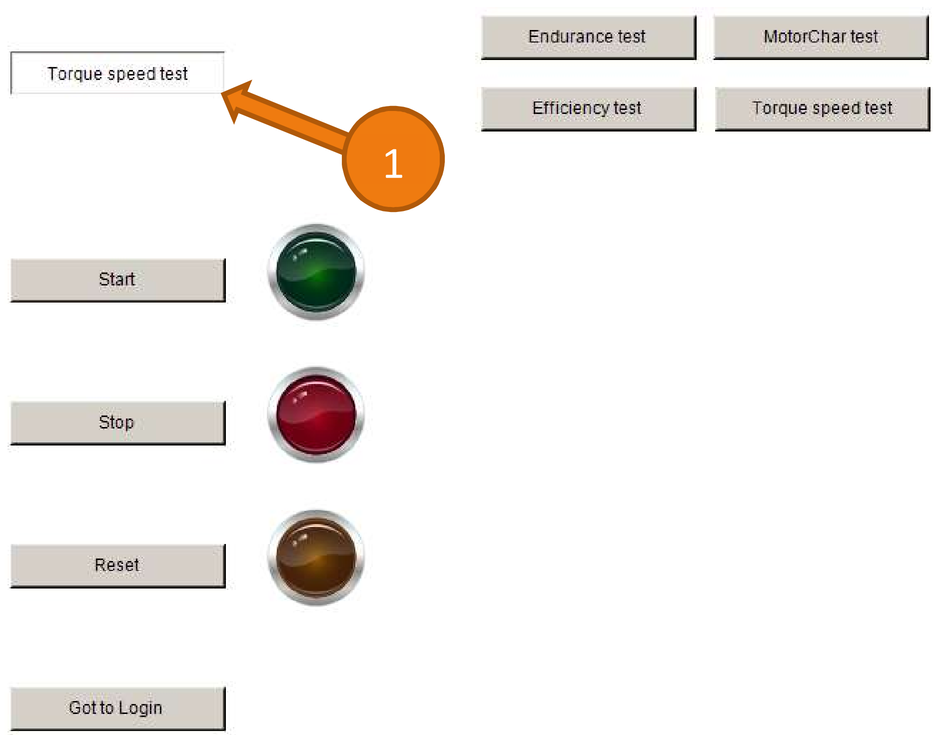


Figure 43: HMI start

7.4.1. Error handling

The selected test can be seen in the text field at the left upper corner in Figure 43 (1). This text field also shows the errors if one occurred. The errors are:

- Error prevention on fixed motor,
- Error on fixed motor,
- Max torque limit reached,
- Tested motor has error.

“*Error prevention on fixed motor*” indicates that the measured torque is too high for the fixed motors. The error prevention can be turned off from the global variable list “Test”. If the error prevention is turned off or does not function, the drive will give a fault error which is shown as “*Error on fixed motor*”. This error needs a reset.

“*Max torque limit reached*” indicates that the maximum value of the torque sensor or the maximum value of the belt was reached. The settings should be lowered so that the torque will not reach the limits any more or, if the torque sensor cannot measure 600Nm (the maximum torque on the belt), the torque should be replaced with one that allows for a higher torque.

“*Tested motor has error*” indicates that the drive of the tested motor returns an error. Check Movisuite for the type of fault. The error can be reset in Movisuite or with a press on the “New motor” button in the login screen.

7.4.2. Input endurance test

The endurance test needs seven more parameters. These are the speed the motor will need to achieve, the torque that will be set as load. Keep in mind that the maximum torque for the fast motor (the one without redactor) is 23Nm and the maximum torque of the slow motor (the one with gear unit) is 74Nm. Also the times it will take to reach that speed, the time that speed will be kept, the time to decelerate to 0 and the time that the motor will stop are all necessary parameters.

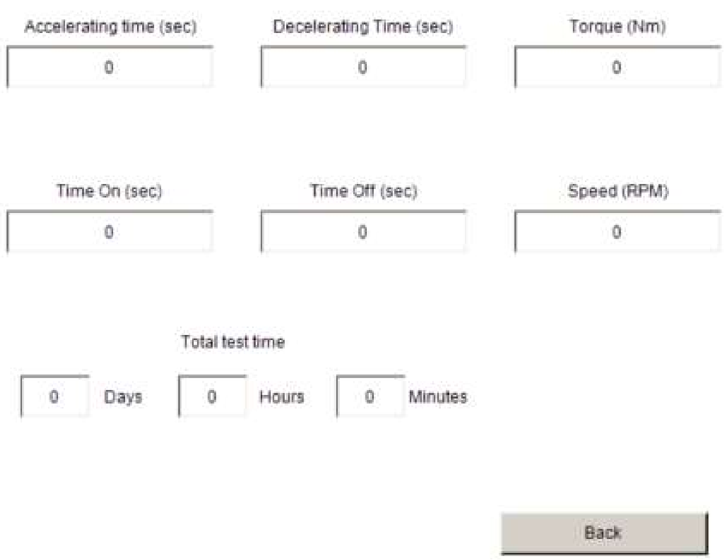


Figure 44: HMI endurance test

7.5. Finishing program

When the test is done open the controller (1) from the device tree and go to "files "(2) as seen in Figure 45.

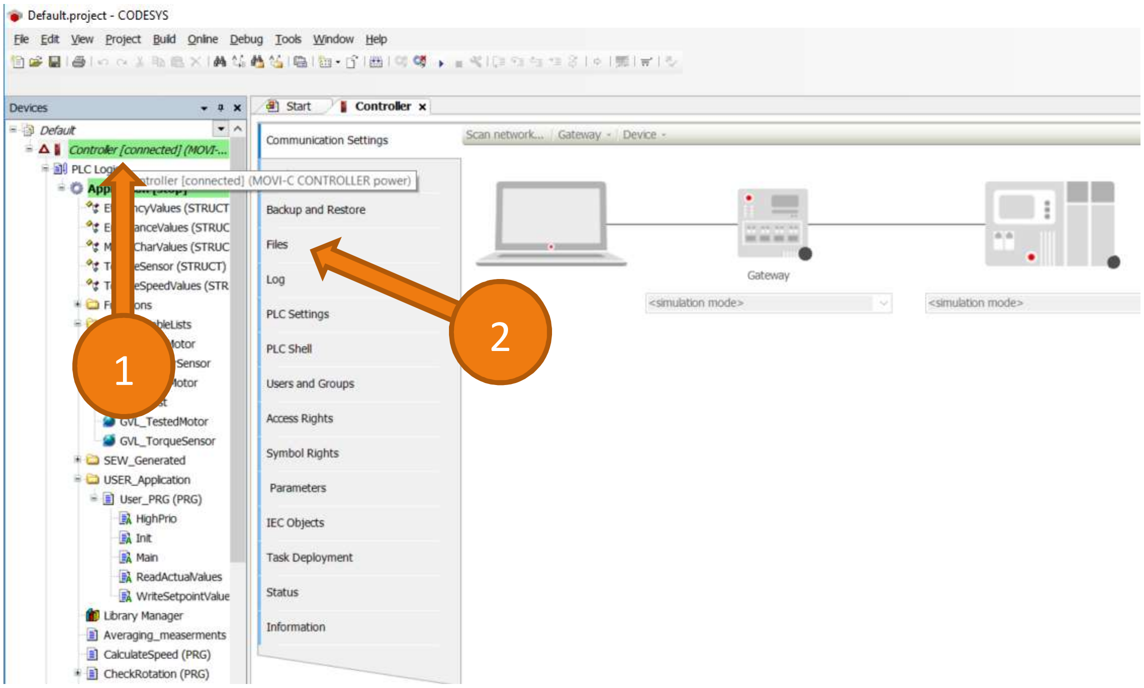


Figure 45: Codesys controller

The screen "files" is divided in two sides as can be seen in Figure 46. Left there is the map structure of your pc, go to a map to place the test data. Right every file on the PLC, click refresh (3) to see the map structure and go to "data". Mark the desired test and click on the arrows (4) to move the file to the pc.

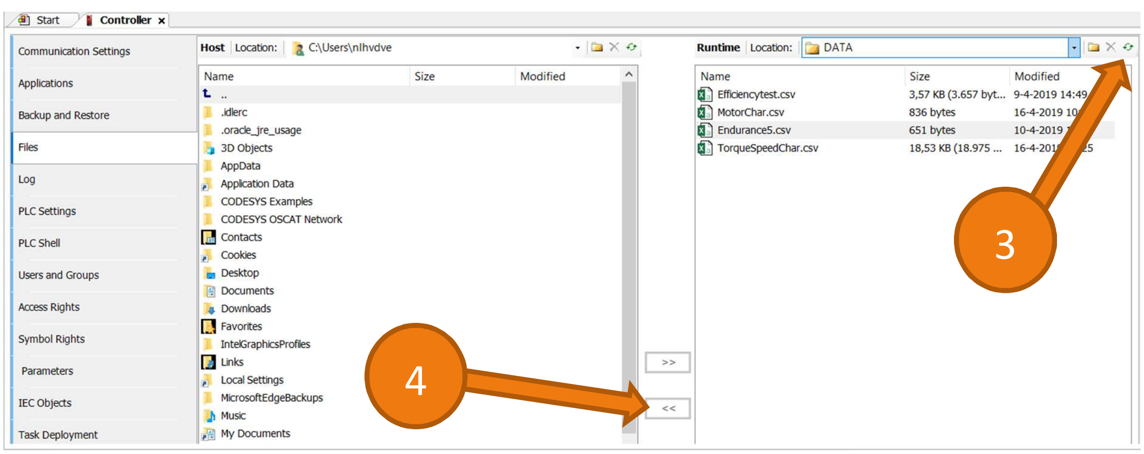


Figure 46: Codesys controller files

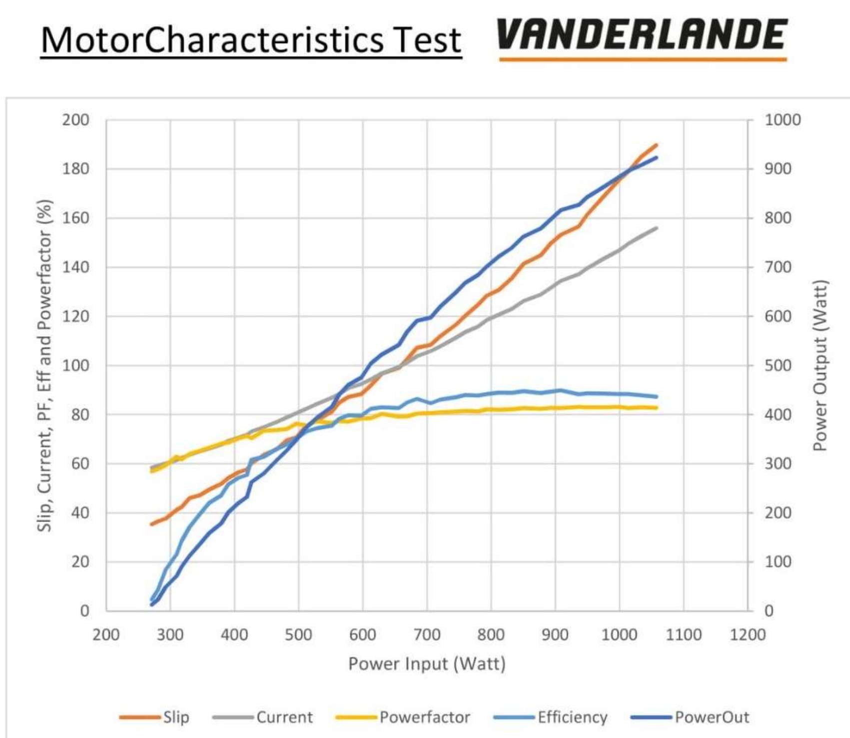
Next step is to open the Excel document and press the start button in this document. This will start the macro to read the test data. First, a file browser will appear where you select the file with raw test data. Afterwards a pop up will appear to ask if all the Excel sheets may be deleted. If you have not saved/printed the test reports press "No", save them and press the start button again, otherwise press "Yes" and the test rapport will be made.

7.6. Interpreting output

This chapter will give information about how de test data is visualised with examples of tests that where conducted with a 550W Nord motor.

7.6.1. Motor characteristics

The motor characteristics show the slip, current, power factor and efficiency in % on the left axis and the power output (in watt) on the right axis. The slip and current are calculated as percentages of the values at nominal output power. The value of 100% slip and 100% current is written in the bottom left field. As is shown in Figure 47.



Test data	
MotorChar Test	
Date	17-5-2019
100% Current	1.989 Amp
100% slip	8.5% Rpm loss

Motor data	
Type	Nord
Power	550 Watt
Motor Gear	6.94
Voltage	380-400
Frequency	50 Hz
Speed	1420 Rpm

Made by: Hendrik Vandervelden

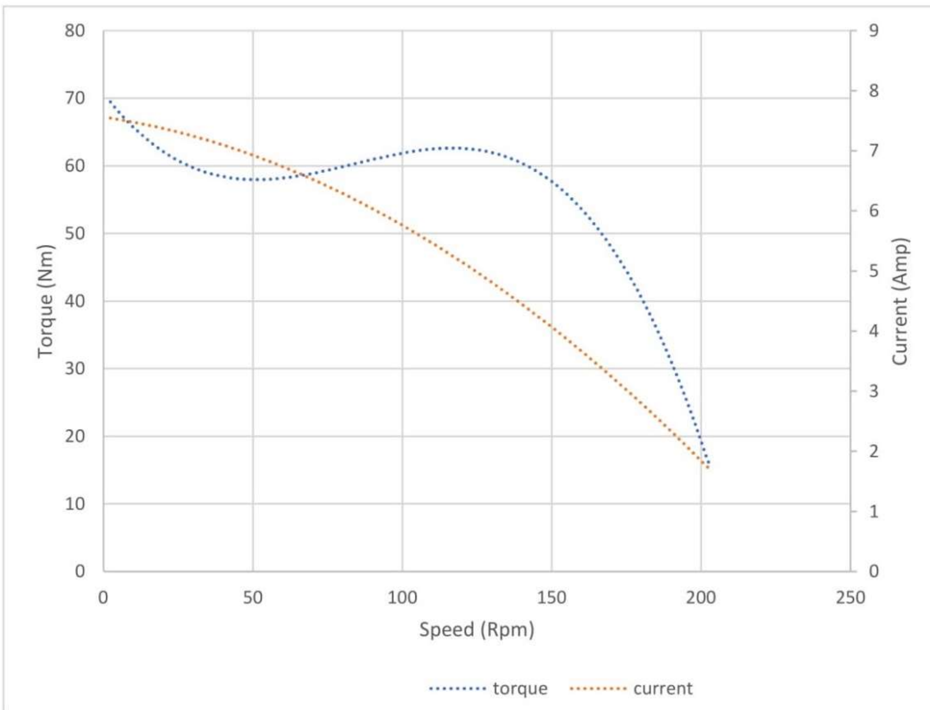
Figure 47: Example motor characteristics test

7.6.2. Torque-Speed characteristics

The torque-speed and current-speed curve are respectively fourth and second order regressions. In the bottom left field is the R^2 value of the torque and current regression. The higher this value (with the max value being 1) the better the measured values fit the curve in the graph. As shown in Figure 48.

Torque-Speed Test

VANDERLANDE



Test data	
Torque-Speed Test	
Date	9-5-2019
Regression torque	0,9856151
Regression current	0,997616

Motor data	
Type	Nord
Power	550 Watt
Motor Gear	6.94
Voltage	380-400
Frequency	50 Hz
Speed	1420 Rpm

Made by: Hendrik Vandervelden

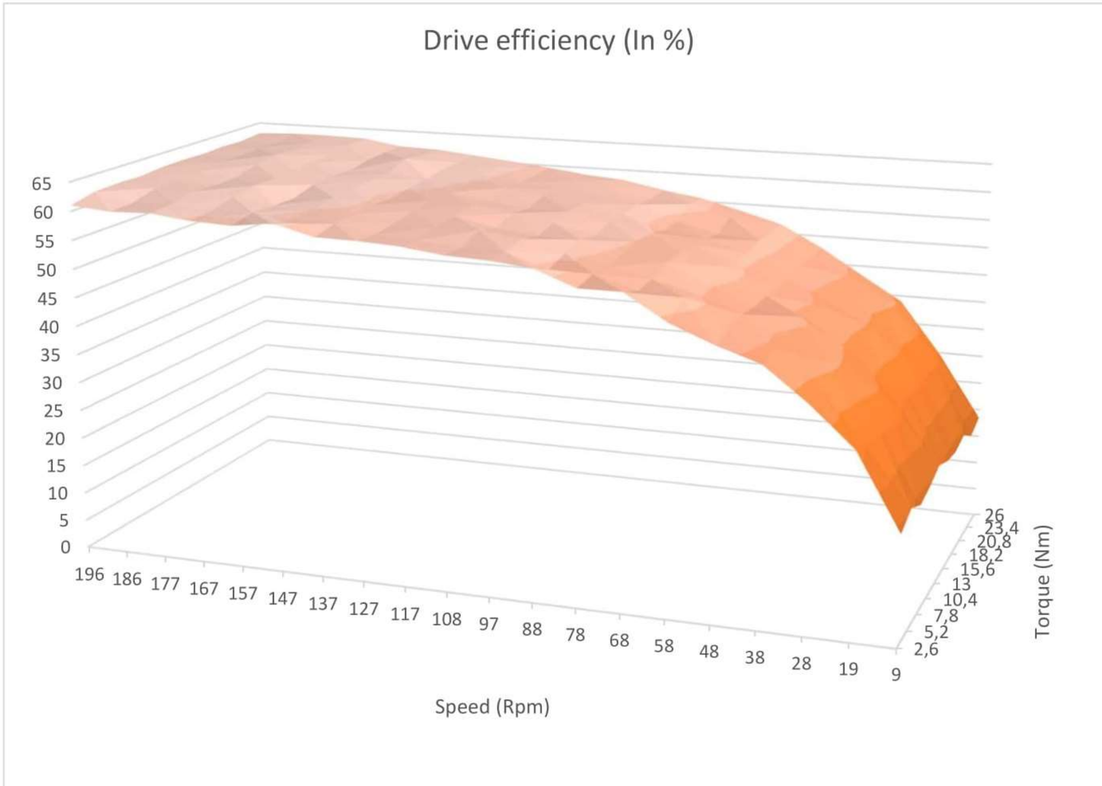
Figure 48: Example torque-speed test

7.6.3. *Efficiency test*

The efficiency graph is a three-dimensional graph. The speed axle of this graph can be inverted if the efficiency is lower at low speeds. This is done to get all the data visible but this can make it harder to interpret the data at first sight. As shown in Figure 49.

Efficiency Test

VANDERLANDE



Test data	
Efficiency Test	
Date	9-5-2019

Motor data	
Type	Nord
Power	550 Watt
Motor Gear	6.94
Voltage	380-400
Frequency	50 Hz
Speed	1420 Rpm

Made by: Hendrik Vandervelden

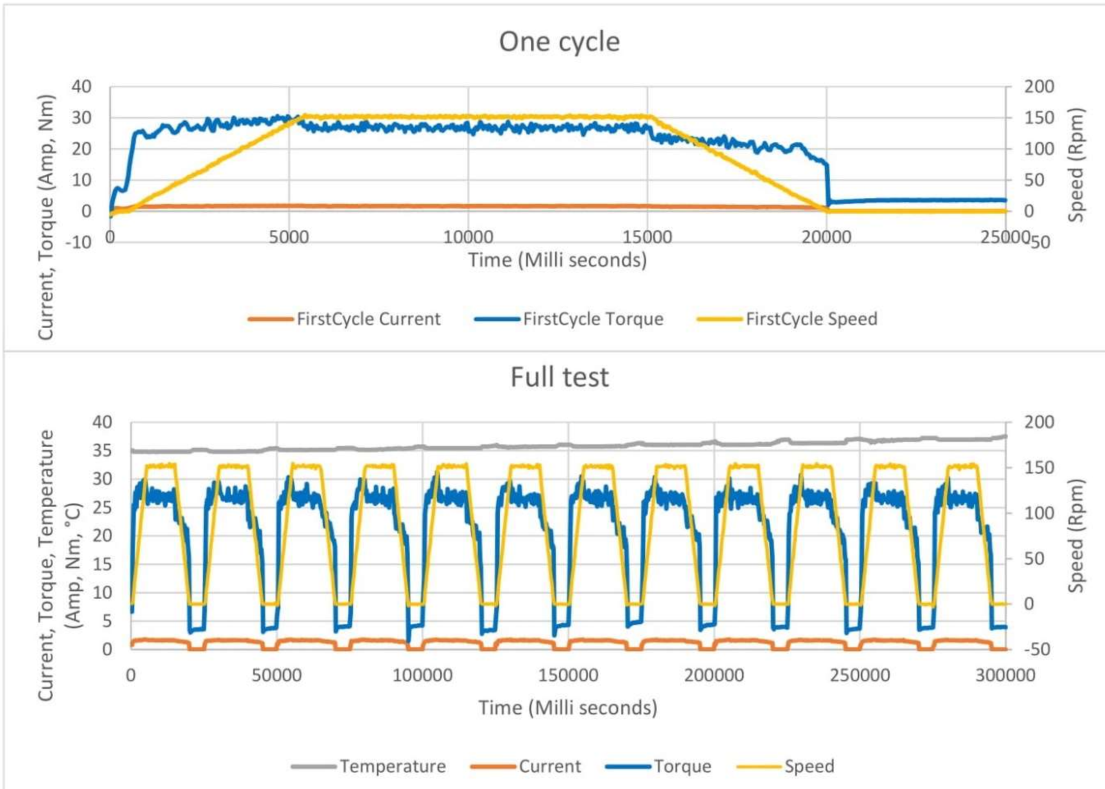
Figure 49: Example efficiency test

7.6.4. Endurance test

The endurance test report contains two graphs. The first graph represents one cycle and shows the current, torque and speed. The second graph shows the current, torque, speed and temperature over the total test time. As shown in Figure 50.

<u>Endurance Test</u>

VANDERLANDE



Test data	
Endurance Test	
Date	14-5-2019
Acceleration Time	5 Sec
Continous Time	10 Sec
Deceleration Time	5 Sec
Stop Time	5 Sec
Total test time	0 days
	0 hours
	5 minuts

Motor data	
Type	Nord
Power	550 Watt
Motor Gear	6.94
Voltage	380-400
Frequency	50 Hz
Speed	1420 Rpm

Made by: Hendrik Vandervelden

Figure 50: Example endurance test

8. CONCLUSIONS

A motor testbench can be programmed with only a SEW motor controller as PLC. To read the torque and input power additional slave units are necessary as slave module for the controller. For this testbench the Beckhoff EL3602 is used to measure the torque and the Beckhoff EL3403 is used to measure the electrical power.

The current testbench can test motors till 155Nm and 1450Rpm or till 390Nm and 450Rpm depending on which servomotor (fixed motor) is selected as brake. There are four tests that can be done. These are the efficiency test, torque-speed test, motor characteristics test and an endurance test. The maximum brake torque selected for the endurance test cannot be larger than the maximum allowed torque of the fixed motor, if it is then the test will continue but the torque in the graph on the test report will not be as asked. The nominal torque of the motor may be larger than torque of the fixed motor for this test. The maximum torque of the motor-on-test has to be lower (or equal) than the maximum torque of the fixed motor for the efficiency test. For the torque-speed test should the peak torque of the motor-on-test always be lower than the maximum torque of the fixed motor. The peak torque of a three-phase asynchronous motor is around three times the nominal torque. The maximum torque that is needed for the motor characteristics test is depended on the overload factor. The maximum overload factor should always be lower than the maximum fixed motor torque divided by the nominal torque of the motor-on-test. This should be taken in to account when Vanderlande redesigns the motor test bench.

The method to switch motors (fixed or on test) should be redesigned to allow for an easier way of changing the motors. Disconnecting and connecting the shafts, when another fixed motor is used, is time-consuming. Switching the motor-on-test is both time-consuming and a heavy since the motor with motor block has to be taken of the testbench.

The electrical cabinet could be redesigned and be equipped with more terminal blocks. Enough to have the power supply for the drive of the motor-on-test come from terminal blocks that are not used for the DOL test and still have three-phase voltage avoidable on terminal blocks. Also, the 24V and 12V should still be available on terminal blocks and should be used more from terminal blocks rather than from the power supply itself. When two 3-phase switches are installed (one between the drive output and terminal blocks and one between the grid and terminal blocks) a test could be set up while the total machine is not disconnected from the grid. When turning the switches, all terminal blocks necessary for the wiring of a test are without a potential difference. These changes allow for faster switching between a test with a drive and one with DOL.

Further developing of this testbench could also consider the regenerative power supply or a HMI. This power supply is interesting when the DC power for the fixed motors cannot be used for the DC power of the motor-on-test. Depending on the amount of endurance tests it might be beneficial to connect the DC power of the motor-on-test and the fixed motors. With the HMI it won’t be necessary to use a PC to start the test.

REFERENCES

[1] J. Kanters, Project Testopstelling koppelmeten motoren, ROC ter AA, Helmond: Ongepubliceerd eindwerk, 2015.

[2] SEW-EURODRIVE, “MOVI-C modular automation system,” [Online]. Available: https://master.sew-eurodrive.com/media/sew_eurodrive/pdf/produkte_1/movi_c_1/movi-c-en-24807346.pdf. [Accessed 27 5 2019].

[3] SEW-EURODRIVE, “MOVI-C controller power UHX85A and power eco UHX84A,” 05 2016. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/22781455.pdf. [Accessed 30 04 2019].

[4] IEEE, IEEE Standaard Test Procedure for Polyphase Induction Motors and Generators, New York: IEEE, 2004.

[5] B. T. Rade Buinac, Determination of the torque-speed characteristic of induction motor in electric machinery education [Paper], Zagreb: Zagreb university of applied sciences, 2013.

[6] D. Polka, Motor & Drives: A practical technology guide, ISA, 2003.

[7] NORD DRIVESYSTEMS, “Loadcurves 90SP/4 400VY 50Hz,” 2014.

[8] SEW-EURODRIVE, “Motor Characteristics DRN 90S 4 1,1kW 4Pol 50Hz,” 2015.

[9] F. d. Bruin, Interviewee, *Movi-C controller.* [Interview]. 10 10 2018.

[10] M. Vanstappen, Interviewee, *Movi-C controller.* [Interview]. 9 1 2019.

[11] 3S-Smart Software Solutions, “help.codesys,” [Online]. Available: https://help.codesys.com/. [Accessed 26 10 2018].

[12] H. Hobbs, Director, *Insane excel challenge - Over 9 hours to reach the bottom of Excel...* [Film]. 2017.

[13] 3S-Smart Software Solutions, “codesys store,” [Online]. Available: store.codesys.com. [Accessed 2018 11 28].

[14] plc2k, “plc2k,” 17 9 2015. [Online]. Available: http://plc2k.com/how-to-read-values-from-controllogix-l61-directly-in-excel. [Accessed 27 11 2018].

[15] J. Schop, “Tools for PLC programming,” [Online]. Available: http://www.plcdev.com/connecting_controllogix_excel. [Accessed 27 11 2018].

[16] F. Noble, Director, *Beckhoff PLC Control via TwinCAT visualisation.* [Film]. 2018.

[17] M.Schaber, “Scripting examples and receipes,” 5 5 2011. [Online]. Available: https://forum.codesys.com/viewtopic.php?t=1890. [Accessed 24 10 2018].

[18] GitHub, “IronPython documentation,” [Online]. Available: https://ironpython-test.readthedocs.io/en/latest/. [Accessed 26 10 2018].

[19] SEW-EURODRIVE, Manual MOVISUITE, 2017.

[20] SEW-EURODRIVE, “Manual movidrive modular,” 03 2017. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/23085711.pdf. [Accessed 5 11 2018].

[21] SEW EURODRIVE, “Operating instructions synchronous conpact servomotors,” 02 2012. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/11477016.pdf. [Accessed 20 3 2019].

[22] SEW-EURODRIVE, *Circuit diagram synchronous motor with terminal box cmp71-100, cmp112,* Bruchsal, 2016.

[23] SEW-EURODRIVE, “Motor Designs - CMP Servomotors,” 09 2009. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/17014417_G04.pdf. [Accessed 30 4 2019].

[24] Janitza, “Janitza,” [Online]. Available: Janitz.com. [Accessed 13 11 2018].

[25] SEW-EURODRIVE, “Manual MOVI-PLC I/O System C,” 11 2014. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/21276595.pdf. [Accessed 30 4 2019].

[26] I. Giangrandi, “Giangrandi,” 11 2014. [Online]. Available: www.griangrandi.ch. [Accessed 23 11 2018].

[27] Beckhoff, “Beckhoff,” [Online]. Available: beckhoff.com. [Accessed 15 11 2018].

[28] MEGADYNE, “Isoran Synchronous Timing belt,” 02 2018. [Online]. Available: https://resources.megadynegroup.com/resources/isoran-belts. [Accessed 19 02 2018].

[29] MEGADYNE, “MEGAPULLEY,” 12 2014. [Online]. Available: https://resources.megadynegroup.com/resources/megapulley-pulley-flanges-timing-bars-clamping-plates-taper-bushes. [Accessed 15 2 2019].

[30] 3S- Smart Software Solutions, “Dimoulas,” 20 08 2003. [Online]. Available: http://www.dimoulas.com.gr/PDF/WAGO%20PLC/CoDeSys_Visu_E.pdf. [Accessed 9 5 2019].

[31] SEW-EURODRIVE, “MOVIDRIVE modular Parameter Description,” 03 2017. [Online]. Available: https://download.sew-eurodrive.com/download/pdf/22730257.pdf. [Accessed 30 4 2019].

APPENDICES

- Appendix A: List of motors CD-ROM
- Appendix B: Eplan CD-ROM
- Appendix C: Codesys program CD-ROM
- Appendix D: Movisuite test1 CD-ROM
- Appendix E: Movisuite test2 CD-ROM
- Appendix F: Script CD-ROM
- Appendix G: Excel code CD-ROM