BACHELOR THESIS – ME 141502

DEVELOPMENT OF LOAD CHARACTERISTIC FOR THE MAIN ENGINE AND ITS PLC COMPATIBLE PREPARATION IN COOPERATION WITH THE WATER BRAKE AS THE GENERATOR

MUHAMMAD FAIZ RIFQI ARDYATAMA
NRP. 4213 101 041

Supervisors :
Prof. Dr.-Ing. Matthias Markert
Dipl.-Ing. Hartmut Schmidt

DOUBLE DEGREE PROGRAM OF
MARINE ENGINEERING DEPARTMENT
Faculty of Marine Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017

BACHELOR THESIS AND COLLOQUIUM – ME141502

DEVELOPMENT OF LOAD CHARACTERISTIC FOR THE MAIN ENGINE AND ITS PLC COMPATIBLE PREPARATION IN COOPERATION WITH THE WATER BRAKE AS THE GENERATOR

MUHAMMAD FAIZ RIFQI ARDYATAMA
NRP. 4213 101 041

Academic Supervisor 1
Prof. Dr.-Ing. Matthias Markert
NIP. -

Academic Supervisor 2
Dipl.-Ing. Hartmut Schmidt
NIP. -

DOUBLE DEGREE PROGRAM OF
MARINE ENGINEERING DEPARTMENT
FACULTY OF MARINE TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2017

SKRIPSI – ME141502

PENGEMBANGAN KARAKTERISTIK BEBAN UNTUK MESIN UTAMA DAN PERSIAPAN KOMPATIBEL PLC-NYA DALAM KERJASAMA DENGAN WATER BRAKE SEBAGAI GENERATOR

MUHAMMAD FAIZ RIFQI ARDYATAMA
NRP. 4213 101 041

Dosen Pembimbing 1
Prof. Dr.-Ing. Matthias Markert
NIP. –

Dosen Pembimbing 2
Dipl.-Ing. Hartmut Schmidt
NIP. -

PROGRAM GELAR GANDA
DEPARTEMEN TEKNIK SISTEM PERKAPALAN
FAKULTAS TEKNOLOGI KELAUTAN
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2017

APPROVAL FORM

Development of Load Characteristic for The Main Engine and Its PLC Compatible Preparation in Cooperation with The Water Brake as The Generator

BACHELOR THESIS

Submitted to Comply One of The Requirements to Obtain a Bachelor Engineering Degree
In
Double Degree of Marine Engineering Program
Department of Marine Engineering
Faculty of Marine Technology
Institut Teknologi Sepuluh Nopember
and
Department of Maritime Studies
Faculty of Engineering
Hochschule Wismar, University of Applied Sciences Technology, Business and Design

Submitted by:

Muhammad Faiz Rifqi Ardyatama

NRP. 4213101041

Approved by Supervisors:

Prof. Dr.-Ing. Matthias Markert ( )

Dipl.-Ing. Hartmut Schmidt ( )

ROSTOCK
JULY, 2017

"This page intentionally left blank"

APPROVAL FORM

Development of Load Characteristic for The Main Engine and Its PLC Compatible Preparation in Cooperation with The Water Brake as The Generator

BACHELOR THESIS

Submitted to Comply One of The Requirements to Obtain a Bachelor Engineering Degree
in
Double Degree of Marine Engineering Program
Department of Marine Engineering
Faculty of Marine Technology
Institut Teknologi Sepuluh Nopember
and
Department of Maritime Studies
Faculty of Engineering
Hochschule Wismar, University of Applied Sciences Technology, Business and Design

Submitted by:

Muhammad Faiz Rifqi Ardyatama

NRP. 4213101041

Approved by Head of Marine Engineering Department:

Dr. Eng. Muhammad Badrus Zaman, S.T., M.T.
NIP. 1977 0802 2008 01 1007

DEPARTEMEN TEKNIK SISTEM PERKAPALAN

SURABAYA
JULY, 2017

"This page intentionally left blank"

APPROVAL FORM

Development of Load Characteristic for The Main Engine and Its PLC Compatible Preparation in Cooperation with The Water Brake as The Generator

BACHELOR THESIS

Submitted to Comply One of The Requirements to Obtain a Bachelor Engineering Degree
in
Double Degree of Marine Engineering Program
Department of Marine Engineering
Faculty of Marine Technology
Institut Teknologi Sepuluh Nopember
and
Department of Maritime Studies
Faculty of Engineering
Hochschule Wismar, University of Applied Sciences Technology, Business and Design

Submitted by:

Muhammad Faiz Rifqi Ardyatama

NRP. 4213101041

Approved by Representative of Hochschule Wismar in Indonesia:

Dr.-Ing. Wolfgang Busse
Coordinator of Hochschule Wismar, Germany, in Indonesia

SURABAYA
JULY, 2017

"This page intentionally left blank"

Development of Load Characteristic for The Main Engine and Its PLC Compatible Preparation in Cooperation with The Water Brake as The Generator

Name	**:**	**Muhammad Faiz Rifqi Ardyatama**
NRP	**:**	**4213101041**
Department	**:**	**Double Degree of Marine Engineering**
Supervisors	**:**	**Prof. Dr.-Ing. Matthias Markert**
		Dipl.-Ing. Hartmut Schmidt

ABSTRACT

Water brake can be used in any application where a load brake is required on a rotational load. In the engine industry, dynamometer is used in conjunction with a power source and instrumentation to determine engine HP, Torque, and Efficiency ratings with a high degree of accuracy. Water brake are proven durable designs which use water flowing through the absorber to create a load on the engine. Only the amount of water necessary to provide the load is required.

Precise load control of the dynamometer is as simple as increasing or decreasing water volume flowing through the dynamometer absorption body. The controls ensure engine load remains stable throughout the duration of the test cycle. Torque, horsepower, RPM, and water temperature are displayed on highly accurate digital instrumentation.

Computers can perform both sequential control and feedback control, and typically a single computer will do both in an industrial application. Programmable logic controllers are a type of special purpose microprocessor that replaced many hardware components such as timers and drum sequencers used in relay logic type systems. They can also analyze data and create real time graphical displays for operators and run reports for operators, engineers and management.

Industrial control systems are usually used in all over in control system. A distributed control system refers to a control system in which the controllers are spread throughout the system and connect by networks. Smaller automation applications can be implemented with programmable logic controllers.

This thesis basically is about designing a programmable logic controller compatible for dynamometer to control speed, power and torque of main engine. To test engine performance in the laboratory, the engine is coupled to a dynamometer. In this thesis dynamometer are used to measure speed, power and torque. One of the most essential role of the control engineer is tuning of controller. Hence the performance of the calculated controller parameters depends on the correctness of the identified process model developed from engine behavior.

Keyword: Load Characteristics, Main Engine, PLC, Water Brake, Generator

Pengembangan Karakteristik Beban untuk Mesin Utama dan Persiapan Kompatibel PLC-nya dalam Kerjasama dengan Water Brake sebagai Generator

Nama	**:**	**Muhammad Faiz Rifqi Ardyatama**
NRP	**:**	**4213101041**
Jurusan	**:**	**Teknik Sistem Perkapalan Gelar Ganda**
Pembimbing	**:**	**Prof. Dr.-Ing. Matthias Markert**
		Dipl.-Ing. Hartmut Schmidt

ABSTRAK

Rem air dapat digunakan dalam aplikasi dimana rem beban diperlukan pada beban rotasi. Di industri mesin, dinamometer digunakan bersamaan dengan sumber tenaga dan instrumentasi untuk menentukan nilai dari mesin HP, Torsi, dan Efisiensi dengan tingkat akurasi yang tinggi. Rem air terbukti tahan lama didesain menggunakan air yang mengalir melalui penyerap untuk membuat beban pada mesin. Hanya jumlah dari air yang dibutuhkan untuk menyediakan muatan.

Pengendalian beban yang tepat dari dinamometer sesederhana meningkatkan atau menurunkan volume air yang mengalir melalui badan penyerapan dynamometer. Kontrol memastikan putaran mesin tetap stabil sepanjang durasi siklus uji. Torsi, tenaga kuda, RPM, dan suhu air ditampilkan pada instrumentasi digital yang sangat akurat.

Komputer dapat melakukan kontrol sekuensial dan kontrol umpan balik, dan biasanya komputer tunggal akan melakukan keduanya dalam aplikasi industri. Programmable logic controllers adalah jenis mikroprosesor tujuan khusus yang menggantikan banyak komponen perangkat keras seperti timer dan sequencer drum yang digunakan pada sistem tipe logika relay. Mereka juga dapat menganalisis data dan membuat tampilan grafis real time untuk operator dan menjalankan laporan untuk operator, insinyur dan manajemen.

Sistem kontrol industri biasanya digunakan di seluruh sistem kontrol. Sistem kontrol terdistribusi mengacu pada sistem kontrol dimana pengendali disebarkan ke seluruh sistem dan dihubungkan oleh jaringan. Aplikasi otomasi yang lebih kecil dapat diimplementasikan dengan pengendali logika programmable.

Tesis ini pada dasarnya adalah tentang merancang sebuah programmable logic controller yang kompatibel untuk dinamometer untuk mengendalikan kecepatan, tenaga dan torsi mesin utama. Untuk menguji performa mesin di laboratorium, mesin ini digabungkan ke dynamometer. Dalam tesis ini dinamometer digunakan untuk mengukur kecepatan, tenaga dan torsi. Salah satu peran paling penting dari engineer kontrol adalah tuning controller. Oleh karena itu, kinerja parameter pengendali yang dihitung bergantung pada ketepatan model proses yang diidentifikasi yang dikembangkan dari perilaku mesin.

Kata Kunci: Karakteristik Beban, Mesin Utama, PLC, Water Brake, Generator

PREFACE

Alhamdulillah, all praise and gratitude author said to the presence of Allah SWT, that has given guidance, pleasure, and opportunities so author can finish this bachelor thesis report. In this process of making this bachelor thesis report, the author would like to say thank those who have helped to complete this final project report, including:

1. My father Abdul Rasyid Chalik and My mother Dyah Yekti Nugraheni. My brother Ivan Dwi Hascaryo Ardynugroho and My sister Ardyatifa Siti Nurakhim who always gives me support from afar every time.
2. All of my family member like grandfather, grandmother, uncle, aunt, nephew who always support my study.
3. Mr. Dr. Eng. M. Badrus Zaman, S.T, M.T as Head Department of Marine Engineering Faculty of Marine Technology ITS.
4. Mr. Sutopo Purwono Fitri, S.T., M.Eng., Ph.D., as the supervisor of my collage life from semester 1 until semester 8 at Department of Marine Engineering Faculty of Marine Technology ITS.
5. Mr. Prof. Dr.-Ing. Matthias Markert and Mr. Dipl.-Ing. Hartmut Schmidt as my supervisor of my bachelor thesis at Department of Maritime Studies, Hochschule Wismar, Germany.
6. Mr. Jens Borchhardt as electrical laboratory engineer at Department of Maritime Studies, Hochschule Wismar, Germany.
7. Mutiara Nanda Fitriana who always give me her time for supporting me to finish my bachelor thesis report and always give me spirit every time I fail.
8. BARAKUDA '13 family who always supporting each other from 2013 until now.
9. Double Degree Marine Engineering Program '13 who always together. Making a lot of memory which can't be forgotten for 4 years and always still counting until now.
10. My senior high school friends who not forget to always support me after we have different universities.
11. My Marine Solar Boat Team friends who not forget to always support me although now it has its own activity.
12. All the people that can't mention one by one and has helped me to finish bachelor thesis report completely.

The author realizes that this Bachelor Thesis still far from perfect. Therfore, every constructive and supportive suggestion and idea to improve this Bachelor Thesis are highly expected by the author so that the better version of this Bachelor Thesis can be reach in the future.

Finally, last but not least may Allah SWT bestow his grace, mercy, and blessing to all of us and hopefully this Bachelor Thesis can be advantageous to corresponding parties who may concern.

Surabaya, July 2017

Author

LIST OF CONTENTS

LIST OF FIGURES

"This page intentionally left blank"

LIST OF DIAGRAMS

"This page intentionally left blank"

LIST OF TABLES

"This page intentionally left blank"

LIST OF ABBREVIATIONS AND SYMBOLS

Abbreviations	Description
HP	Horse Power
RPM	Revolutions Per Minute
DC	Direct Current
AC	Alternating Current
PLC	Programmable logic controllers
LNG	Liquefied Natural Gas
MCR	Maximum Continuous Rating
DCS	Distributed Control System
USA	United States of America
ROM	Read Only Memory
RAM	Random Access Memory
CPU	Central Processing Unit
PS	Power Supply
I/O	Input / Output
V	Volt
A	Ampere
RS	Reset & Set
PID	Proportional Integral Derivative
FBD	Function Block Diagram
LD	Ladder Diagram
SFC	Sequential Function Control
IL	Instruction List
ST	Structured Text
STL	Statement List
LAD	Ladder Logic
WinCC	Windows Control Center
OB	Organization Block
FC	Function
FB	Function Block
DB	Data Block
MMC	Micro Memory Card
MPI	Message Passing Interface
kB	Kilo Byte
MCC	Mobile Country Code
LED	Light Emitting Diode
PtP	Precision Time Protocol
DI	Digital Input

DO	Digital Output
AI	Analog Input
AO	Analog Output
P	Power
T	Torque
S	Speed
kW	Kilo Watt
Nm	Newton Meter
mm	Milimeter
g/kWh	Specific Fuel Consumption
m^3/h	Mass Flow Rate
Ton	Tonnage
IMO	International Maritime Organization

CHAPTER 1
INTRODUCTION

1.1. Background

Dynamometer (water brake) can be used in any application where a load brake is required on a rotational load. In the engine industry, dynamometer used in conjunction with a power source and instrumentation to determine engine HP, Torque, and Efficiency ratings with a high degree of accuracy.

Other characteristics of an operating engine can only be determined with a dynamometer, such as: heat rise, bearing performance, winding design rating, insulation integrity, harmonic distortion, rotor bar lo oneness, DC brush contact, variable frequency drive influences, and engine slip characteristics. A properly operating engine should last longer, run better, and cost less to maintain.

Dynamometer (water brake) utilize water flow proportional to the applied load to create resistance to the engine. A controlled flow of water through the inlet manifold is directed at the center of the rotor in each absorption section. This water is then expelled towards the outside of the dynamometer body by centrifugal force. As it is directed outward, the water is accelerated into pockets on the stationary stator plates where it is decelerated. This continuous acceleration/deceleration of the water creates the applied load to the engine. Through this transfer of energy, the water is heated and discharged.

Dynamometer (water brake) are proven durable designs which use water flowing through the absorber to create a load on the engine. Only the amount of water necessary to provide the load is required. Sufficient internal clearance within the absorber body eliminates the need for cooling water and thermal overload protection, unlike typical industry offerings. If the water overheats under test and vaporizes, then the dynamometer simply stops absorbing the load.

Precise load control of the dynamometer is as simple as increasing or decreasing water volume flowing through the dynamometer absorption body. The controls ensure engine load remains stable throughout the duration of the test cycle. Torque, horsepower, RPM, and water temperature are displayed on highly accurate digital instrumentation [1].

1.2. Statement of Problems

The statement of problem of this thesis are;

1. How the influence of water brake to apply load to the main engine?
2. How the influence of generator to apply load to the main engine?
3. How to prepare compatible PLC application to make load characteristics for the main engine?

1.3. Research Limitations

The limitations of this thesis are;

1. This thesis is focusing to develop of load characteristic for the main engine.
2. To develop of load characteristic for the main engine use water brake and generator.
3. The input or output data use PLC application.
4. The study conducted is only based on the data obtained from in the Laboratory Building of the Maritime Department of Hochschule Wismar.

1.4. Research Objectives

The objectives of this thesis are;

1. To develop load characteristics for various load states and load processes of water brake load aggregates.
2. To develop load characteristics for various load states and load processes of generator load aggregates.
3. To prepared compatible PLC application to make load characteristics for the main engine.

1.5. Research Benefits

The benefit of this thesis are;

1. Understand how to develop and influence of load characteristics for the main engine use water brake and generator.
2. Understand how to be prepared compatible PLC application to make load characteristics for the main engine.

CHAPTER 2
LITERATURE STUDY

2.1. Marine Propulsion

Marine propulsion is the mechanism or system used to generate thrust to move a ship or boat across water. Most modern ships are propelled by mechanical systems consisting of an electric motor or engine turning a propeller, or less frequently, in pump jets, an impeller. Marine engineering is the discipline concerned with the engineering design process of marine propulsion systems.

Marine steam engines were the first mechanical engines used in marine propulsion, however they have mostly been replaced by two stroke or four stroke diesel engines, outboard motors, and gas turbine engines on faster ships. Nuclear reactors producing steam are used to propel warships and icebreakers. Nuclear reactors to power commercial vessels has not been adopted by the marine industry. Electric motors using electric battery storage have been used for propulsion on submarines and electric boats and have been proposed for energy-efficient propulsion. Development in liquefied natural gas (LNG) fueled engines are gaining recognition for their low emissions and cost advantages. Stirling engines, which are more efficient, quieter, smoother running producing less harmful emissions than diesel engines, propel a number of small submarines. The Stirling engine has yet to be upgrade for larger surface ships.

2.2. Diesel Engine

The majority of large commercial ships, from freighters to oil tankers, use two stroke *(see figure 2.1),* slow speed diesel engines connected directly to their propeller. The large two stroke diesel engines are efficient for constant speed steaming and running off of heavy fuel oil. They run at speeds up to 100 or 150 revolutions per minute and compared to other types of propulsion plants, provide commercial mariners with a fairly simple propulsion plant to maintain. Compared to other fuels, this is very cheap but also known for lots of heavy air pollutants including sulfur oxides and nitrogen oxides [2].

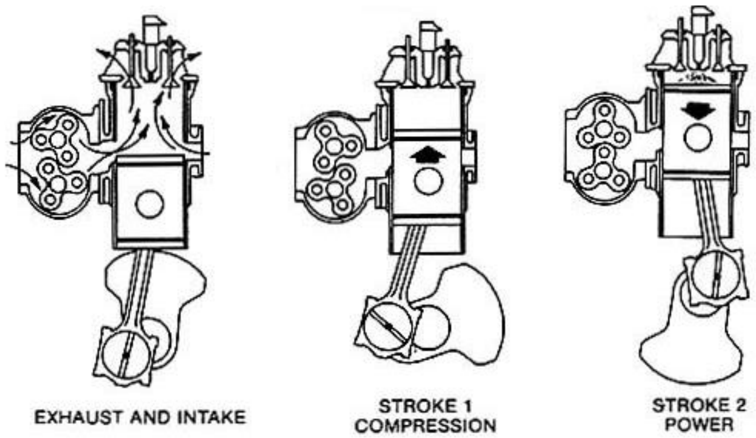


Figure 2.1: Two Stroke Diesel Engine.
(source:http://www.dieselduck.info/machine/01%20prime%20movers/diesel_engine/diesel_engine.01.htm)

Four stroke diesel engines are designed to operate at multiple speeds and are usually connected to the propeller through a reduction gear *(see figure 2.2)*. This piece of machinery takes the high-speed revolutions that four-stroke diesel engines are known for (can be upwards of several thousand revolutions per minute) and lowers it to a usable speed for the propeller (around 70-130 revolutions per minute). These engines are more common on Naval and Coast Guard vessels or work boats that operate that operate over a range of speeds. Many Naval and Coast Guard vessels also utilize gas turbines, the same kind that power airplanes, as they can be more efficient and more powerful than diesel engines and give those vessels the ability to travel 30 or 40 knots, or even faster.

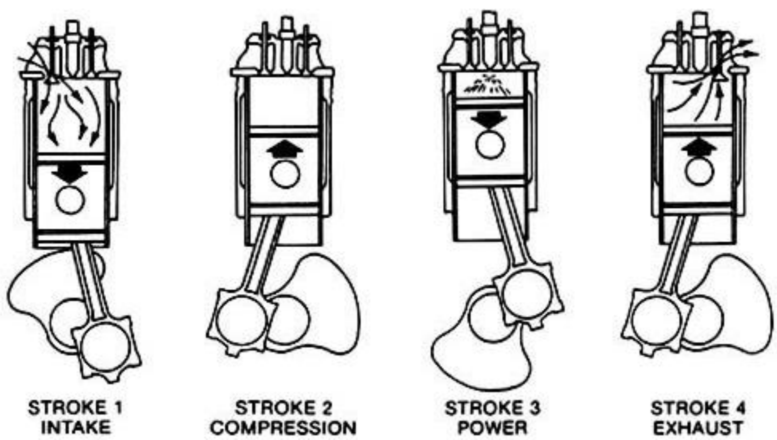


Figure 2.2: Four Stroke Diesel Engine.
(source:http://www.dieselduck.info/machine/01%20prime%20movers/diesel_engine/diesel_engine.02.htm)

2.3. Load Characteristic

In the below plot, the propeller demand curve with generator is shown to pass through the Maximum Continuous Rating (MCR) of the engine. The coordinates on the power rpm chart of MCR represent the peak value of (continuous) engine power delivery and the engine speed limit *(see figure 2.3).*

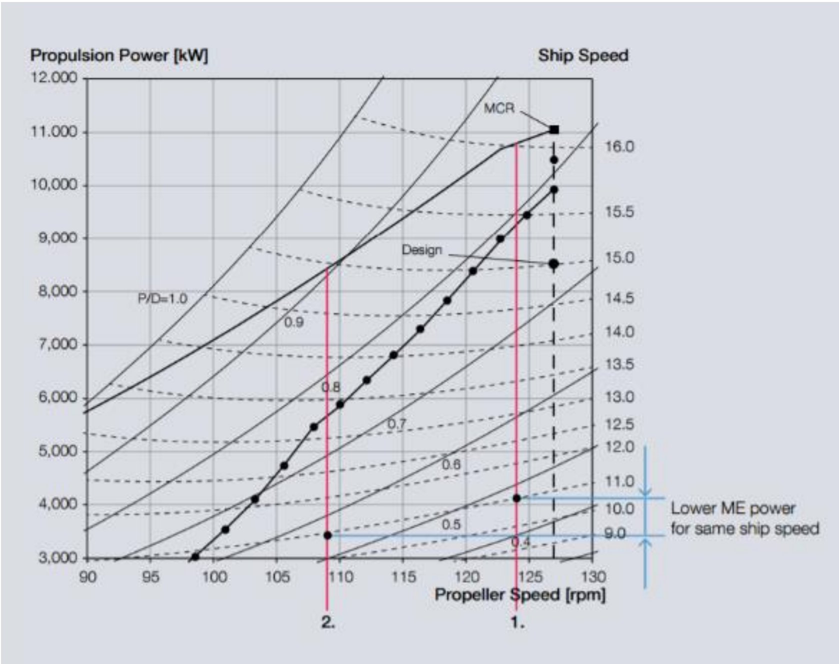


Figure 2.3: Chart Used for Engine Propeller Matching.
(source: Shaft Generator for Low Speed Main Engine. MAN Diesel & Turbo)

Propeller power absorption characteristic can be modified during service due to a variety of factors. Generally, increased severity of any of these factors requires a power increase in order to drive the ship at the same speed. In turn, this has an effect of moving the propeller power demand curve leftwards towards curve. Therefore, allowances need to be made for the propulsion plant to be able to develop full power under less profitable, located rightwards of the nominal (ideal) propeller demand curve, is selected for engine–propeller matching. This concept of difference in performance introduces the term "sea margin", in order to ensure that the ship propulsion plant has sufficient power available in service.

Another important aspect of marine engine operation is that engine torque delivery is proportional to index position for constant engine speed (rpm) and provided that the engine is running with "excess air", i.e. adequate air supply for perfect combustion. Note that this comes is not contradiction with the linear steady-state power dependence upon index, as the rpm value is different from one steady-state operating point to another [3].

Electric power with the same frequency generated from a shaft generator is possible within two propeller speed alternatives. Improved propeller efficiency for part load operation is possible. Slightly improved engine thermal efficiency is achieved because of the lower engine speed.

2.4. Generator

An electric generator *(see figure 2.4)* is a device that produces electrical energy from mechanical energy source, usually by means of electromagnetic induction. This process is known as electricity generation. Although generators and motors have many similarities, but the motor is a device that converts electrical energy into mechanical energy. Generator push electric charges to move through an external electrical circuit, but the generator does not generate electricity which is already in the cable windings. They are two types of AC generator, AC synchronous generator and AC asynchronous generator.

Figure 2.4: Generator on Ship.
(source: http://lokerpelaut.com/penyebab-turun-nya-kinerja-aux-engine-yang-perlu-di-ketahui.html*)*

AC synchronous generators lock in with fundamental line frequency and rotate at a synchronous speed related to the number of poles similar to that of AC synchronous motors. AC synchronous generator stator windings are similar to a three-phase synchronous motor stator winding. AC synchronous generator rotor fields may be either salient or non-salient pole. These pole windings are wound around field poles.

Asynchronous generator is a type of alternating current (AC) electrical generator that uses the principles of induction motors to produce power. Asynchronous generator operated by mechanically turning their rotors faster than synchronous speed. A regular AC asynchronous motor usually can be used as a generator, without any internal modifications.

Asynchronous generator is useful in applications such as mini hydro power plants, wind turbines, or in reducing high-pressure gas streams to lower pressure, because they can recover energy with relatively simple controls. An induction generator usually draws its excitation power from an electrical grid; sometimes, however, they are self-excited by using phase-correcting capacitors. Because of this, induction generators cannot usually "black start" a de-energized distribution system [4].

2.4.1. Design

In an AC generator, current flows through fixed coils. There is a magnet that moves, and both ends of the magnet cause the flow of current to remain in an opposite direction.

AC and DC generators have a commutator (slip rings) and an armature coil *(see figure 2.5)*. The rings are constructed of metal. The connection type between the outside circuit and the armature affects the type of current produced. Hence, these generators can produce varied current depending on the connection of the armatures to the external circuits.

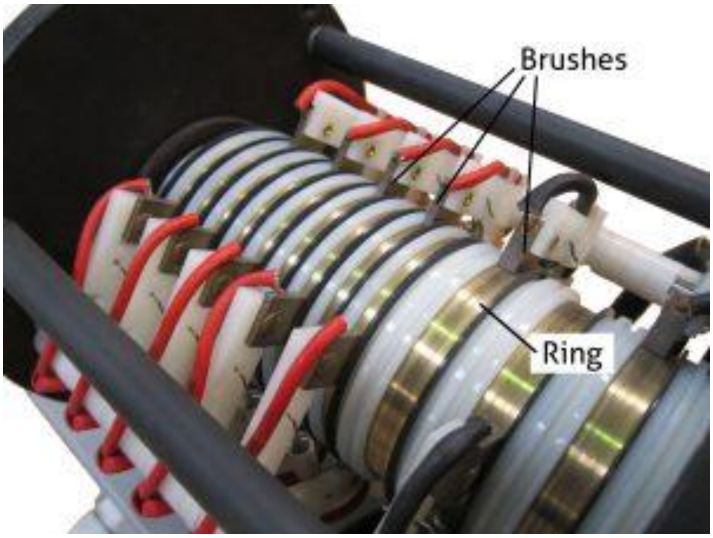


Figure 2.5: Inside Generator.
(source: https://powerbyproxi.com/slip-ring*)*

There are two metal rings in AC generators. The armature coil and commutators rotate in a simultaneous manner. This is made possible since there is a separate commutator in which either end of the armature is connected. These commutators include a carbon. However, AC generators contain a brush that remains steady and unmoving. The end of the armature and the outside circuit are both connected to one brush. The brushes receive the current that flows through the slip rings [5].

2.4.2. Components

1. Field

The field in an AC generator consists of coils of conductors within the generator that receive a voltage from a source (called excitation) and produce a magnetic flux. The magnetic flux in the field cuts the armature to produce a voltage. This voltage is ultimately the output voltage of the AC generator.

2. Prime Mover

The prime mover is the component that is used to drive the AC generator. The prime mover maybe any type of rotating machine, such as a diesel engine, a steam turbine, or a motor.

3. Rotor

The rotor is a mechanical device that rotates / vane, for example in electric motor vehicle, generator, alternator or pump.

4. Stator

The stator is the stationary part of a rotary system, found in electric generators, electric motors, sirens. The main use of a stator is to keep the field aligned.

5. Slip rings

A slip ring is an electromechanical device that allows the transmission of power and electrical signals from a stationary to a rotating structure. A slip ring can be used in any electromechanical system that requires rotation while transmitting power or signals. It can improve mechanical performance, simplify system operation and eliminate damage-prone wires dangling from movable joints.

2.5. Water Brake

A dynamometer *(see figure 2.6)* is a load device which is generally used for measuring the power output of an engine. Several kinds of dynamometers are common, some of them being referred to as “breaks” or “break dynamometers”: dry friction break dynamometers, hydraulic or water break dynamometers and eddy current dynamometers.

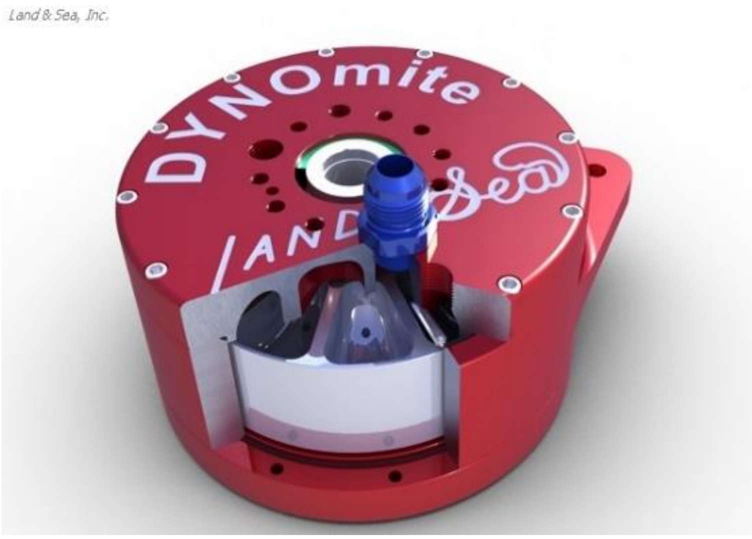


Figure 2.6: Water Brake Dynamometer.
(source: http://www.dynomitedynamometer.com/absorber/dynamometer_water-brake_absorber.htm*)*

Dry friction dynamometers are the oldest kind, and consist of some sort of mechanical breaking device, often a belt or frictional "shoe" which rubs a rotating hub or shaft. The hub or shaft is spun by the engine. Increasing tension in the belt, or force of the shoe against the hub increases the load on the engine.

Hydraulic dynamometers are basically hydraulic pumps where the impeller is spun by the engine. Load on the engine is varied by opening or closing a valve, which changes back pressure on the hydraulic pump.

Eddy current dynamometers are electromagnetic load devices. The engine being tested spins a disk in the dynamometer. Electrical current passes through coils surrounding the disk, and induce a magnetic resistance to the motion of the disk. Varying the current varies the load on the engine [6].

The dynamometer applies a resistance to the rotation of the engine. If the dynamometer is connected to the engine's output shaft it is referred to as an Engine Dynamometer. The force exerted on the dynamometer housing is resisted by a strain measuring device *(see figure 2.7)*.

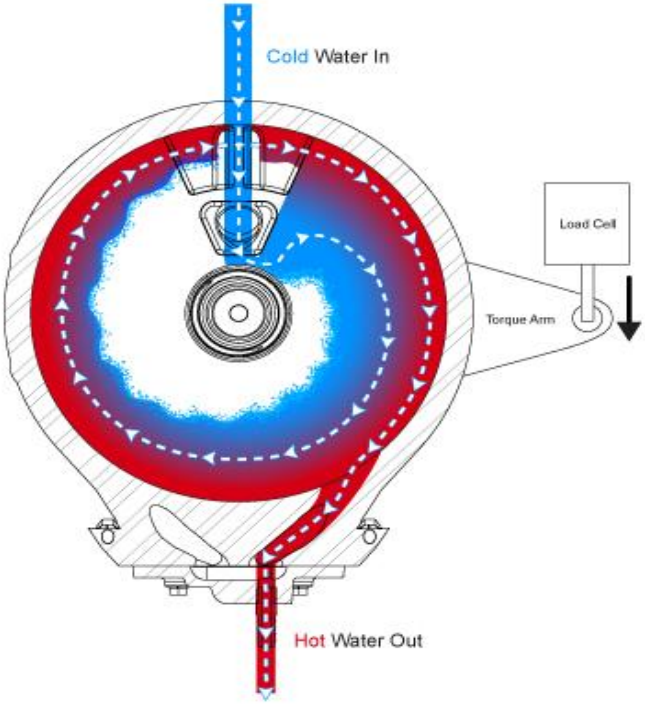


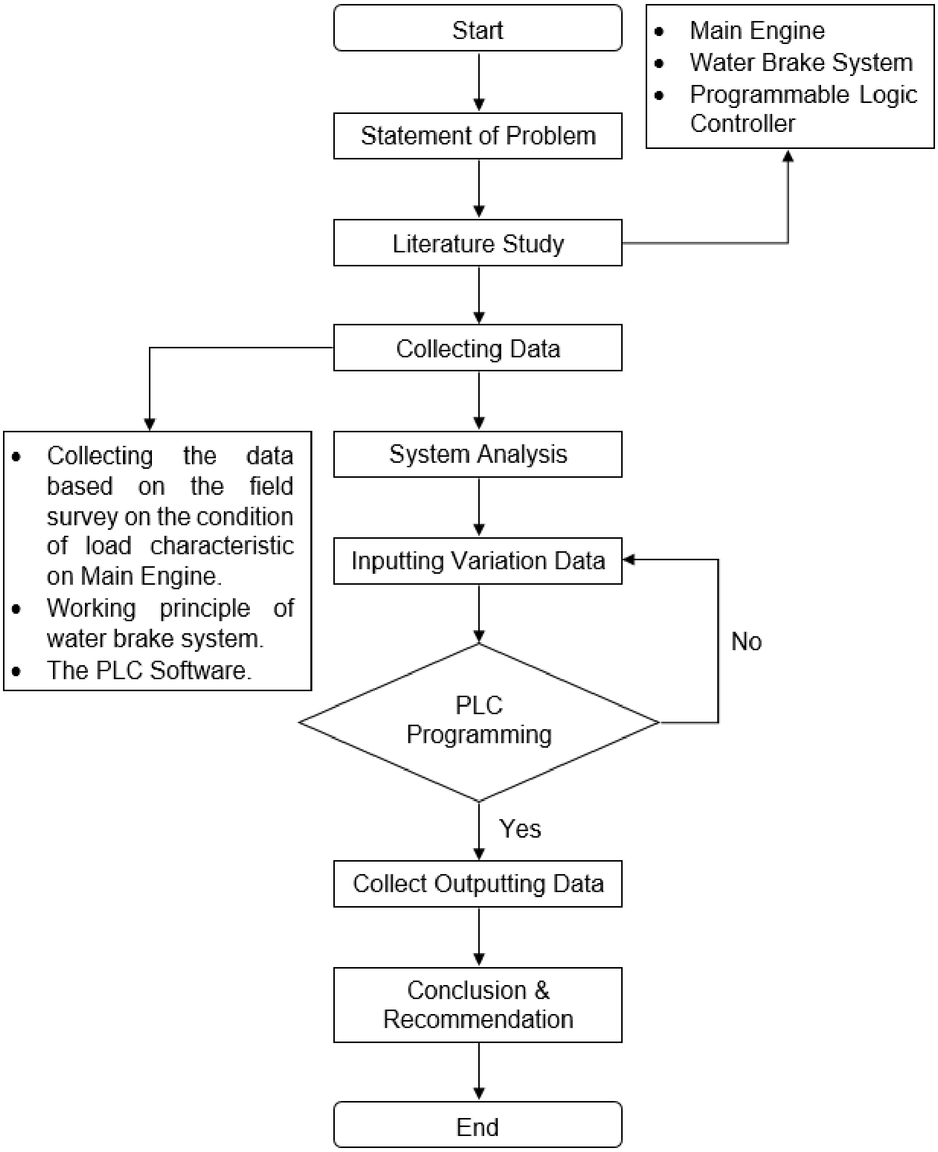
Figure 2.7: Dynamometer Operation.
(source: http://www.stuskadyno.com/tech-notes*)*

Several components are typically packaged together in a dynamometer: the shaft with bearings, the resistance surface, the resistance mechanism in a "free" rotating housing, a strain gage, and a speed sensor. Usually some method of cooling is also required, and this may require either a heat exchanger or air or water circulation. The entire assembly is typically mounted on a stout frame, which is mechanically linked to the frame of the engine being tested [7].

Water brake dynamometers are similar to a centrifugal pump or a torque converter. A rotor is turning inside a stator housing. Water is introduced creating friction and heat with torque being applied to the stator. The stator would spin with the rotor except the stator is anchored with a torque arm and a load cell. The torque that is being loaded gets transmitted through the torque arm and into the load cell that measures the force. The amount of heat generated is directly proportional to the power being absorbed. The water brake is convenient because the water that creates the load also carries away the heat as it is discharged from the brake.

CHAPTER 3
METHODOLOGY

3.1. Methodology Flow Chart



3.2. Definition of Methodology Flow Chart

3.2.1. Statement of Problems

This stage is an early stage to construct the thesis. In this stage, questions and problems are being prepared specifically in order to determine the specific objectives of this thesis. The content of the thesis is to overcome the statement of the problems mentioned earlier and it will be done by collect some information about the problems. Therefore, the purpose of this thesis can be understood in this stage.

3.2.2. Literature Study

Right after the problems is raised, a literature study is performed. In this stage, literature will be used to connect the problems with existing theories and facts from various sources. Since this thesis is an implementation of many aspects discipline, various literature topics is required to be constructed into one project. The study of literature is done by reading papers, journals, thesis, media and literature books that relates and able to support this thesis.

3.2.3. Collecting Data

After literature study which support the thesis has been done, collecting data is being performed. Data collection is done by gather information to develop the analysis.

3.2.4. System Analysis

After the required data has been collected, the next step is to analyzing system used. At this stage is certain that the system and components is a compatible or not. Because if the system and component not compatible, so the system must be repaired.

3.2.5. Inputting Variation Data

After the system is declared safe and functioning properly then able to use the system. From various sources has been obtained then made a wide variety of data to be entered into the software PLC as the beginning of the experiment.

3.2.6. PLC Processing

At this stage, the time from a wide variety of data inputted before, then the software will process the data automatically. Some variation of these data will generate a wide variety of different results. We can observe the difference

3.2.7. Collect Outputting Data

This is the last stage of the experiment. From a variety of different results, we can collect the results from such data. We collect the data that can be used and discard the data that can't be used. If the data can't be used then testing can be done again.

3.2.8. Conclusion & Recommendation

After obtaining all the necessary data, then we can process and analyze the data. Than processed products and analysis of these data we can conclude from the results of this experiment. Conclusion of this experiment can produce a suggestion that can be useful to improve the situation if any deficiencies.

"This page intentionally left blank"

CHAPTER 4
DISCUSSION & ANALYSIS

4.1. Overview

Computers can perform both sequential control and feedback control, and typically a single computer will do both in an industrial application. Programmable logic controllers (PLC) are a type of special purpose microprocessor that replaced many hardware components such as timers and drum sequencers used in relay logic type systems. General purpose process control computers have increasingly replaced itself by controllers, with a single computer able to perform the operations of hundreds of controllers. Process control computers can process data from a network of PLC, instruments and controllers in order to implement typical control of many individual. They can also analyze data and create real time graphical displays for operators and run reports for operators, engineers and management.

Industrial control systems are usually no all over used in large applications such as paper machines, chemical processes or power plants. A distributed control system (DCS) refers to a control system in which the controllers are spread throughout the system and connect by networks. Smaller automation applications can be implemented with programmable logic controllers (PLC). They are also based on microprocessors and software. They are considerably cheaper than the lower level automation systems they replace. PLC systems are widely used in bulk manufacture or if the production is low-volume, flexible and needs fast changes.

PLCs first emerged in USA in the late 1960s to meet the needs of automotive industry. Before PLCs the production was controlled by relay systems with hard-wire control panels that needed time-consuming rewiring every time a change was needed. PLCs faced great success because of the benefits of fast and flexible reprogramming by software revision. The evolution of microprocessors in the 70s and 80s enabled PLCs to become more versatile and has increased their functionality. Today the largest PLC and control systems have very much in common [9].

IEC 61131-3 is the third part of the open international standard IEC 61131 for PLC, and was first published in December 1993 by the IEC. The current edition was published in February 2013. Part 3 of IEC 61131 deals with basic software architecture and programming languages of the control program within PLC.

4.2. Programmable Logic Controller

4.2.1. Components

1. Central Processing Unit (CPU)

Microprocessor based, may allow arithmetic operations, logic operators, block memory moves, computer interface, local area network, functions, etc. CPU makes a great number of check-ups of the PLC controller itself so eventual errors would be discovered early.

2. Memory

System (ROM) to give permanent storage for the operating system and the fixed data used by the CPU. RAM for data. This is where information is stored on the status of input and output devices and the values of timers and counters and other internal devices. EPROM for ROM‟s that can be programmed and then the program made permanent.

3. I/O Sections

Inputs connect field devices, such as switches and sensors. Outputs control connect other devices, such as motors, pumps, solenoid valves, and lights.

4. Power Supply

Most PLC controllers work either at 24 VDC or 230 VAC. Some PLC controllers have electrical supply as a separate module, while small and medium series already contain the supply module.

5. PLC Communication

Serial communication is used for transmitting data over long distances. Might be used for the connection between a computer and a PLC.

6. PLC Programming

There are six programming language available for any PLC.

4.2.2. Structure

Usually a PLC consists of different modules that are not necessarily physically next to each other. All PLCs have the elements *(see figure 4.1)*.

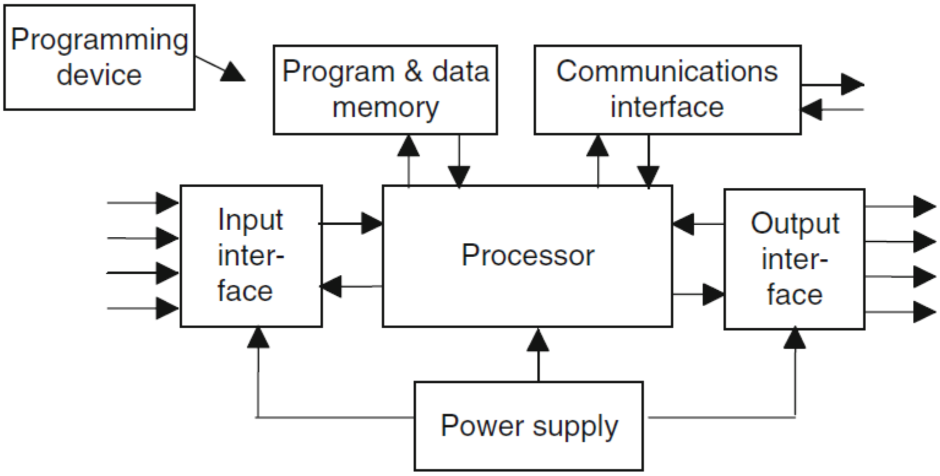


Figure 4.1: PLC Elements.
(source: https://youyunxia.wordpress.com/plc*)*

1. Input modules

All input signals from switches, limit switches, buttons and so on are connected to the input module. The signals can be binary (ON/OFF) or analogue.

2. Output modules

Output module is connected to valves, indicator lights, contactors and so on. Like inputs the outputs can be binary or analogue.

3. CPU (Central Processing Unit) and program memory

CPU consists of a microprocessor and an operating system. Program memory contains the user made application program that instructs the PLC to perform in a desired manner. CPU reads the input modules, executes the application program in program memory and then updates the output module.

CPU can be sweeping or real time. In a sweeping CPU, there are three action phases: reading the inputs, saving them in the auxiliary memory space, running the program and writing the outputs. Cycle time can be fixed or changing. During one sweeping cycle, no changes can be applied. Hence a sweeping CPU is often too slow [9].

4.2.3. Programming Languages

Programming of the PLC is done with a separate programming device, in this case a PC. The program is then transferred to the PLC program memory that executes the program.

Programming of the PLC can be done in different languages such as statement list or logic function block diagram. Basic logic operations such as (AND, OR, NOR, NAND), timers, counters, RS flip-flops and special operations (PID-control, for example) can be used in all languages *(see figure 4.2)*.

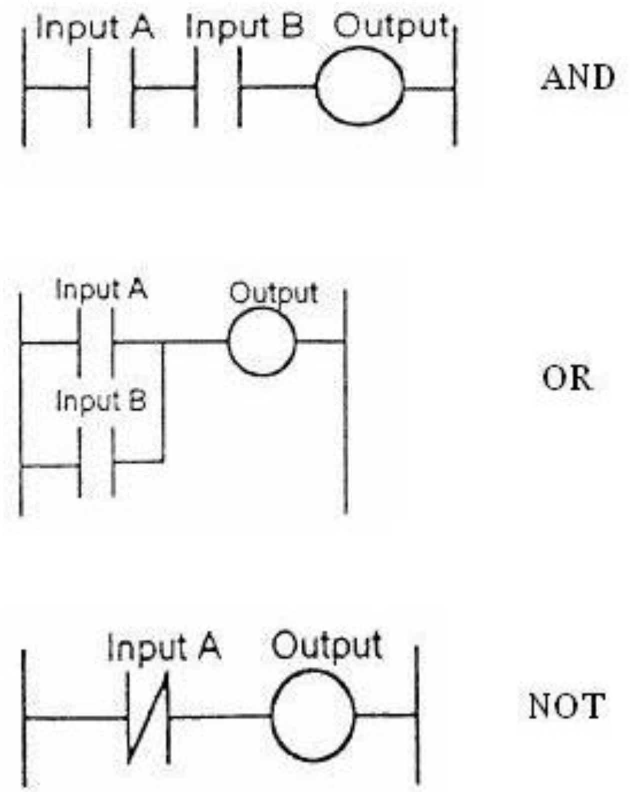


Figure 4.2: Logic Operation in Different Programming Languages.
(source: http://www.plcmanual.com/plc-programming*)*

Ladder diagrams are clear and easily understandable, but they less some special operations and often need to be complemented with connections to program parts created in other languages. Function block diagrams are logic diagrams [9].

4.3. Siemens Step 7

4.3.1. Simatic Manager

The program development during the laboratory for the S7-313C-5BF03 PLC is carried out in the Simatic Manager software. The integrated user interface of Simatic Manager makes it possible to easily edit, compile and debug the operating programs and download them to the PLC. STEP7 includes several low and high-level programming languages *(see figure 4.3):*

- Function Block Diagram (FBD)
- Ladder Diagram (LD)
- Sequential Function Control (SFC)
- Instruction List (IL)
- Structured Text (ST)

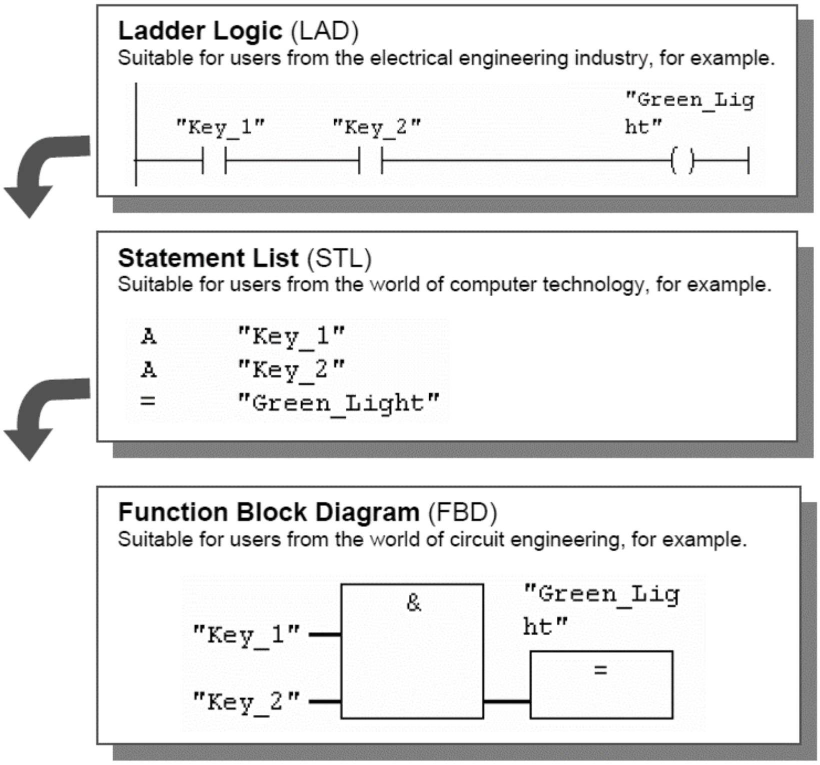


Figure 4.3: Siemens Programming Languages.
(source: Simatic Manager Manual Book)

4.3.2. WinCC

SIMATIC WinCC is a software program for the operator control and monitoring of automated processes. WinCC stands for Windows Control Center, meaning that it provides you with all the features and options of a standard Windows environment. WinCC is a visualization system that can be customized and comes with powerful features, an intuitive user interface, and easy to use configuration functions. SIMATIC WinCC provides a complete basic system for operator control and monitoring. It provides a number of editors and interfaces that allow you to create highly efficient configurations for your specific application. All relevant configuration data is stored in a WinCC project *(see figure 4.4)*.

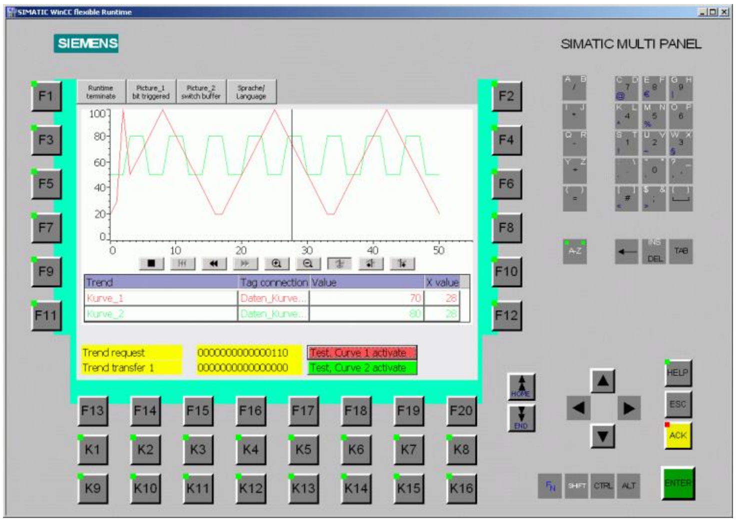


Figure 4.4: Example Siemens WinCC.
(source: https://support.industry.siemens.com*)*

4.3.3. PLC Programming

The PLC configuration and programming is done with the Simatic Manager program. The man-machine interface of the process control system can be created by the WinCC program.

The programming language of the PLC is STEP7. This structured programming language consists of program modules (blocks). There are several types of data *(see table 4.1 & 4.2)*.

Table 4.1: Program Basic Functions.

English Language	Germany Language	Description	Data Type
I/O Signals			
I	E	Input bit	BOOL
IB	EB	Input byte	BYTE, CHAR
IW	EW	Input word	WORD, INT, S5TIME, DATE
ID	ED	Input double word	DWORD, DINT, REAL, TOD, TIME
Q	A	Output bit	BOOL
QB	AB	Output byte	BYTE, CHAR
QW	AW	Output word	WORD, INT, S5TIME, DATE
QD	AD	Output double word	DWORD, DINT, REAL, TOD, TIME
Marker Memory			
M	M	Memory bit	BOOL
MB	MB	Memory byte	BYTE, CHAR
MW	MW	Memory word	WORD, INT, S5TIME, DATE
MD	MD	Memory double word	DWORD, DINT, REAL, TOD, TIME
Peripheral I/O			
PIB	PEB	Peripheral input byte	BYTE, CHAR
PIW	PEW	Peripheral input word	WORD, INT, S5TIME, DATE
PID	PED	Peripheral input double word	DWORD, DINT, REAL, TOD, TIME
PQB	PAB	Peripheral output byte	BYTE, CHAR
PQW	PAW	Peripheral output word	WORD, INT, S5TIME, DATE
PQD	PAD	Peripheral output double word	DWORD, DINT, REAL, TOD, TIME
Timers and Counters			
T	T	Timer	TIMER
C	Z	Counter	COUNTER
Logic Blocks			
FB	FB	Function block	FB
OB	OB	Organization block	OB
FC	FC	Function	FC
SFB	SFB	System function block	SFB
SFC	SFC	System function	SFC
Data Blocks			
DB	DB	Data block	DB, FB, SFB, UDT

(source: http://www.plcdev.com*)*

Table 4.2: Data Types.

Type and Description	Size (Bits)	Format Options	Range and Number Notation (lowest to highest values)
BOOL (Bit)	1	Boolean text	TRUE/FALSE
BYTE (Byte)	8	Hexadecimal number	B#16#0 to B#16#FF
WORD (Word)	16	Binary number	2#0 to 2#1111_1111_1111_1111
		Hexadecimal number	W#16#0 to W#16#FFFF
		BCD	C#0 to C#999
		Decimal number unsigned	B#(0,0) to B#(255,255)
DWORD (Double word)	32	Binary number	2#0 to 2#1111_1111_1111_1111
		Hexadecimal number	W#16#0000_0000 to W#16#FFFF_FFFF
		Decimal number unsigned	B#(0,0,0,0) to B#(255,255,255,255)
INT (Integer)	16	Decimal number signed	-32768 to 32767
DINT (Double integer)	32	Decimal number signed	L#-2147483648 to L#2147483647
REAL (Floating-point number)	32	IEEE Floating-point number	Upper limit +/-3.402823e+38
			Lower limit +/-1.175495e-38
S5TIME (SIMATIC time)	16	S7 time in steps of 10ms (default)	S5T#0H_0M_0S_10MS to
			S5T#2H_46M_30S_0MS and
			S5T#0H_0M_0S_0MS
TIME (IEC time)	32	IEC time in steps of 1 ms	T#24D_20H_31M_23S_648MS
			to
			T#24D_20H_31M_23S_647MS
DATE (IEC date)	16	IEC date in steps of 1 day	D#1990-1-1 to
			D#2168-12-31
TIME OF DAY (Time)	32	Time in steps of 1 ms	TOD#0:0:0.0 to
			TOD#23:59:59.999
CHAR (Character)	8	ASCII characters	A', 'B' etc.

(source: http://www.plcdev.com*)*

4.4. Type of Blocks

The program for SIMATIC S7-300 is written in so-called blocks. The Main organization block is already included as standard. This block represents the interface to the operating system of the CPU and is automatically called and cyclically processed by this operating system. For all control tasks, the program is divided into small, manageable program blocks sorted according to functions of the process. These blocks are then called from organization blocks. At the block end, the program jumps back to the calling organization block-specifically, to the line directly after the call [11].

4.4.1. Organization Block

Organization blocks (OB) is called cyclically by the operating system and thus forms the interface between the user program and the operating system. In this OB, block call commands are used to inform the PLC's sequence control which program blocks it is to process. An OB form the interface between the operating system and the user program. They are called from the operating system and control the following operations:

- Startup characteristics of the automation system
- Cyclic program execution
- Interrupt-driven program execution
- Error handling

I can program the organization blocks as I wish and thus determine the conduct of the CPU. Various options are available for using organization blocks in my program:

- Startup OB, cycle OB, time error OB, and diagnostics OB:

It is easy to insert and program these organization blocks in my project. I do not have to call or assign parameters to these organization blocks.

- Hardware interrupt OB and cyclic interrupt OB:

After these organization blocks have been inserted into my program, I need to assign parameters to them. Hardware interrupt OBs can also be attached to an event during run time or detached.

- Time-delay interrupt OB:

I can insert and program the time-delay interrupt OB in my project. I must additionally call it in the user program using the SRT_DINT instruction. Parameter assignment is not necessary.

- Start information

At the start of some organization blocks, the operating system outputs information that can be evaluated in the user program. This can be very helpful for error diagnostics, in particular. The descriptions of the organization blocks contain information on whether information is output and, if so, what information.

4.4.2. Function

A function contains a program that is executed whenever the function is called from another code block. Functions (FC) are code blocks without memory. After the function has been processed, the data in the temporary tags is lost. Functions can use global data blocks in order to save data permanently. A function does not have an assigned memory area. The local data of a function is lost after the function is processed. It is also possible to call other FB and FC within a function. Functions can be used, for example, for the following purposes:

- To return function values to the calling block, e.g., for mathematical functions.
- To execute technological functions, e.g., individual controls using binary logic.

A function can also be called several times at different points in a program. This facilitates the programming of frequently recurring complex functions.

4.4.3. Function Block

Function blocks contain subroutines that are executed whenever a function block is called by another code block. Function blocks are code blocks that store their values permanently in instance data blocks, so that they remain available even after the block has been processed. They store the in/out parameters and the result and their internal conditions permanently in instance data blocks. Consequently, these parameters are still available after the block has been processed. For this reason, they are also referred to as blocks with "memory" [11].

The FB requires an assigned memory area for each call (instance). When an FB is called, a data block (DB) can be assigned to it as an instance DB, for example. The data in this instance DB is then accessed using the tags of the FB. Various memory areas must be assigned to an FB if it will be called multiple times. It is also possible to call other FB and FC within a function block. Function blocks are used for tasks that cannot be implemented with functions:

- In all cases when timers and counters are required in the blocks. This is can use in function also.
- In all cases when information must be saved in the program. For example: preselection of the operating mode with a button.

A function block can also be called several times at different points in a program. This facilitates the programming of frequently recurring complex functions.

4.4.4. Data Block

In contrast to code blocks, data blocks contain no instructions. Rather, they serve as memory for user data. Data blocks thus contain variable data used by the user program. Global data blocks store data that can be used by all other blocks. The maximum size of data blocks varies depending on the CPU. You can define the structure of global data blocks as required [8].

DBs are used to provide memory space for data tags. There are two types of data blocks. Global DBs where all OB, FB, and FC can read the saved data and themselves write data to the DB, and instance DBs that are assigned to a particular FB. Application examples are:

- Saving of information about a storage system. “Which product is located where?”
- Saving of recipes for particular products.

Every function block, every function, and every organization block can read the data from a global data block or write data to a global data block. This data is retained in the data block, even when the data block is exited.

The call of a function block is referred to as an instance. An instance data block is assigned to every call of a function block that transfers parameters. This instance data block serves as a data memory. It stores the actual parameters and the static data of the function block.

The maximum size of instance data blocks varies depending on the CPU. The tags declared in the function block determine the structure of the instance data block. A global data block and an instance data block can be open at the same time [11].

4.5. Type of Logics

4.5.1. SR Flip Flop

SR flip flop is set if signal state is "1" in S input, and is "0" in R input. Conversely, if the signal state is "0" in S input and signal state is "1" in R input then flip flop will be reset. If the RLO on both inputs is "1", the command is in the first interest. The first SR flip flop executes the instruction set and then reset instructions on <address>, so this address remains reset for program scanning reminders. Set and reset instructions are only executed when RLO is "1". If RLO is "0" then this instruction has no effect and the state of signal <address> is not changed *(see figure 4.5)*.

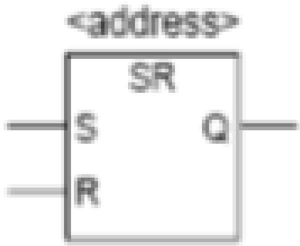


Figure 4.5: SR Flip Flop Logic.
(source: https://tanotocentre.wordpress.com*)*

4.5.2. Timer

As part of its CPU, a PLC has a control clock that can be used to time events and deliver output when certain timer parameters are met. The timers can be represented in ladder logic as output devices with corresponding contacts or as block functions. Notice that the ladder logic programs below perform the same function. The left one uses output devices or coils and the right one uses block functions. The exactly style of timer will depend on the manufacturer and brand of the PLC that the ladder logic development program is used for [8].

There are many types of timers for PLCs. The two most basic types are the *on-delay* and the *off-delay* timer. With an on-delay timer, once the timer coil gains power, its corresponding contact will not turn on immediately. Instead a timer will start and once it is complete only then will the timer contact be in its active state. Anytime the timer coil release power, the timer is reset and the timer contact also release power *(see figure 4.6)*.

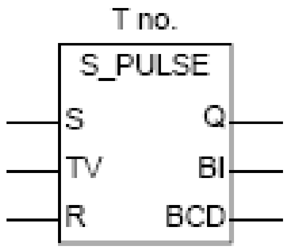


Figure 4.6: Timer Logic.
(source: https://tanotocentre.wordpress.com*)*

4.5.3. Comparison

Ladder logic for PLCs often includes instructions that can compare two values and operate a contact based on the result. If the two values are IN1 and IN2 they can be compared as follows:

- == IN1 is equal to IN2
- <> IN1 is not equal to IN2
- > IN1 is greater than IN2
- < IN1 is less than IN2
- >= IN1 is greater than or equal to IN2
- <= IN1 is less than or equal to IN2

In this case, function blocks are used as the graphical symbol in the ladder logic *(see figure 4.7)*.

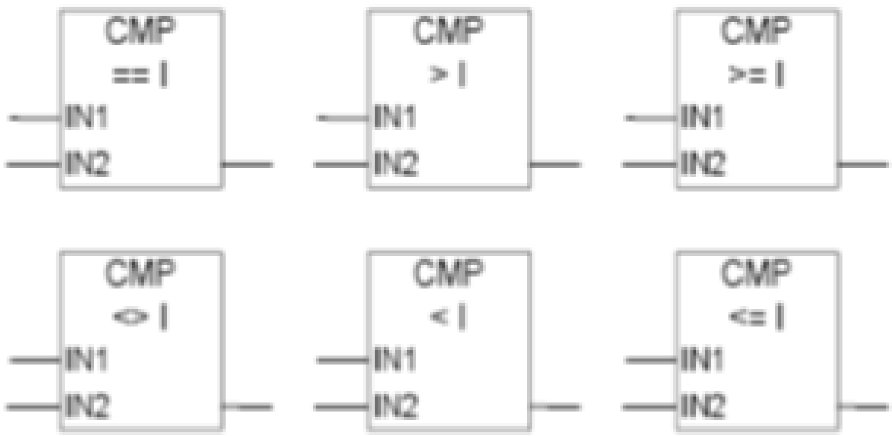


Figure 4.7: Comparison Logic.
(source: https://tanotocentre.wordpress.com*)*

The input values IN1 and IN2 can be programmed directly in to the comparer or they can be taken from other elements in the ladder program such as timers and counters [8].

4.5.4. Move

The move instruction is enabled by Enable Input (EN). The (IN) source input value is copied to the address specified in the OUT parameter. ENO has the same logical state as EN. MOVE can only copy data objects BYTE, WORD, or DWORD. Data types such as aray or structure must be copied with the system function *(see figure 4.8)*.

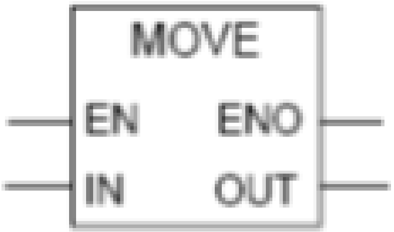


Figure 4.8: Move Logic.
(source: https://tanotocentre.wordpress.com*)*

4.5.5. Counter

A counter is used to count and store the number of occurrences of an input signal. Like a timer, a counter can be represented by an output device and contact pair or a block function. A counter is set to some preset number value and when this value of input pulses has been received it will operate its associated contacts.

There are two types of counters: up-counters and down-counters. An up-counter starts a zero and count up to a preset limit. A down-counter starts a preset limit and counts down to zero. When the counter reaches the set value, its contacts are activated. Counters also have a reset coil that when activated re-initialized the current count value. For up-counters it is set to zero. For down counters a reset would set the current count value to the preset *(see figure 4.9)*.

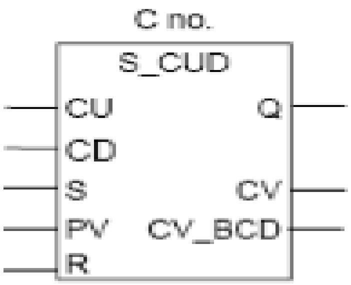


Figure 4.9: Counter Logic.
(source: https://tanotocentre.wordpress.com*)*

4.6. Hardware

The VIPA 313C-5BF03 PLC is from the VIPA 300 family of low-middle category compact controllers. The PLC has been designed for innovative systems solutions with the emphasis on production engineering, and is a universal automation platform providing optimum solutions for applications with central or distributed designs *(see figure 4.10)*.

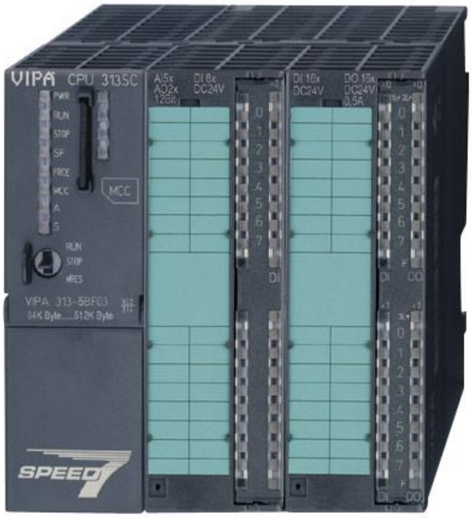


Figure 4.10: VIPA 313C-5BF03.
(source: http://www.vipa.at/index.php?lang=E&category=data&products=141*)*

4.6.1. Structure

This is the structure of VIPA 313C-5BF03 *(see figure 4.11)*.

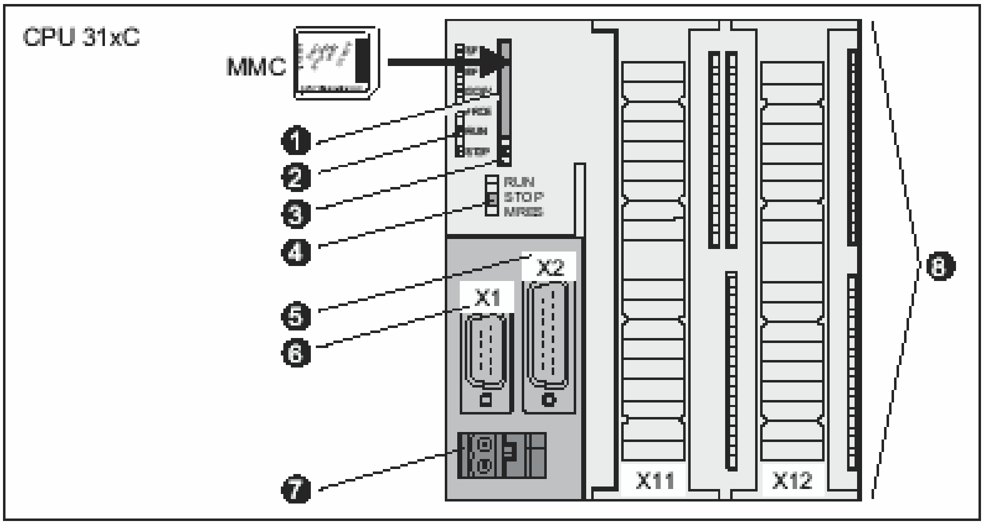


Figure 4.11: The PLC Front Panel.
(source: Application Techniques of Logic Controllers)

1. Slot for the SIMATIC Micro Memory Card.
2. Status and error displays.
   - SF – Hardware or software error (red)
   - BF – PROFIBUS error (red)
   - 5VDC – 5-V power for CPU and S7-300 bus is OK (green)
   - FRCE – Force job is active (yellow)
   - RUN – in start mode it flashes, in normal mode on (green)
   - STOP – CPU in STOP and HOLD or STARTUP (green)
3. MMC (Micro Memory Card) ejector.
4. Mode selector.
   - RUN – The CPU executes the user program.
   - STOP – The CPU does not execute a user program.
   - MRES – CPU memory reset.
5. X2 PROFIBUS connector.
6. X1 MPI connector.
7. 24V-os connection.
8. Connections of the integrated I/O.

4.6.2. Properties

- Speed technology integrated.
- Integrated 24V power supply unit.
- 64kB total memory (32kB code, 32kB data).
- Memory expandable to max 512kB (256kB code, 256kB data).
- MCC slot for external memory cards and memory extension.
- Status LED for operating state and diagnosis.
- Real time clock battery buffered.
- Ethernet PG/OP interface integrated.
- RS 485 interface configuration for PtP communication
- Digital I/O: DI 24xDC 24V / DO 16xDC 24V, 0.5A
- Analog I/O: AI 4x12Bit / AO 2x12Bit / AI 1xPt100

4.6.3. Address Assignment

This is the address assignment of VIPA 313C-5BF03 *(see table 4.3 & 4.4)*.

Table 4.3: Input Range.

Sub Module	Default Address	Access	Assignment
DI24/DO16	124	Byte	Digital Input I+0.0 … I+0.7
	125	Byte	Digital Input I+1.0 … I+1.7
	126	Byte	Digital Input I+2.0 … I+2.7
AI5/AO2	752	Word	Analog Input Channel 0
	754	Word	Analog Input Channel 1
	756	Word	Analog Input Channel 2
	758	Word	Analog Input Channel 3
	760	Word	Analog Input Channel 4
Counter	768	DInt	Channel 0: Count Value / Frequency Value
	772	DInt	Channel 1: Count Value / Frequency Value
	776	DInt	Channel 2: Count Value / Frequency Value
	780	DInt	Reserved

(source: VIPA 313C-5BF03 Handbook)

Table 4.4: Output Range.

Sub Module	Default Address	Access	Assignment
DI24/DO16	124	Byte	Digital Output Q+0.0 … Q+0.7
	125	Byte	Digital Output Q+1.0 … Q+1.7
AI5/AO2	752	Word	Analog Input Channel 0
	754	Word	Analog Input Channel 1
Counter	768	DWord	Reserved
	772	DWord	Reserved
	776	DWord	Reserved
	780	DWord	Reserved

(source: VIPA 313C-5BF03 Handbook)

4.7. Discussion

Testing of engine performance is often important in the development of engine. Many parameters affect an engine's performance: the basic engine design, speed, torque & power. Proper tuning of an engine requires careful measurements of power, torque and speed as a function of throttle position.

In this study, measurement of speed, power and torque as function of throttle position is required. In above system overview testing of engine performance in the laboratory shows that the engine under test is coupled to a dynamometer. The dynamometer, which provides a load to the engine, can be easily controlled to allow testing under a wide range of speeds and torque.

If load increases engine speed decreases & torque increases because load cell is kept inside dynamometer. Three sensors are used such as throttle position sensor, magnetic pick up tachometer as speed sensor, load cell for torque measurement.

An integral component of a dynamometer is its data acquisition system. The system is typically comprised of two units, a Commander and Workstation, connected by an Ethernet cable. The Workstation operates the precision load and throttle control systems, collects the data, and sends it to the Commander to be processed, stored and analyzed *(see figure 4.12)*.

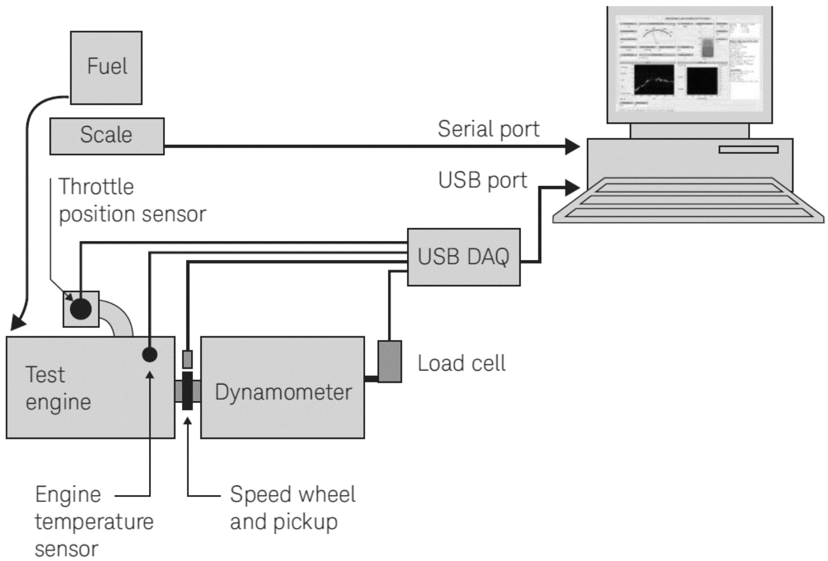


Figure 4.12: Schematic System.
(source: Dynamometer Basics, Dr. Horizon Gitano)

The basic operation of a water brake dyno uses the principle of viscous coupling. The output shaft of the engine is coupled to a fan that spins inside a concentric housing. While the engine is running, the housing is filled with a controlled amount of water. The more water that is allowed into the housing, the more load the engine will feel *(see figure 4.13)*.

As the fan spins through the water, the water is whipped around as well. Newtons 3rd Law says that the water will push on by the housing with the equal and opposite force that the fan is pushing on the water. In this case, the shear forces in the water are acting tangential to the housing radius. There is a load cell at a measured distance from the center of the housing. The load cell is also oriented perpendicular to the arm extending from the housing. The torque output of the engine is just the force measured at the load cell multiplied by the distance to where the extended arm and load cell connect.

There is also a data acquisition system on the dyno. The data acquisition systems sensors measure force, input shaft RPM, oil pressure, fuel pressure, fuel flow rate. From these sensors, math channels are set up to calculate whatever you want. A few of the most useful things this system calculates are torque, horsepower, and brake specific fuel consumption. Horsepower is just a function of torque and RPM. This is realized in the water brake system in laboratory building of the Maritime Department of Hochschule Wismar.

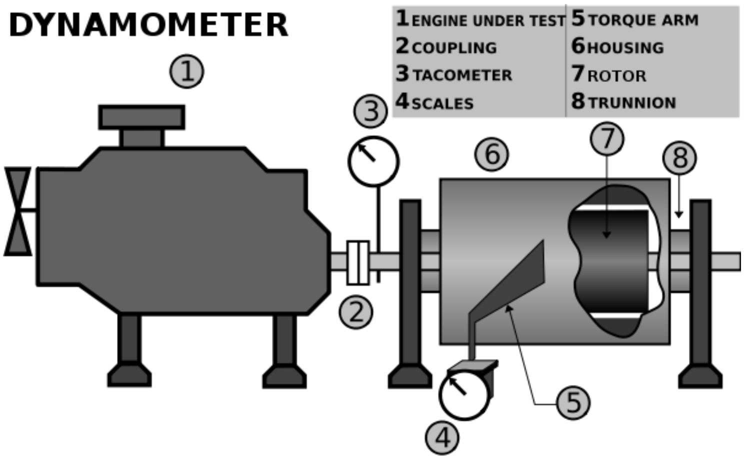


Figure 4.13: Dynamometer.
(source: https://www.setra.com/blog/test-and-measurement-dynamometer*)*

If the units are in Newton-meters and shaft speed (S) is measured in radians per second, then the shaft power or break power (P) of the engine can be calculated by multiplying the speed and the torque (T):

$$P = T \times S$$

In Speed Controlled mode a set speed is given to the controller, if the measured speed of the shaft is less than that of the set speed, the load is decreased. If the measured speed of the shaft is greater than that of the set speed, then the load is increased. Assuming the engine has sufficient torque to attain the set speed, this will maintain a constant speed of engine which is under test. Where throttle valve is 100% open to test speed parameter of engine, by varying load & keeping speed constant of main engine.

In Load Controlled mode a set load is given to the controller. If the measured load on the dynamometer is greater than that of the set load, the load is decreased. If the measured load on the dynamometer is less than that of the set load, then the load is increased. Assuming the engine has sufficient torque to attain the set load, this will maintain a constant load while the speed varies.

An Engine is a device which transforms chemical energy of fuel into thermal energy & uses thermal energy to produce mechanical work. Engine normally converts thermal energy in to mechanical work. This thesis shown testing of engine using dynamometer. An engine dyno calculates power output directly by measuring the force (torque) required to hold a spinning engine at a set speed (rpm) means through engine behavior as shown in below table I here two quantities are under controlled such as torque & speed [15].

4.8. Analysis

4.8.1. Collect Data

Before made some analysis for this thesis, I collect all of the data from Laboratory Building (House 5) of the Maritime Department of Hochschule Wismar. I got the main engine data *(see table 4.5)*, water brake data *(see table 4.6)*, main engine and water brake performance *(see diagram 4.1) (see attachment 1)* from Mr. Hartmut Schmidt. This data as a guide to take the next step to make some calculation and plc programming.

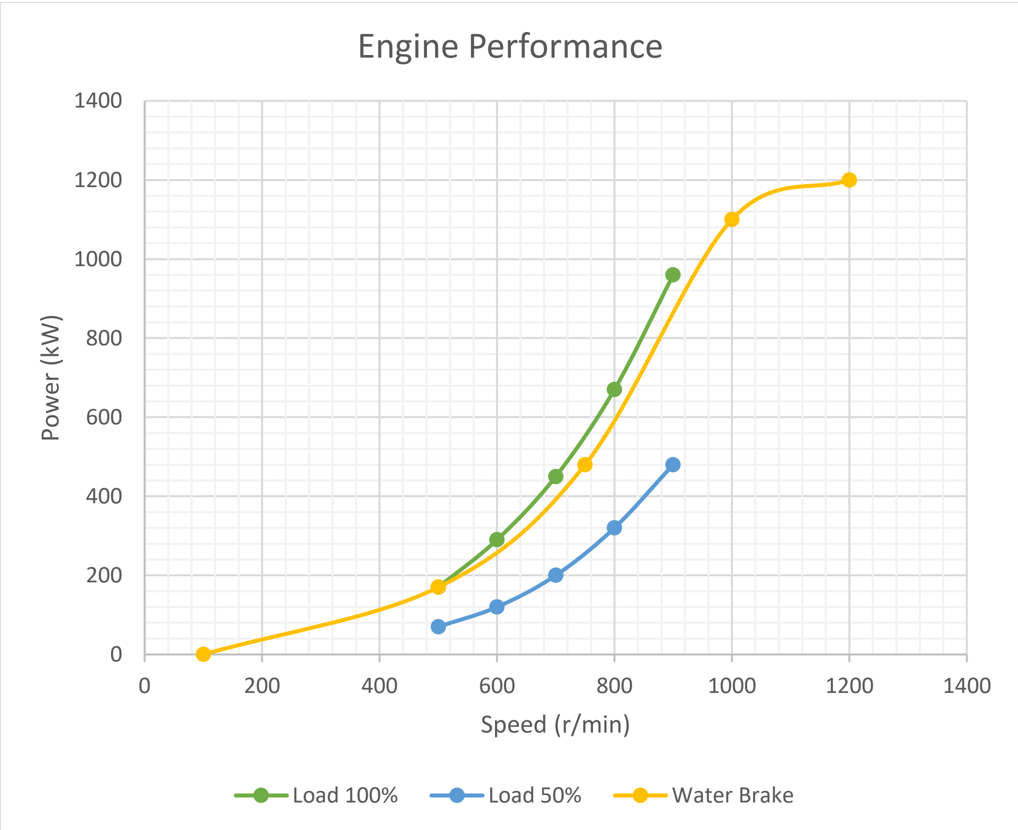


Diagram 4.1: Main Engine & Water Brake Performance.
(source: Laboratory Building of the Maritime Department of Hochschule Wismar)

Figure 4.14: MAN B&W 6L23/30A.
(source: Laboratory Building of the Maritime Department of Hochschule Wismar)

Table 4.5: MAN B&W 6L23/30A Basic Data.

Engine Data		
Category	**Content**	**Unit**
Manufactur	MAN B&W	
Type	6L23/30A	
Cylinder	6-4 Stroke	
Power	960	kW
Power/Cylinder	160	kw
Speed	900	r/min
Stroke	300	mm
Bore	225	mm
Stroke/Bore	1.33	ratio
Compression	12.5	ratio
SFOC	194	g/kWh
SLOC	1	g/kWh
Dry mass	11	Ton

(source: Marine Engine IMO Tier II and Tier III Programme 2017. MAN Diesel & Turbo)

Figure 4.15: Zollner 9N38/12F.
(source: Laboratory Building of the Maritime Department of Hochschule Wismar)

Table 4.6: Zollner 9N38/12F Basic Data.

Water Brake Data		
Category	**Content**	**Unit**
Manufacture	Zollner	
Type	9N38/12F	
Power	1200	kW
Speed	3500	r/min
Torque	7161	Nm
Flow	25	m3/h

(source: Zollner 9N38/12F Project Guide)

4.8.2. Data Variations

The next step is make some variations data. I make some variations data regarding to visualization from the data block of PLC in Laboratory Building (House 2) of the Maritime Department of Hochschule Wismar.

First thing I have to do is open the file in the data block of PLC. And then look the address about the power and the speed of the engine. I save every address for every load, from 0% of load until 100% of load in my PLC program. After that stored the value that I got from Laboratory Building (House 5) of the Maritime Department of Hochschule Wismar into my PLC program. And this is the data I make in my PLC program *(see table 4.7)*.

Table 4.7: Engine Performance.

Load	Axis (Y)	Axis (X)	Axis (Y)
	Torque	Speed	Power
%	Nm	r/min	kW
0%	0	0	0
10%	2196	417,7	96
20%	3485	526,3	192
30%	4567	602,5	288
40%	5533	663,1	384
50%	6420	714,3	480
60%	7250	759,1	576
70%	8035	799,1	672
80%	8782	835,5	768
90%	9500	868,9	864
100%	10191	900	960

(source: Visualization from the Data Block of PLC)

After stored the address and value into my PLC program. I transfer the load characteristics (characteristic points) from visualization, I make some calculation of the intermediate values by interpolation to get some variations data for various load states. For this calculation using this formula:

$$Y = Y1 + ((X - X1) / (X2 - X1) \times (Y2 - Y1)$$

X is the value I set in my PLC program. In this case I set the speed value to get the actual power. Actual power is Y in this formula. The value of X1 is the lower value of speed than the value of X. The value of X2 is the bigger value of speed than the value of X. So, the value of X is located between X1 and X2. The value of X1 and X2 is regarding to the table *(see table 4.7)*.

The value of Y1 is the lower value of power than the value of Y. The value of Y2 is the bigger value of power than the value of Y. So, the value of Y is located between Y1 and Y2. The value of Y1 and Y2 is regarding to the table *(see table 4.7)*. The X axis is speed of engine and the Y axis is the power of engine.

For example, if I want to get the power between load 10% and load 20% so for example I set the value of actual speed in the 450 rpm. The value of X1 is 417,7 (load 10%) and the value of X2 is 526,3 (load 20%). The value of Y1 is 96 kW (load 10%) and the value of Y2 is 192 kW (load 20%). Regarding to the formula, so we can get the value of actual power is 124 kW. This value is located between load 10% and load 20% of engine power.

After I make some interpolation calculation using that formula I make some various load states *(see attachment 2) and then, make the diagram to show the various load states using interpolation (see diagram 4.2)*. To make the diagram easy to read so I make two diagrams and to make comparison every load. First diagram is Variations Interpolation Between Power and Time *(see diagram 4.3)*. Second diagram is Variations Interpolation Between Speed and Time *(see diagram 4.4)*. I choose time for variable comparison because time prove it. I make diagram for every 100 second per interpolation off percentage load states. For example, in the 10% of load in 100 second produce more power in a longer time than 20% of load. And then, make some comparison between power, speed and torque *(see diagram 4.5)*.

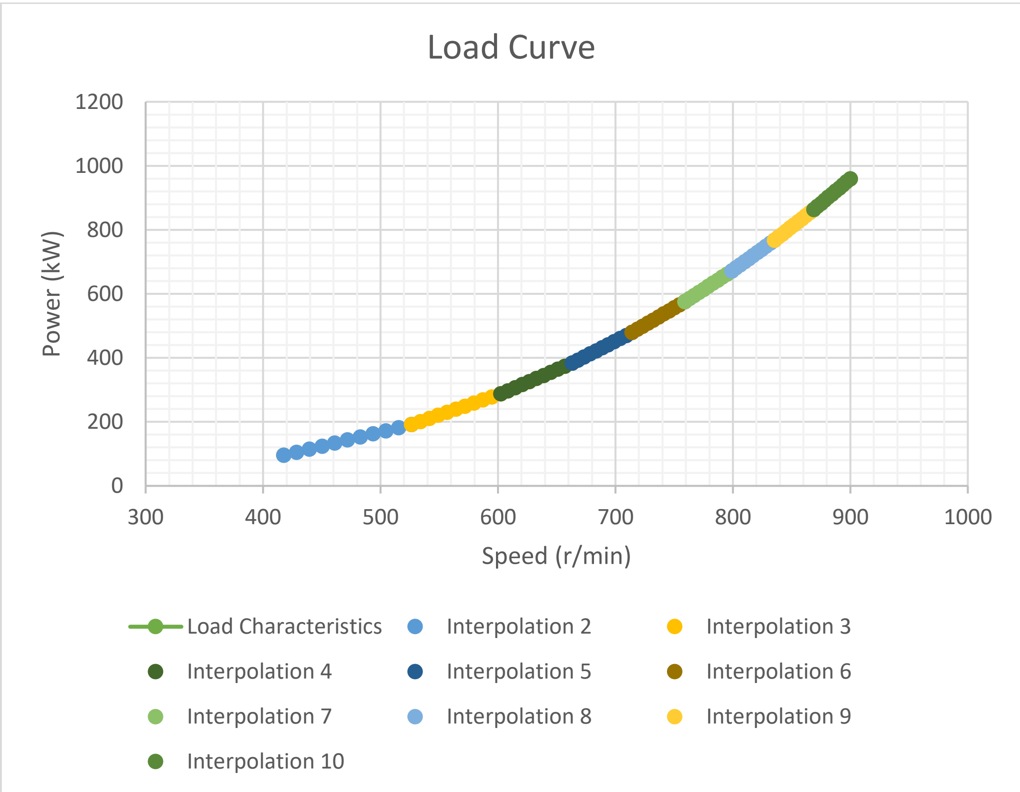


Diagram 4.2: Load Curve.
(source: Visualization from the Data Block of PLC)

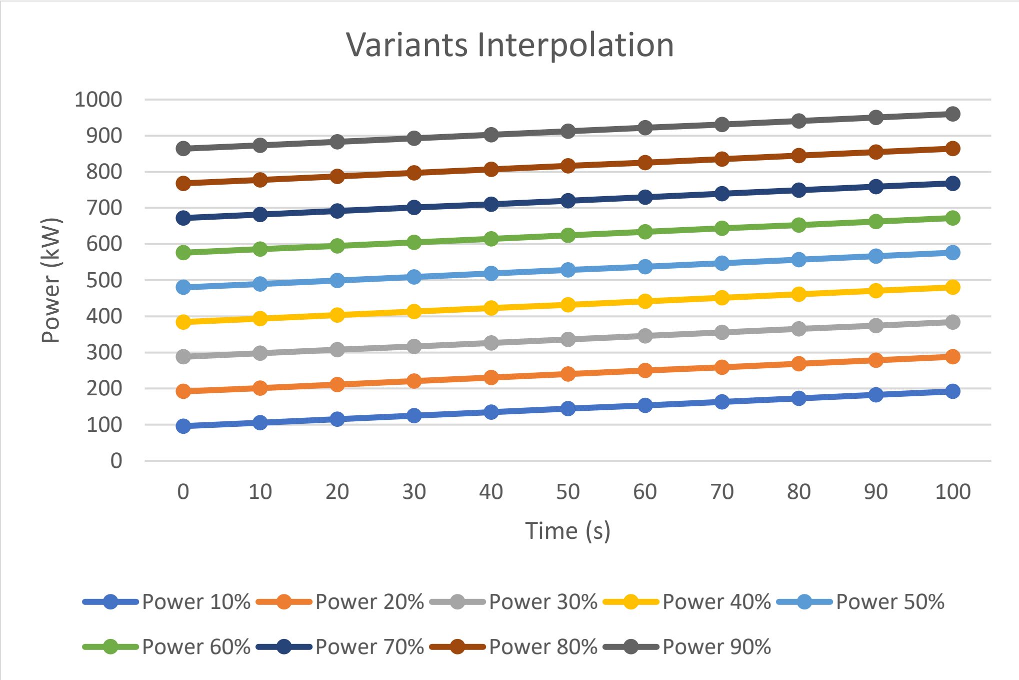


Diagram 4.3: Variations Interpolation Between Power and Time.
(source: Own Calculation)

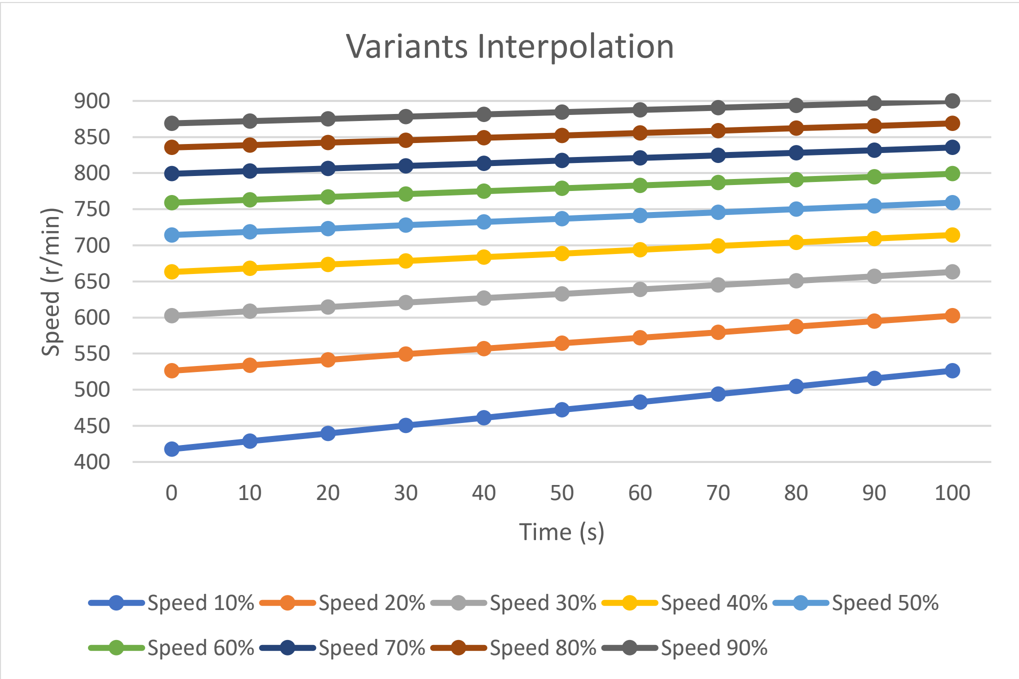


Diagram 4.4: Variations Interpolation Between Speed and Time.
(source: Own Calculation)

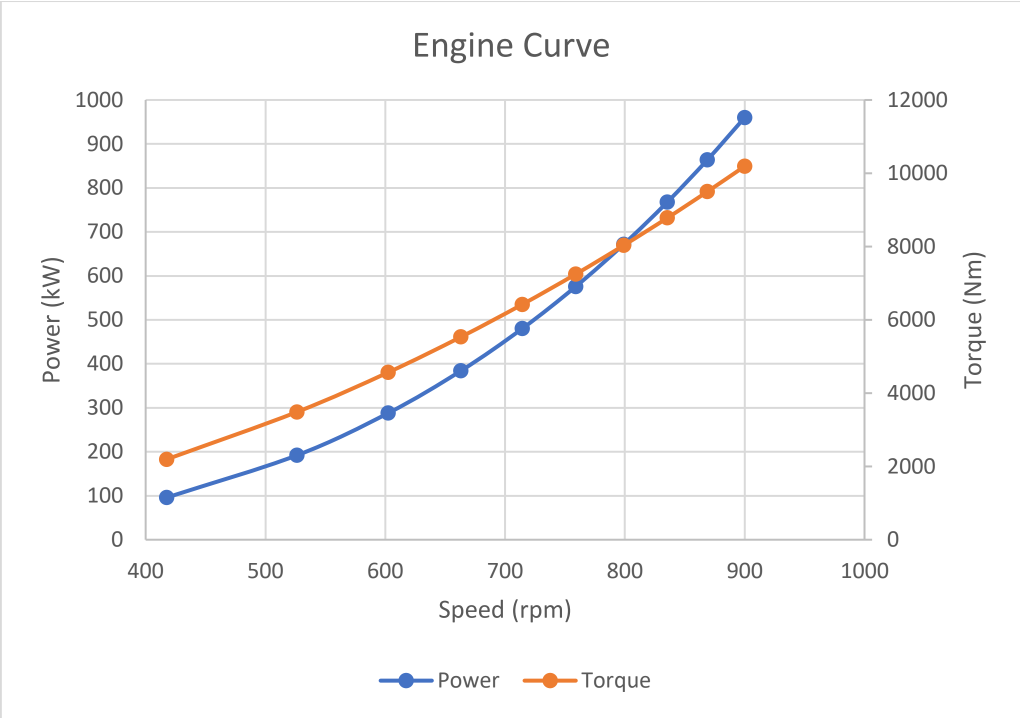


Diagram 4.5: Engine Curve Between Power, Speed and Torque.
(source: Own Calculation)

Regarding the data I got from Laboratory Building (House 5) of the Maritime Department of Hochschule Wismar *(see diagram 4.1)*. The maximum capacity that can be produce from water brake is 90% of load. So, I can make some comparison producing power use water brake to give load to the main engine *(see table 4.8)*. For example, if I setup water brake in 10% of load we get the power in table 10 % of power. And I make comparison of main engine from 10% of power until 100% of power *(see diagram 4.6)*.

Table 4.8: Load Characteristic.

Load	Axis (X)	Axis (Y)									
	Speed	Power									
%	r/min	kW									
		100%	90%	80%	70%	60%	50%	40%	30%	20%	10%
0	0	0	0	0	0	0	0	0	0	0	0
10	417,7	96	86,4	76,8	67,2	57,6	48	38,4	28,8	19,2	9,6
20	526,3	192	172,8	153,6	134,4	115,2	96	76,8	57,6	38,4	19,2
30	602,5	288	259,2	230,4	201,6	172,8	144	115,2	86,4	57,6	28,8
40	663,1	384	345,6	307,2	268,8	230,4	192	153,6	115,2	76,8	38,4
50	714,3	480	432	384	336	288	240	192	144	96	48
60	759,1	576	518,4	460,8	403,2	345,6	288	230,4	172,8	115,2	57,6
70	799,1	672	604,8	537,6	470,4	403,2	336	268,8	201,6	134,4	67,2
80	835,5	768	691,2	614,4	537,6	460,8	384	307,2	230,4	153,6	76,8
90	868,9	864	777,6	691,2	604,8	518,4	432	345,6	259,2	172,8	86,4
100	900	960	864	768	672	576	480	384	288	192	96

(source: Own Calculation)

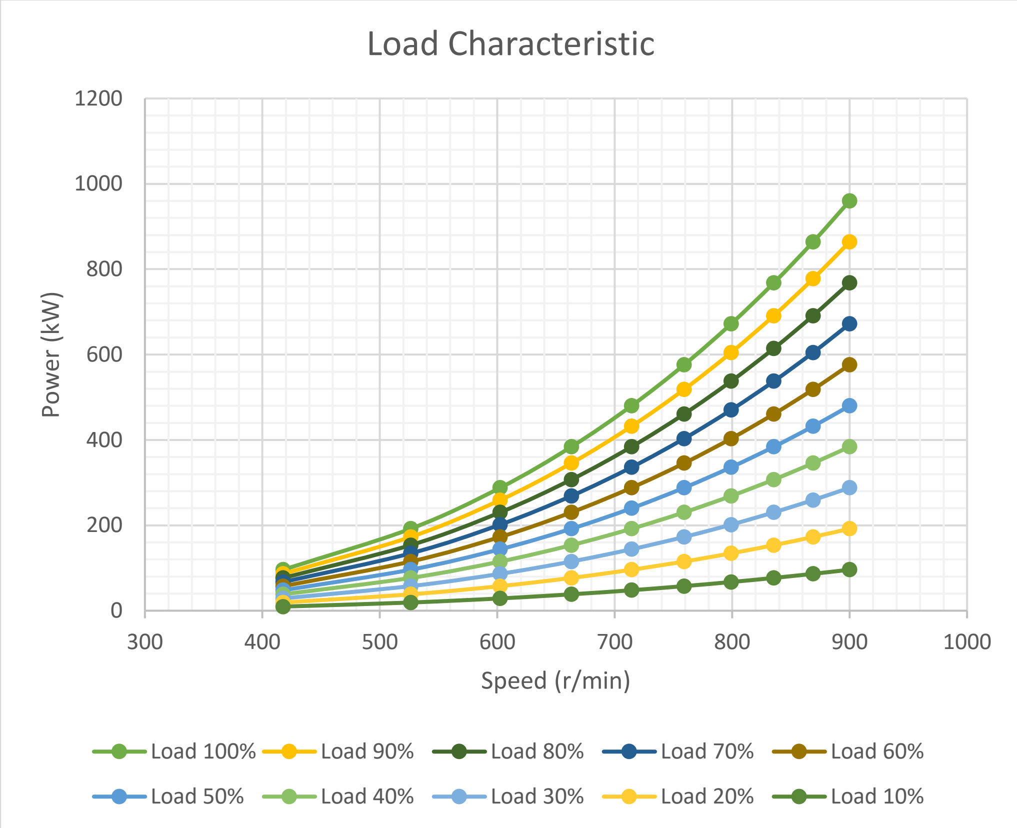


Diagram 4.6: Load Characteristic.
(source: Own Calculation)

4.8.3. PLC Programming

After got the data from visualization and make some variations data using interpolation. The final step is programming and storing of the load characteristics in the PLC. Before I can store the load characteristics, I have to make the program.

The first thing I have to do is add the hardware. And then add the rack rail to make a slot to input the PS (power supply) and CPU (Central Processing Unit). I choose PS 307 5A and CPU 313C DI24/DO16 AI5/AO6. Because the device in the Laboratory Building (House 2) of the Maritime Department of Hochschule Wismar using that device.

After I add the PS and CPU I have to connect the computer and the device with make the same IP address between the computer and the device. Before I make the same IP address I have to input ethernet to the slot in the rack rail. I choose CP 343-1 for the ethernet *(see attachment 3)*. After I add the hardware and connect the computer with the hardware, I can start to make the symbols which will be used in my program *(see add on attachment)*.

After finish to make all of symbols to use in the program, so I can start to make the all of function to make the program works. First thing I have to do is make OB1. OB1 is the main program. All of function will save and call in the OB1. In the OB1 I have all of function which will be used to make the program works.

After that I make all of function which is contained all of input, output, and memory. In the FC1 I insert all of input I will use in my program like engine button on and off, stop button and emergency stop button. In the FC2 I make all of memory about engine operation like engine will be turn on if I turn on the engine button on. The engine will be turn off if I turn off the engine button off.

In the FC3 I make all of memory about power operation of engine. In the FC4 I make all of memory about speed operation of engine. In the FC6 I insert all of output I will use in my program like the engine turn on and the alarm will be turn on if the engine in the over speed condition. That function will be use if the PLC program is connected with the visualization program which have all of the button and the visualization interface. For the function *(see attachment 4)*.

In the FC5 I make all of calculation to decide the input of set value get into which range. For the example if we set the value in 450 rpm, the input will be get into range 1. Because the value of range 1 is between 418 rpm (load 10%) until 526 rpm (load 20%). For more details *(see add on attachment)*.

After I input some set value of speed and the FC5 decide the set value of speed get into which range. The output is will be jump to another function (FC101-FC109). In the FC101-FC109 I make all of calculation for every range (range 1-9). Range 1 is between load 10% - load 20% and so on. For the example, after input the set value of speed in the 450 rpm FC5 will be decide this value get into FC101. Because in FC101 I make all of calculation to make the set value of power using interpolation formula. The value which is used in the interpolation formula regarding to reference value *(see table 4.7)*.

And then the output of FC101-FC109 to show the result for interpolation calculation. For example, is the input is 450 rpm (actual value of speed) so the output is 124 kW (actual value of power). The speed input in the visualization is just for the test, I must use the Input from the speed interface. For the visualization *(see figure 4.16)*.

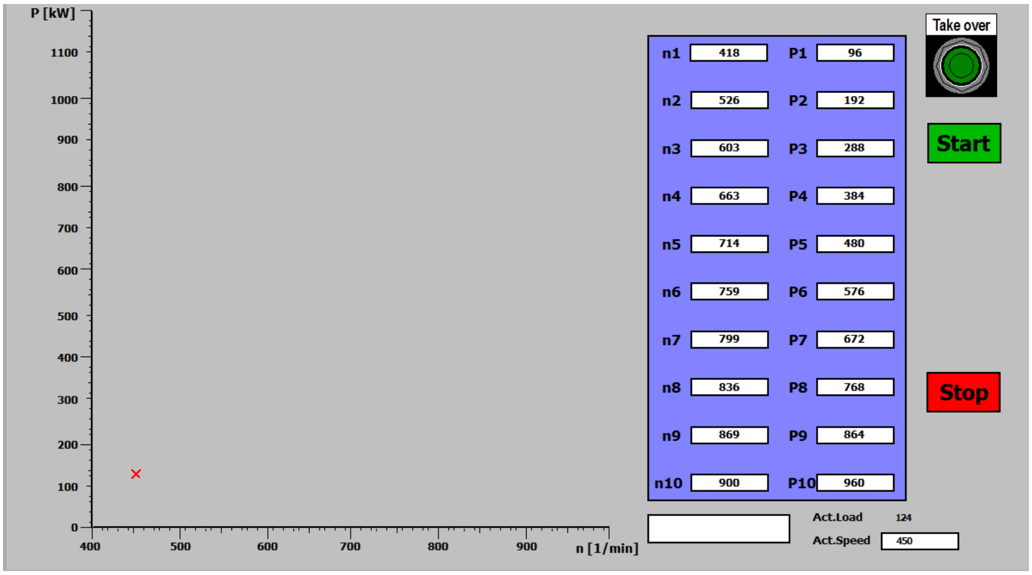


Figure 4.16: Visualization.
(source: Visualization test using Mr. Dannemann program)

In the FC7 I make all of function to move the reverence value of power from DB3 into DB7 to make visualization. In the FC8 I make all of function to move the reverence value of speed from DB4 into DB6 to make visualization. This purpose if I change the reference value of power and speed, in the visualization no problem will occur. After that, activated the take-over button to change the value in visualization before into the new value.

The reference value of power I save in the DB7 and reference value of speed I save in the DB6. And the result for all of calculation I save in the DB5. In DB4 I can change the percentage of water brake load. For example, if I set the percentage of water brake load in the 100%. The input value of speed is 418 rpm so the value of power is 96 kW. But if I set the percentage of load water brake in the 90%. So, the value of power will be decrease into 86 kW. For more details *(see add on attachment)*.

Before I can run using the device (online) I can run using simulation (offline). And then before simulated the program I have to download all of the function into simulation. After that to check the program is works using RUN-P. EB is input Bits and AB is output in Bits. The address is 124 because I have to match the address with the hardware. MB is the memory in Bits. T is timer and Z is counter *(see figure 4.17)*.

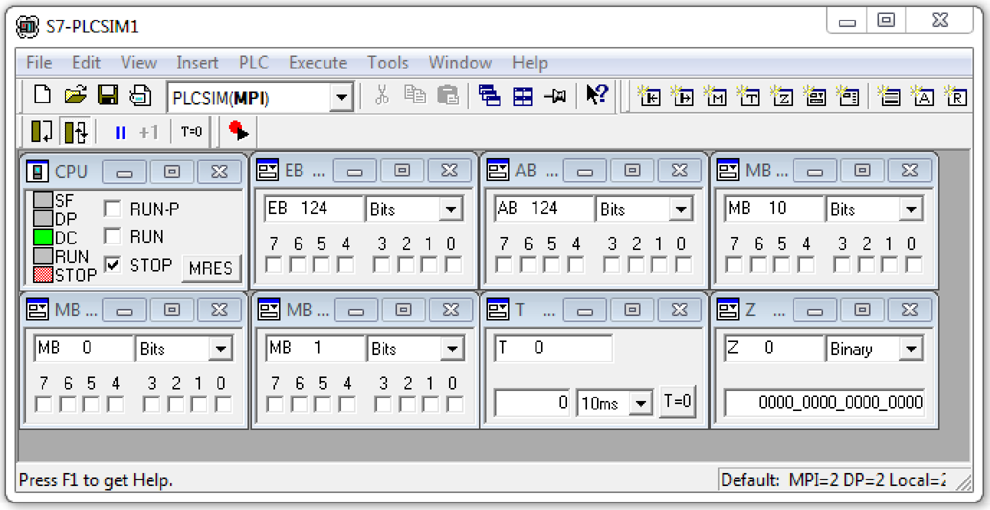


Figure 4.17: Simulation.
(source: Simulation of Simatic Manager)

"This page intentionally left blank"

CHAPTER 5
CONCLUSION

On this day, PLC can solve many problems in various fields. All systems that are in all places are almost compatible with PLC. For the examples such as in buildings, public transportation, industry and so forth. You should also realize that right now, if you had a PLC and enough ladder logic knowledge you could construct a sophisticated machine with little problem as far as operational logic is concerned.

PLC are flexible and can be reapplied to control other systems quickly and easily. They are cost effective for controlling complex systems. They possess high computational ability that allows more sophisticated control through ladder logic. Trouble shooting aids make programming easier and reduce downtime. Reliable components make PLCs likely to operate for years before failure.

With a variety of basic functions of existing programming can be made a useful program. This program can be connected to the device. Where this program can run a visualization to show load characteristics of the engine. How its work is give the set value of speed and we will get the actual value of power. It is obtained from the calculation of interpolation that everything is done automatically by PLC program that has been made *(see add on attachment)*.

Regarding to the diagram that already made *(see diagram 4.2, 4.3 and 4.4)*, we know the characteristic of the engine. In the low rpm to reach the set value rpm need a lot of time than to reach the set value rpm in the high rpm. I choose time for variable comparison because time prove it.

So, the function of dynamometer *(water brake)* is a load device which is generally used for measuring the power output of an engine. And to make water brake give load to the engine we don't use hand. But we use automatic system (PLC) with operator.

The aim of this thesis to analysis of possible load states of the main engine which can be provided by the water break is complete realized *(see diagram 4.6 and table 4.8)*. And then to make the PLC compatible is complete realized use test *(see figure 4.16 & 4.17)*. And for the calculation of interpolation to get intermediate value is complete realized *(see attachment 2)*.

"This page intentionally left blank"

REFERENCES

[1] Water Brake Dynamometers. Switzerland: 2013. Phenix Technologies.

[2] Toepfer, Josiah: *What Kind of Engines are Used in Big Ships*. www.quora.com *(2015)*

[3] The Marine Diesel Propulsion System. 2002. Robust Control of Diesel Ship Propulsion.

[4] Application Note AC Synchronous Generator. Tampa, Florida. PdMA Corporation.

[5] Manney, David: *Breaking Down the Difference Between AC and DC Generators*. www.lselectric.com *(2015)*

[6] Bolotinha, Manuel: *Basic Theory of Generators*. www.linkedin.com *(2016)*

[7] Gitano, Horizon: *Dynamometer Basics*. University Science Malaysia, 2008

[8] Odesie: *Programmable Logic Controller (PLC)*. www.myodesie.com

[9] *Laboratory Exercises of Process Engineering Programmable Logic Controller in a dosing process*. University of Oulu, Department of Process and Environmental, Engineering Systems Engineering Laboratory, 2013

[10] Kaushal, Neha: *Role of PLC in Automation and its Various Applications*. Department of Electrical Engineering, Bhutta Group of Institutions, Punjab, India, 2015

[11] SCE Training Curriculum for Integrated Automation Solution Totally Integrated Automation (TIA). TIA Portal Modul, 2013. Siemens.

[12] Tanoto, Eko Wahyu: *Programmable Logic Controller*. www.tanotocentre.wordpress.com (2010)

[13] Quick Start Manual Vipa CPU 313SC. Herzogenaurach: 2008. VIPA.

[14] Kulkarni, Pradnya & Kadu, Chandrakant: *PID Based Dynamometer Used in Engine Test*. (2015)

[15] Lish, Tom: *What is a Dynamometer and how does it work*. www.setra.com/blog/test-and-measurement-dynamometer (2015)

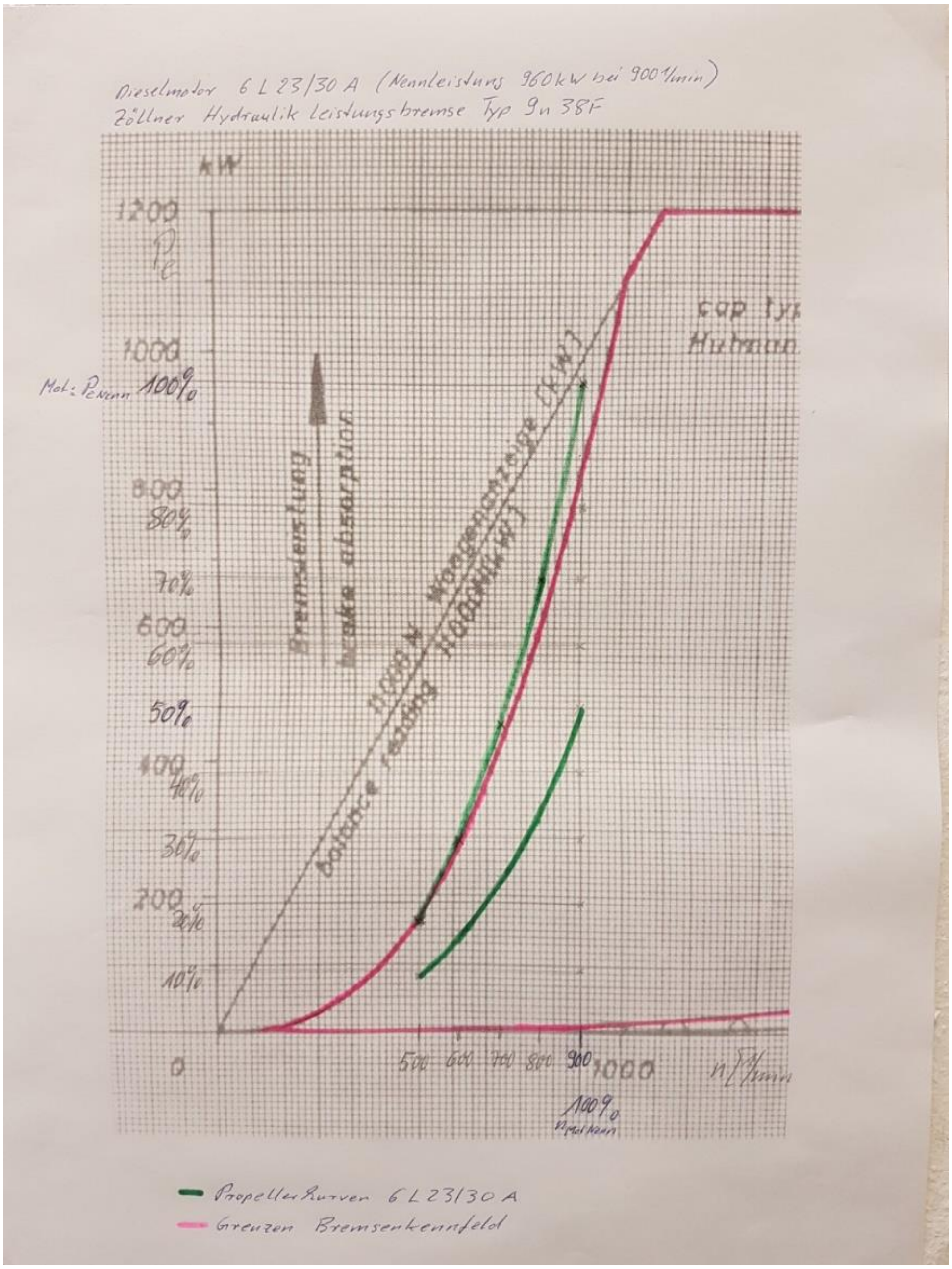
"This page intentionally left blank"

AUTHOR BIOGRAPHY

Muhammad Faiz Rifqi Ardyatama born in Semarang, 29 November 1994. The author is the first child of three sons. Author father's name is Abdul Rasyid Chalik and author mother's name is Dyah Yekti Nugraheni. Author brother's name is Ivan Dwi Hascaryo Ardynugroho and author sister's is Ardyatifa Siti Nurakhim The author accomplished his formal education in SD Islam Al-Azhar 14 Semarang, SMP Islam Al-Azhar 14 Semarang, and SMA Negeri 4 Semarang. After three years in SMA Negeri 4 Semarang, the author proceeds to bachelor degree program in Double Degree Marine Engineering Department, Faculty of Marine Technology, Institut Teknologi Sepuluh Nopember (ITS) Surabaya with Hochschule Wismar Germany. The author has done his On The Job Training (OJT) in Bandar Abadi Shipyard Batam and Pertamina Trans Kontinental (PTK) Jakarta. The author has joined Marine Solar Boat Team (MSBT) in 2014 to do some competion in Amsterdam. In his final year, the author placed in Marine Machinery System (MMS) in Indonesia and Laboratory Operation University of Applied Sciences Wismar in Warnemünde, Rostock, Germany for bachelor thesis construction and discussion.

"This page intentionally left blank"

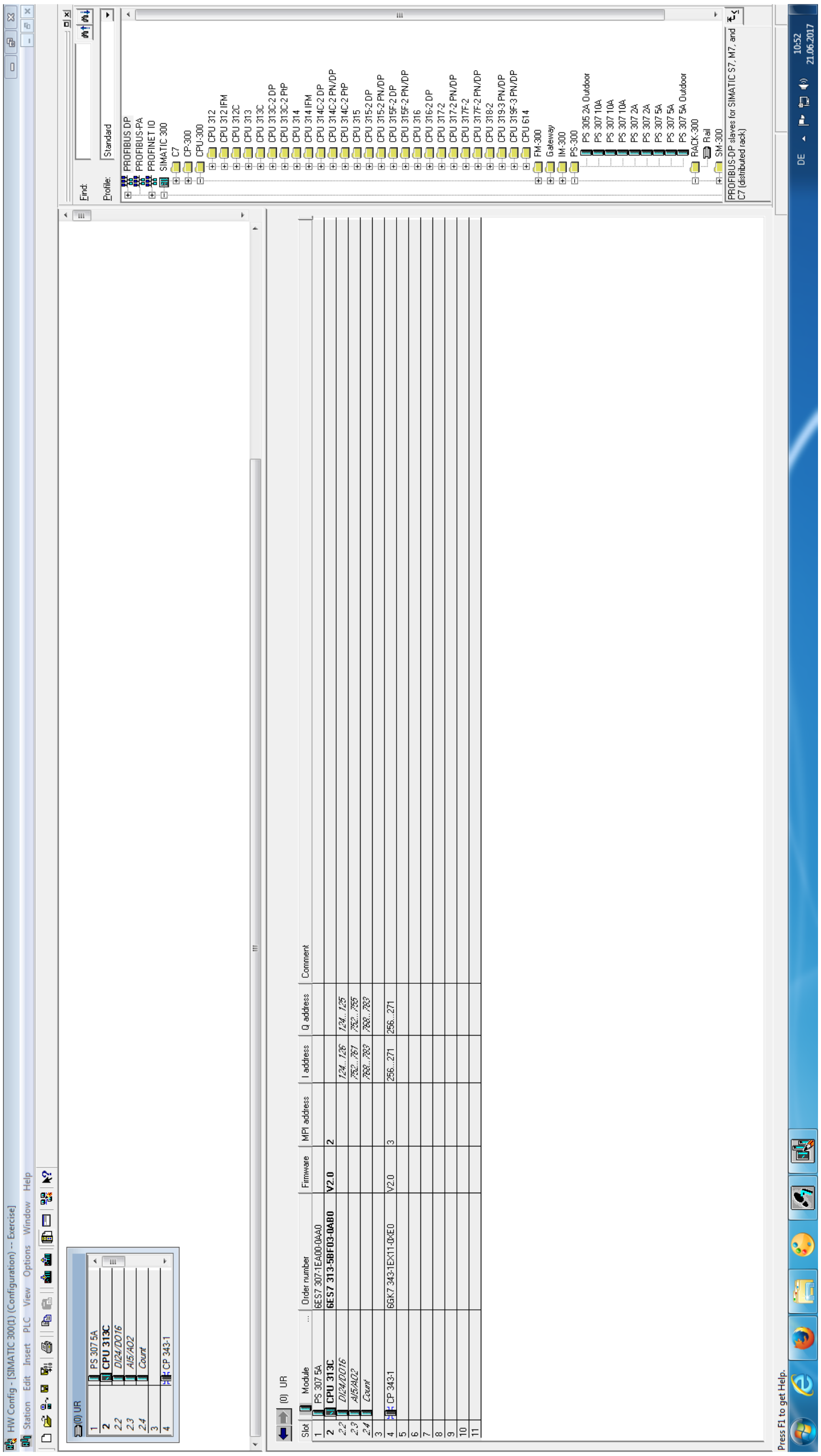
ATTACHMENT 1



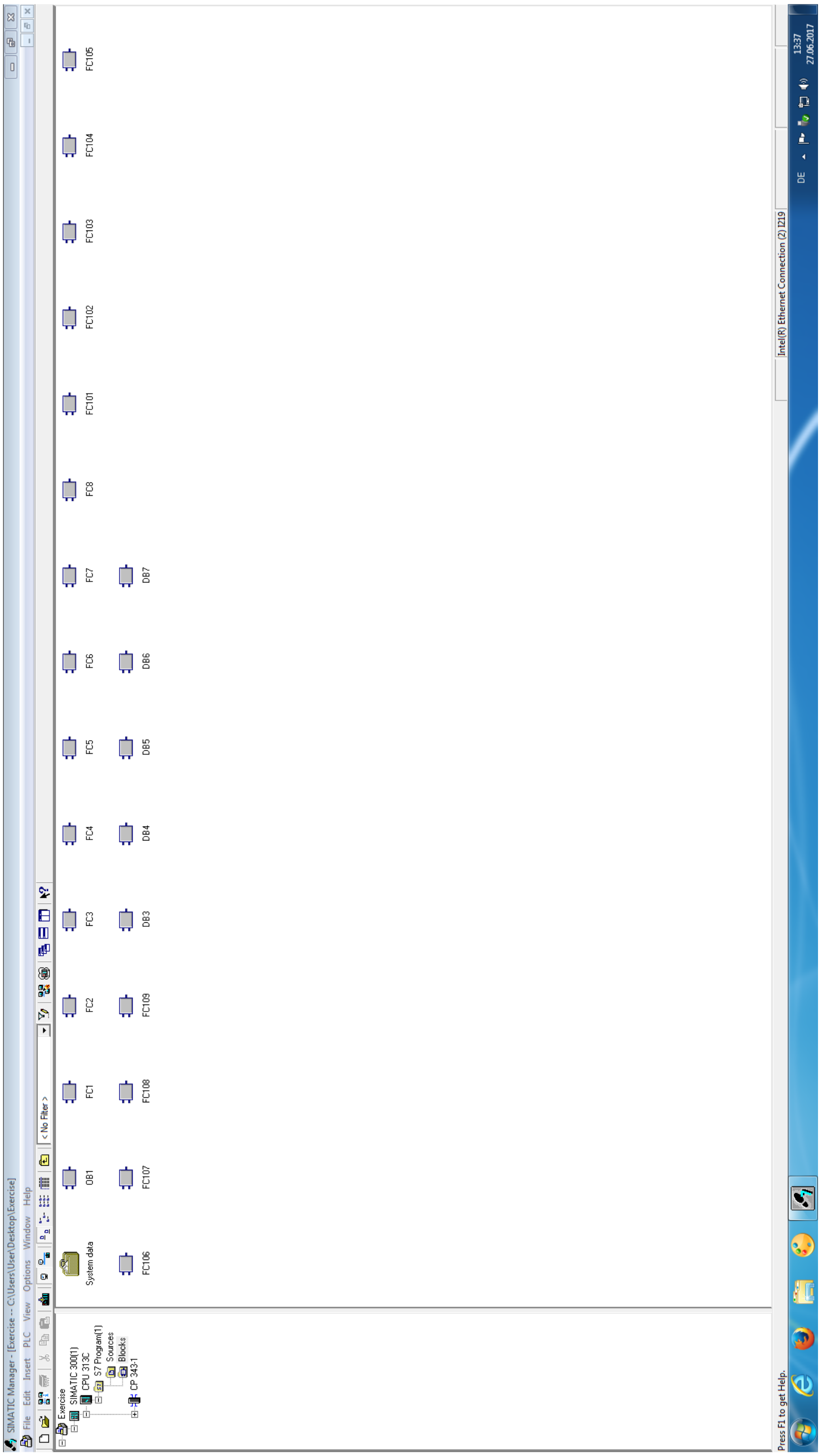
ATTACHMENT 2

Time	Interpolation																			
	1		2		3		4		5		6		7		8		9		10	
t/s	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**	**Axis (X)**	**Axis (Y)**
0	0	0	417,7	96	526,3	192	602,5	288	663,1	384	714,3	480	759,1	576	799,1	672	835,5	768	868,9	864
10	41,77	9,6	428,56	105,6	533,92	201,6	608,56	297,6	668,22	393,6	718,78	489,6	763,1	585,6	802,74	681,6	838,84	777,6	872,01	873,6
20	83,54	19,2	439,42	115,2	541,54	211,2	614,62	307,2	673,34	403,2	723,26	499,2	767,1	595,2	806,38	691,2	842,18	787,2	875,12	883,2
30	125,31	28,8	450,28	124,8	549,16	220,8	620,68	316,8	678,46	412,8	727,74	508,8	771,1	604,8	810,02	700,8	845,52	796,8	878,23	892,8
40	167,08	38,4	461,14	134,4	556,78	230,4	626,74	326,4	683,58	422,4	732,22	518,4	775,1	614,4	813,66	710,4	848,86	806,4	881,34	902,4
50	208,85	48	472	144	564,4	240	632,8	336	688,7	432	736,7	528	779,1	624	817,3	720	852,2	816	884,45	912
60	250,62	57,6	482,86	153,6	572,02	249,6	638,86	345,6	693,82	441,6	741,18	537,6	783,1	633,6	820,94	729,6	855,54	825,6	887,56	921,6
70	292,39	67,2	493,72	163,2	579,64	259,2	644,92	355,2	698,94	451,2	745,66	547,2	787,1	643,2	824,58	739,2	858,88	835,2	890,67	931,2
80	334,16	76,8	504,58	172,8	587,26	268,8	650,98	364,8	704,06	460,8	750,14	556,8	791,1	652,8	828,22	748,8	862,22	844,8	893,78	940,8
90	375,93	86,4	515,44	182,4	594,88	278,4	657,04	374,4	709,18	470,4	754,62	566,4	795,1	662,4	831,86	758,4	865,56	854,4	896,89	950,4
100	417,7	96	526,3	192	602,5	288	663,1	384	714,3	480	759,1	576	799,1	672	835,5	768	868,9	864	900	960

ATTACHMENT 3



ATTACHMENT 4



ADD ON ATTACHMENT

"This page intentionally left blank"

SIMATIC 300(1)

UR - Rack (0)

Short description:	UR
Order no.:	6ES7 390-1???0-0AA0
Designation:	UR

Rack (0), Slot 1

Short description:	PS 307 5A
Order no.:	6ES7 307-1EA00-0AA0
Designation:	PS 307 5A
Width:	1
Comment:	- - -

Rack (0), Slot 2

Short description:	CPU 313C
Firmware version:	V2.0
Order no.:	6ES7 313-5BF03-0AB0
Designation:	CPU 313C
Width:	1
MPI address:	2
Highest MPI address:	31
Baud rate:	187.5 Kbps
Comment:	- - -

Rack (0), Slot 2, Interface 2

Short description:	DI24/DO16
Order no.:	- - -
Designation:	DI24/DO16
Digital channels:	24 Inputs 16 Outputs
Width:	1
Comment:	- - -

Addresses
- Inputs
  - Start: 124
  - End: 126
- Outputs
  - Start: 124
  - End: 125

Rack (0), Slot 2, Interface 3

Short description:	AI5/AO2
Order no.:	- - -
Designation:	AI5/AO2
Analog channels:	5 Inputs 2 Outputs
Width:	1
Comment:	- - -

Addresses
- Inputs
  - Start: 752
  - End: 761
- Outputs
  - Start: 752
  - End: 755

Rack (0), Slot 2, Interface 4

Short description:	Count
Order no.:	- - -
Designation:	Count

Width: 1
Comment: - - -

Addresses
Inputs
Start: 768
End: 783
Outputs
Start: 768
End: 783

Rack (0), Slot 4

Short description: CP 343-1

Infotext:
S7 CP for Industrial Ethernet ISO and TCP/IP with SEND/RECEIVE and FETCH/WRITE interface, long data, UDP, TCP, ISO, S7 communication, routing, module replacement without PG, 10/100 Mbps, initialization over LAN, IP multicast, firmware V2.
0

Order no.: 6GK7 343-1EX11-0XE0
Name: CP 343-1
Firmware version: V2.0
Comment:
- - -

Location
Station: SIMATIC 300(1)
Width: 1

MPI address: 3
Name of MPI network: - - -

Network
Network type: Ind. Ethernet
Network name: Ethernet(1)
IP address: 10.10.252.13
Subnet mask: 255.255.255.0
Router address: - - -

Addresses
Inputs
Start: 256
End: 271
System default: No
Outputs
Start: 256
End: 271
System default: No

Security
Security enabled: - - -
Access over FTPS: - - -
Access over HTTPS: - - -
Expanded NTP configuration: - - -

Program structure (call structure)

Block(symbol), Instance DB(symbol)	Local	Language	Location				Local
S7 Program							
OB1 (Main Engine) [maximum: 22]	[22]						[22]
FC1 (Input)	[22]	STL	NW	1	Sta	1	[0]
FC2 (Engine Operation)	[22]	STL	NW	2	Sta	1	[0]
DB7 (Power Visualization)	[22]	FBD	NW	6			[0]
FC3 (Power Operation)	[22]	STL	NW	3	Sta	1	[0]
DB3 (Power Data)	[22]	FBD	NW	1			[0]
FC4 (Speed Operation)	[22]	STL	NW	4	Sta	1	[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
FC5 (Calculation)	[22]	STL	NW	5	Sta	1	[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB5 (Result Calculation)	[22]	FBD	NW	10			[0]
FC6 (Output)	[22]	STL	NW	6	Sta	1	[0]
FC7 (Visualization of Power)	[22]	STL	NW	7	Sta	1	[0]
DB3 (Power Data)	[22]	FBD	NW	1			[0]
DB7 (Power Visualization)	[22]	FBD	NW	1			[0]
FC8 (Visualization of Speed)	[22]	STL	NW	8	Sta	1	[0]
DB6 (Speed Visualization)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
FC101 (Range 1)	[22]	STL	NW	9	Sta	1	[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC102 (Range 2)	[22]	STL	NW	10	Sta	1	[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC103 (Range 3)	[22]	STL	NW	11	Sta	1	[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC104 (Range 4)	[22]	STL	NW	12	Sta	1	[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC105 (Range 5)	[22]	STL	NW	13	Sta	1	[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC106 (Range 6)	[22]	STL	NW	14	Sta	1	[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC107 (Range 7)	[22]	STL	NW	15	Sta	1	[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC108 (Range 8)	[22]	STL	NW	16	Sta	1	[0]

Block(symbol), Instance DB(symbol)	Local	Language	Location				Local
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]
FC109 (Range 9)	[22]	STL	NW	17	Sta	1	[0]
DB5 (Result Calculation)	[22]	FBD	NW	1			[0]
DB4 (Speed Data)	[22]	FBD	NW	1			[0]
DB3 (Power Data)	[22]	FBD	NW	3			[0]

Cross-reference list

Address (symbol)	Block (symbol)	Type	Language	Location
⊟ A 124.0 (Engine Running)	FC6 (Output)	W	FBD	NW 1 /R
				NW 1 /S
⊟ A 124.1 (Alarm On)	FC6 (Output)	W	FBD	NW 2 /R
				NW 2 /S
E 124.0 (Button On)	FC1 (Input)	R	FBD	NW 1 /U
E 124.1 (Button Off)	FC1 (Input)	R	FBD	NW 2 /U
E 124.2 (Button Reset)	FC1 (Input)	R	FBD	NW 3 /U
E 124.3 (Emergency Stop)	FC1 (Input)	R	FBD	NW 4 /U
E 124.4 (Sensor Power)	FC1 (Input)	R	FBD	NW 5 /U
E 124.5 (Sensor Speed)	FC1 (Input)	R	FBD	NW 6 /U
⊟ M 0.0 (Engine Start)	FC2 (Engine Operation)	R	FBD	NW 3 /U
				NW 4 /U
		W	FBD	NW 1 /R
				NW 1 /S
	FC6 (Output)	R	FBD	NW 1 /U
⊟ M 0.1 (Stop)	FC2 (Engine Operation)	R	FBD	NW 1 /U
				NW 3 /U
				NW 4 /U
		W	FBD	NW 2 /=
	FC3 (Power Operation)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
	FC4 (Speed Operation)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
	FC6 (Output)	R	FBD	NW 1 /U
				NW 2 /U
⊟ M 0.2 (Power Active)	FC2 (Engine Operation)	R	FBD	NW 5 /U
		W	FBD	NW 3 /R
				NW 3 /S
	FC3 (Power Operation)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U

Address (symbol)	Block (symbol)	Type	Language	Location
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
	FC6 (Output)	R	FBD	NW 1 /U
⊟ M 0.3 (Speed Active)	FC2 (Engine Operation)	R	FBD	NW 5 /U
		W	FBD	NW 4 /R
				NW 4 /S
	FC4 (Speed Operation)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
	FC6 (Output)	R	FBD	NW 1 /U
M 0.4 (Engine Active)	FC2 (Engine Operation)	W	FBD	NW 5 /=
⊟ M 0.5 (Alarm)	FC5 (Calculation)	W	FBD	NW 11 /=
	FC6 (Output)	R	FBD	NW 2 /U
⊟ M 0.6 (Activated 1)	FC5 (Calculation)	W	FBD	NW 1 /=
	FC101 (Range 1)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
⊟ M 0.7 (Activated 2)	FC5 (Calculation)	W	FBD	NW 2 /=
	FC102 (Range 2)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
⊟ M 1.0 (Activated 3)	FC5 (Calculation)	W	FBD	NW 3 /=
	FC103 (Range 3)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U

Address (symbol)	Block (symbol)	Type	Language	Location
				NW 8 /U
M 1.1 (Activated 4)	FC5 (Calculation)	W	FBD	NW 4 /=
	FC104 (Range 4)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 1.2 (Activated 5)	FC5 (Calculation)	W	FBD	NW 5 /=
	FC105 (Range 5)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 1.3 (Activated 6)	FC5 (Calculation)	W	FBD	NW 6 /=
	FC106 (Range 6)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 1.4 (Activated 7)	FC5 (Calculation)	W	FBD	NW 7 /=
	FC107 (Range 7)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 1.5 (Activated 8)	FC5 (Calculation)	W	FBD	NW 8 /=
	FC108 (Range 8)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 1.6 (Activated 9)	FC5 (Calculation)	W	FBD	NW 9 /=
	FC109 (Range 9)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U

Address (symbol)	Block (symbol)	Type	Language	Location
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
M 10.0 (Engine On)	FC1 (Input)	W	FBD	NW 1 /=
	FC2 (Engine Operation)	R	FBD	NW 1 /U
M 10.1 (Engine Off)	FC1 (Input)	W	FBD	NW 2 /=
	FC2 (Engine Operation)	R	FBD	NW 2 /O
M 10.2 (Reset)	FC1 (Input)	W	FBD	NW 3 /=
	FC2 (Engine Operation)	R	FBD	NW 2 /O
M 10.3 (Error)	FC1 (Input)	W	FBD	NW 4 /=
	FC2 (Engine Operation)	R	FBD	NW 2 /O
M 10.4 (Monitoring Power)	FC1 (Input)	W	FBD	NW 5 /=
	FC2 (Engine Operation)	R	FBD	NW 3 /U
M 10.5 (Monitoring Speed)	FC1 (Input)	W	FBD	NW 6 /=
	FC2 (Engine Operation)	R	FBD	NW 4 /U
M 10.7 (Active Take Over)	FC2 (Engine Operation)	W	FBD	NW 6 /=
	FC7 (Visualization of Power)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
	FC8 (Visualization of Speed)	R	FBD	NW 1 /U
				NW 2 /U
				NW 3 /U
				NW 4 /U
				NW 5 /U
				NW 6 /U
				NW 7 /U
				NW 8 /U
				NW 9 /U
				NW 10 /U
T 0 (Timer 0)	FC3 (Power Operation)	R	FBD	NW 1 /UN
				NW 2 /U
		W	FBD	NW 1 /R
				NW 1 /SS
	FC4 (Speed Operation)	R	FBD	NW 1 /UN
				NW 2 /U
		W	FBD	NW 1 /R
				NW 1 /SS
T 1 (Timer 1)	FC3 (Power Operation)	R	FBD	NW 2 /UN
				NW 3 /U
		W	FBD	NW 2 /R

Address (symbol)	Block (symbol)	Type	Language	Location
				NW 2 /SS
	FC4 (Speed Operation)	R	FBD	NW 2 /UN
				NW 3 /U
		W	FBD	NW 2 /R
				NW 2 /SS
T 2 (Timer 2)	FC3 (Power Operation)	R	FBD	NW 3 /UN
				NW 4 /U
		W	FBD	NW 3 /R
				NW 3 /SS
	FC4 (Speed Operation)	R	FBD	NW 3 /UN
				NW 4 /U
		W	FBD	NW 3 /R
				NW 3 /SS
T 3 (Timer 3)	FC3 (Power Operation)	R	FBD	NW 4 /UN
				NW 5 /U
		W	FBD	NW 4 /R
				NW 4 /SS
	FC4 (Speed Operation)	R	FBD	NW 4 /UN
				NW 5 /U
		W	FBD	NW 4 /R
				NW 4 /SS
T 4 (Timer 4)	FC3 (Power Operation)	R	FBD	NW 5 /UN
				NW 6 /U
		W	FBD	NW 5 /R
				NW 5 /SS
	FC4 (Speed Operation)	R	FBD	NW 5 /UN
				NW 6 /U
		W	FBD	NW 5 /R
				NW 5 /SS
T 5 (Timer 5)	FC3 (Power Operation)	R	FBD	NW 6 /UN
				NW 7 /U
		W	FBD	NW 6 /R
				NW 6 /SS
	FC4 (Speed Operation)	R	FBD	NW 6 /UN
				NW 7 /U
		W	FBD	NW 6 /R
				NW 6 /SS
T 6 (Timer 6)	FC3 (Power Operation)	R	FBD	NW 7 /UN
				NW 8 /U
		W	FBD	NW 7 /R
				NW 7 /SS
	FC4 (Speed Operation)	R	FBD	NW 7 /UN
				NW 8 /U
		W	FBD	NW 7 /R
				NW 7 /SS
T 7 (Timer 7)	FC3 (Power Operation)	R	FBD	NW 8 /UN
				NW 9 /U
		W	FBD	NW 8 /R
				NW 8 /SS
	FC4 (Speed Operation)	R	FBD	NW 8 /UN

Address (symbol)	Block (symbol)	Type	Language	Location
				NW 9 /U
		W	FBD	NW 8 /R
				NW 8 /SS
⊟ T 8 (Timer 8)	FC3 (Power Operation)	R	FBD	NW 9 /UN
				NW 10 /U
		W	FBD	NW 9 /R
				NW 9 /SS
	FC4 (Speed Operation)	R	FBD	NW 9 /UN
				NW 10 /U
		W	FBD	NW 9 /R
				NW 9 /SS
⊟ T 9 (Timer 9)	FC3 (Power Operation)	R	FBD	NW 10 /UN
		W	FBD	NW 10 /R
				NW 10 /SS
	FC4 (Speed Operation)	R	FBD	NW 10 /UN
		W	FBD	NW 10 /R
				NW 10 /SS

Properties of symbol table

Name:	Symbols
Author:	
Comment:	
Created on	06/07/2017 02:24:16 PM
Last modified on:	06/27/2017 01:20:09 PM
Last filter criterion:	All Symbols
Number of symbols:	64/64
Last Sorting:	Address Ascending

Status	Symbol	Address	Data type	Comment
	Engine Running	A 124.0	BOOL	
	Alarm On	A 124.1	BOOL	
	Power Data	DB 3	DB 3	This data block is use to save the reference value of power and calculation.
	Speed Data	DB 4	DB 4	This data block is use to save the reference value of speed and calculation.
	Result Calculation	DB 5	DB 5	This data block to save the value of result interpolation calculation
	Speed Visualization	DB 6	DB 6	This data block to save value for visualization of the speed.
	Power Visualization	DB 7	DB 7	This data block to save value for visualization of the power.
	Button On	E 124.0	BOOL	
	Button Off	E 124.1	BOOL	
	Button Reset	E 124.2	BOOL	
	Emergency Stop	E 124.3	BOOL	
	Sensor Power	E 124.4	BOOL	
	Sensor Speed	E 124.5	BOOL	
	Input	FC 1	FC 1	
	Engine Operation	FC 2	FC 2	
	Power Operation	FC 3	FC 3	
	Speed Operation	FC 4	FC 4	
	Calculation	FC 5	FC 5	
	Output	FC 6	FC 6	
	Visualization of Power	FC 7	FC 7	
	Visualization of Speed	FC 8	FC 8	
	Range 1	FC 101	FC 101	
	Range 2	FC 102	FC 102	
	Range 3	FC 103	FC 103	
	Range 4	FC 104	FC 104	
	Range 5	FC 105	FC 105	
	Range 6	FC 106	FC 106	
	Range 7	FC 107	FC 107	
	Range 8	FC 108	FC 108	
	Range 9	FC 109	FC 109	
	Engine Start	M 0.0	BOOL	
	Stop	M 0.1	BOOL	
	Power Active	M 0.2	BOOL	
	Speed Active	M 0.3	BOOL	
	Engine Active	M 0.4	BOOL	
	Alarm	M 0.5	BOOL	
	Activated 1	M 0.6	BOOL	
	Activated 2	M 0.7	BOOL	
	Activated 3	M 1.0	BOOL	
	Activated 4	M 1.1	BOOL	
	Activated 5	M 1.2	BOOL	
	Activated 6	M 1.3	BOOL	
	Activated 7	M 1.4	BOOL	
	Activated 8	M 1.5	BOOL	
	Activated 9	M 1.6	BOOL	
	Engine On	M 10.0	BOOL	
	Engine Off	M 10.1	BOOL	

Status	Symbol	Address	Data type	Comment
	Reset	M 10.2	BOOL	
	Error	M 10.3	BOOL	
	Monitoring Power	M 10.4	BOOL	
	Monitoring Speed	M 10.5	BOOL	
	Take Over	M 10.6	BOOL	
	Active Take Over	M 10.7	BOOL	
	Main Engine	OB 1	OB 1	
	Timer 0	T 0	TIMER	
	Timer 1	T 1	TIMER	
	Timer 2	T 2	TIMER	
	Timer 3	T 3	TIMER	
	Timer 4	T 4	TIMER	
	Timer 5	T 5	TIMER	
	Timer 6	T 6	TIMER	
	Timer 7	T 7	TIMER	
	Timer 8	T 8	TIMER	
	Timer 9	T 9	TIMER	

OB1 - <offline>

"Main Engine"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 11:34:36 AM
Interface: 02/15/1996 04:51:12 PM
Lengths (block/logic/data): 00418 00274 00022

Name	Data Type	Address	Comment
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Block: OB1 "Main Program Sweep (Cycle)"

This function contains all of main program for all of operation.

Network: 1 Call FC1

This network contains all of function about input program.

```
CALL  "Input"  FC1
NOP   0
```

Network: 2 Call FC2

This network contains all of function about engine operation program.

```
CALL  "Engine Operation"  FC2
NOP   0
```

Network: 3 Call FC3

This network contains all of function about power operation program.

```
CALL  "Power Operation"  FC3
NOP   0
```

Network: 4 Call FC4

This network contains all of function about speed operation program.

```
CALL  "Speed Operation"  FC4
NOP   0
```

Network: 5 Call FC5
This network contains all of function about calculation of range between references value of power and speed.

```
CALL  "Calculation"  FC5
NOP   0
```

Network: 6 Call FC6
This network contains all of function about output program.

```
CALL  "Output"  FC6
NOP   0
```

Network: 7 Call FC7
This network contains all of function to move the reference value of power to the visualization.

```
CALL  "Visualization of Power"  FC7
NOP   0
```

Network: 8 Call FC8
This network contains all of function to move the reference value of speed to the visualization.

```
CALL  "Visualization of Speed"  FC8
NOP   0
```

Network: 9 FC101
This network contains all of function about interpolation calculation in range 1 (Between Load 10% - Load 20%).

```
CALL  "Range 1"  FC101
NOP   0
```

Network: 10 FC102
This network contains all of function about interpolation calculation in range 2 (Between Load 20% - Load 30%).

```
CALL  "Range 2"  FC102
NOP   0
```

Network: 11 FC103
This network contains all of function about interpolation calculation in range 3 (Between Load 30% - Load 40%).

```
CALL  "Range 3"  FC103
NOP   0
```

Network: 12 FC104
This network contains all of function about interpolation calculation in range 4 (Between Load 40% - Load 50%).

```
CALL  "Range 4"  FC104
NOP   0
```

Network: 13 FC105
This network contains all of function about interpolation calculation in range 5 (Between Load 50% - Load 60%).

```
CALL  "Range 5"  FC105
NOP   0
```

Network: 14 FC106
This network contains all of function about interpolation calculation in range 6 (Between Load 60% - Load 70%).

```
CALL  "Range 6"  FC106
NOP   0
```

Network: 15 FC107
This network contains all of function about interpolation calculation in range 7 (Between Load 70% - Load 80%).

```
CALL  "Range 7"  FC107
NOP   0
```

Network: 16 FC108
This network contains all of function about interpolation calculation in range 8 (Between Load 80% - Load 90%).

```
CALL  "Range 8"  FC108
NOP   0
```

Network: 17 FC109
This network contains all of function about interpolation calculation in range 9 (Between Load 90% - Load 100%).

```
CALL  "Range 9"  FC109
NOP   0
```

FC1 - <offline>

"Input"

Name: **Family:**

Author: **Version:** 0.1

Block version: 2

Time stamp Code: 06/15/2017 01:14:16 PM

Interface: 05/23/2017 01:52:33 PM

Lengths (block/logic/data): 00128 00026 00000

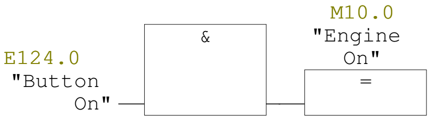
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC1 Input

This function contains all of input program.

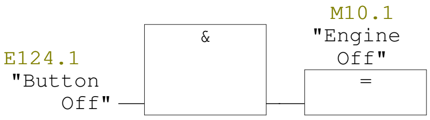
Network: 1 Engine On

This memory for activate the engine button on.



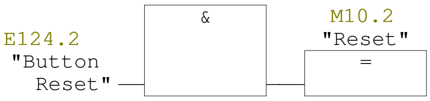
Network: 2 Engine Off

This memory for activate the engine button off.

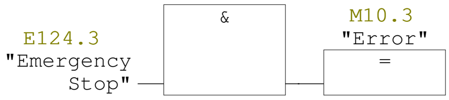


Network: 3 Reset Release

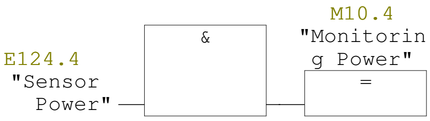
This memory for activate the engine reset button.



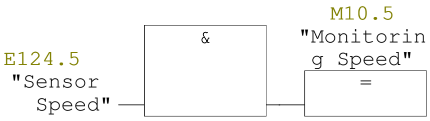
Network: 4	Error
This memory for activate the engine emergency stop button.	



Network: 5	Monitoring Power
This memory for activate the engine sensor of power.	



Network: 6	Monitoring Speed
This memory for activate the engine sensor of speed.	



FC2 - <offline>

"Engine Operation"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 11:25:53 AM
Interface: 05/23/2017 01:52:51 PM
Lengths (block/logic/data): 00160 00058 00000

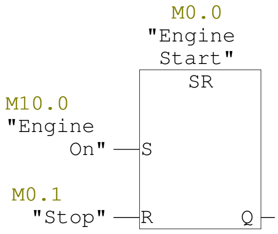
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC2 Engine Operation

This function contains all of engine operation program.

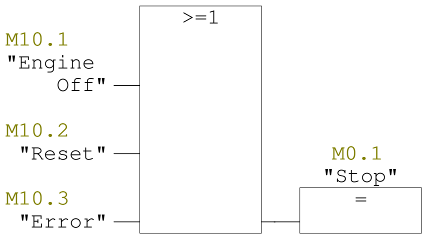
Network: 1 Engine Start

This memory for start and stop the engine.



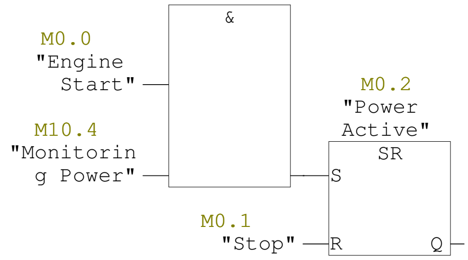
Network: 2 Stop

This memory for activate all of memory to make engine stop.



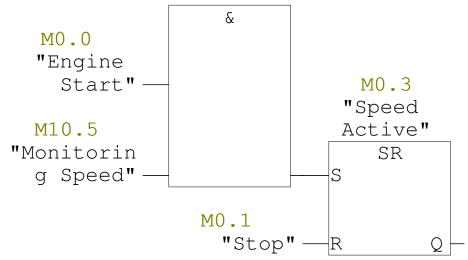
Network: 3 Power Active

This memory if the engine and sensor power is turn on, it will activate monitoring the power of engine.



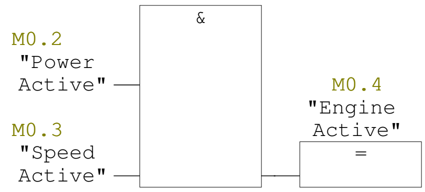
Network: 4 Speed Active

This memory if the engine and sensor speed is turn on, it will activate monitoring the speed of engine.



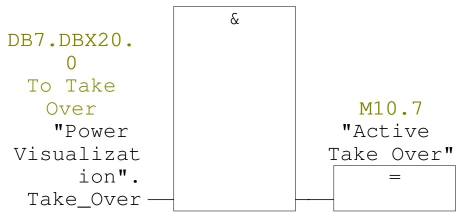
Network: 5 Engine Active

This memory if the monitoring power and speed active, the engine can start running.



Network: 6 Take Over

This memory to active the take over. This purpose to move the reference data to the visualization.



FC3 - <offline>

"Power Operation"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/15/2017 02:18:19 PM
Interface: 05/23/2017 01:53:17 PM
Lengths (block/logic/data): 00632 00500 00000

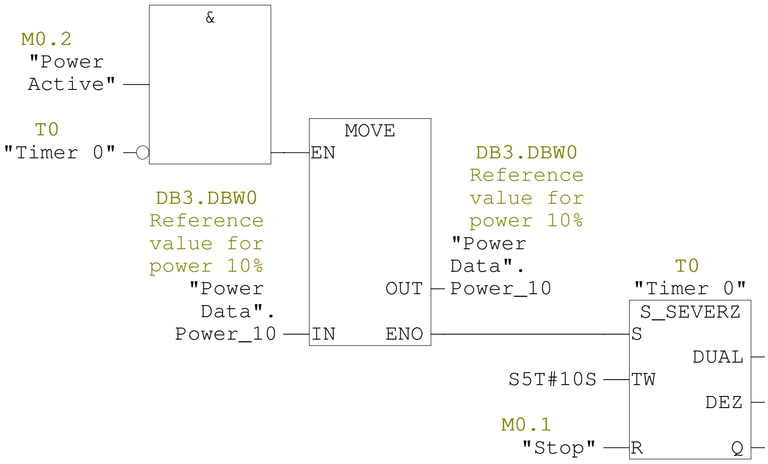
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC3 Power Operation

This function contains all of power operation program.

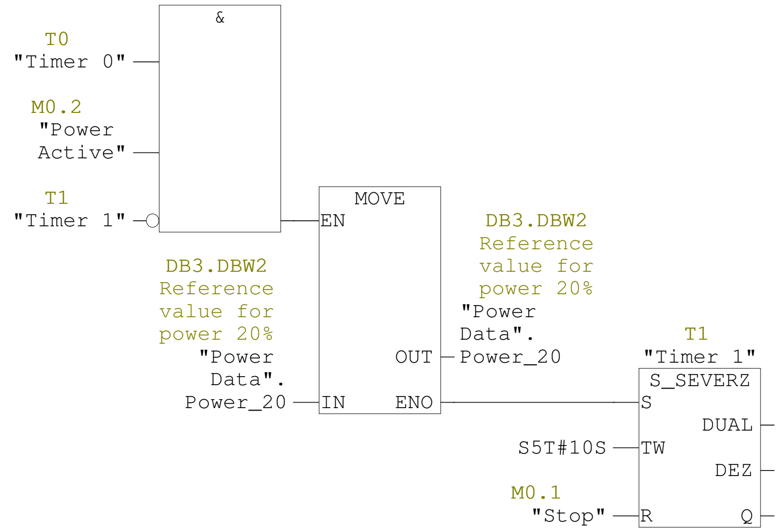
Network: 1 Load 10%

This memory for make a delay time to reach the set value of power (Load 10%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



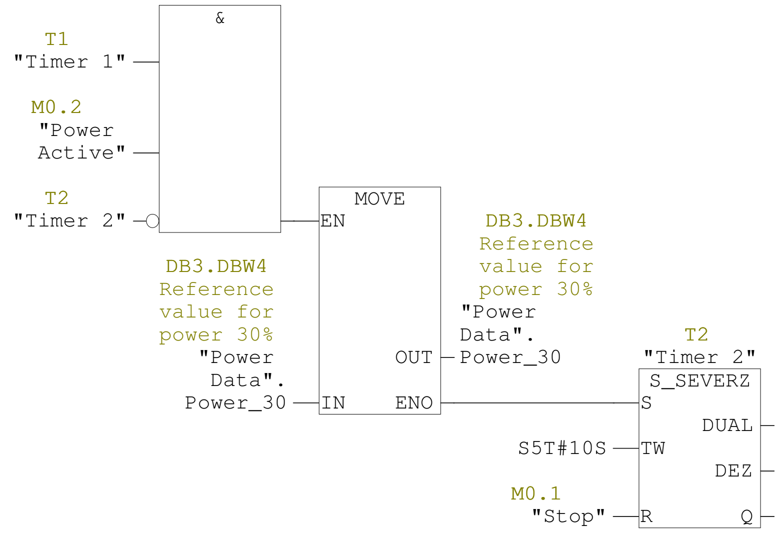
Network: 2	Load 20%

This memory for make a delay time to reach the set value of power (Load 20%).
Because to reach the power that we want have a delay time. We can change the time of delay.



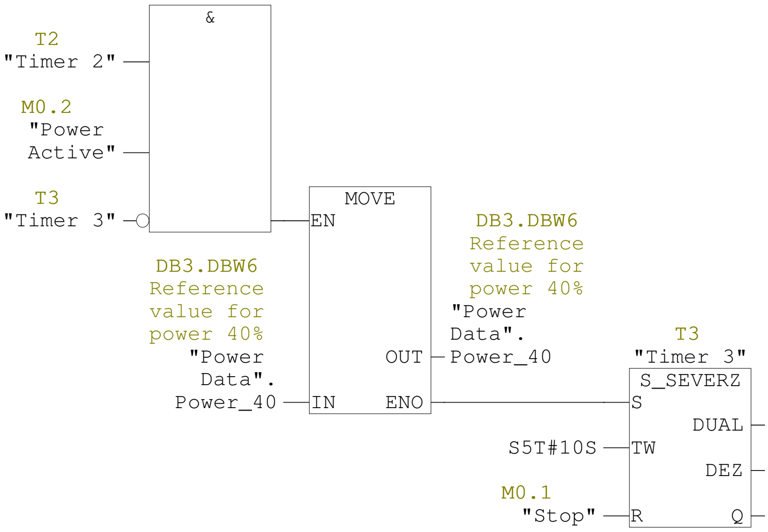
Network: 3	Load 30%

This memory for make a delay time to reach the set value of power (Load 30%).
Because to reach the power that we want have a delay time. We can change the time of delay.



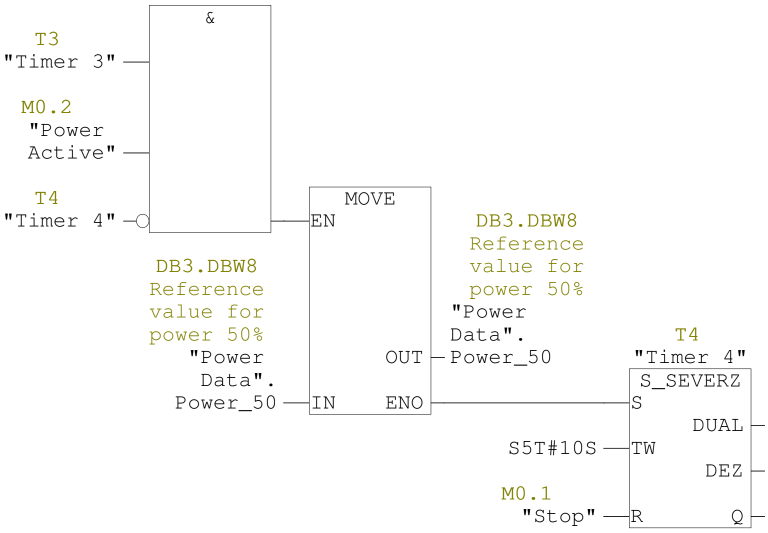
Network: 4 Load 40%

This memory for make a delay time to reach the set value of power (Load 40%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



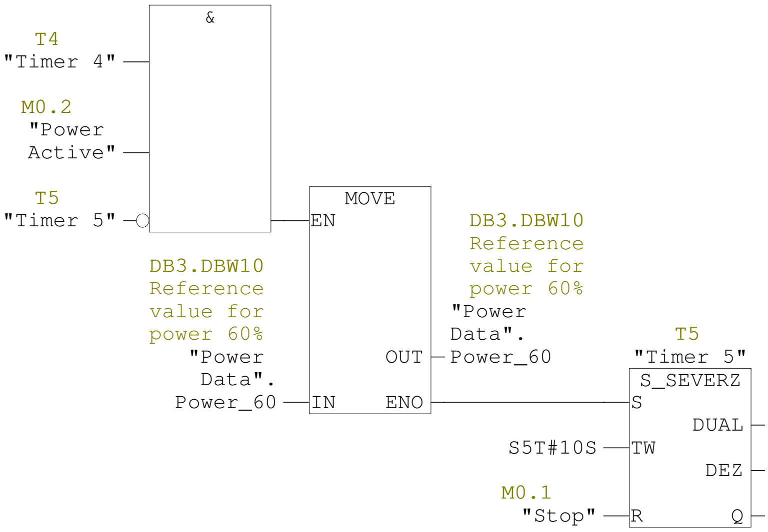
Network: 5 Load 50%

This memory for make a delay time to reach the set value of power (Load 50%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



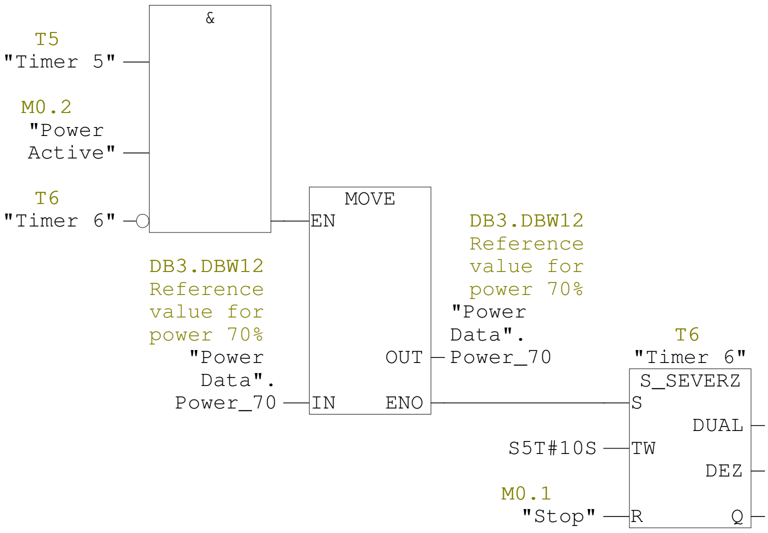
Network: 6 Load 60%

This memory for make a delay time to reach the set value of power (Load 60%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



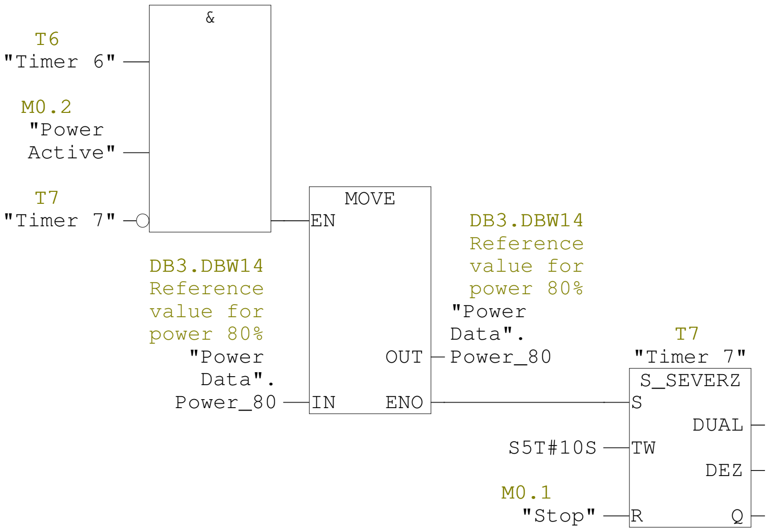
Network: 7 Load 70%

This memory for make a delay time to reach the set value of power (Load 70%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



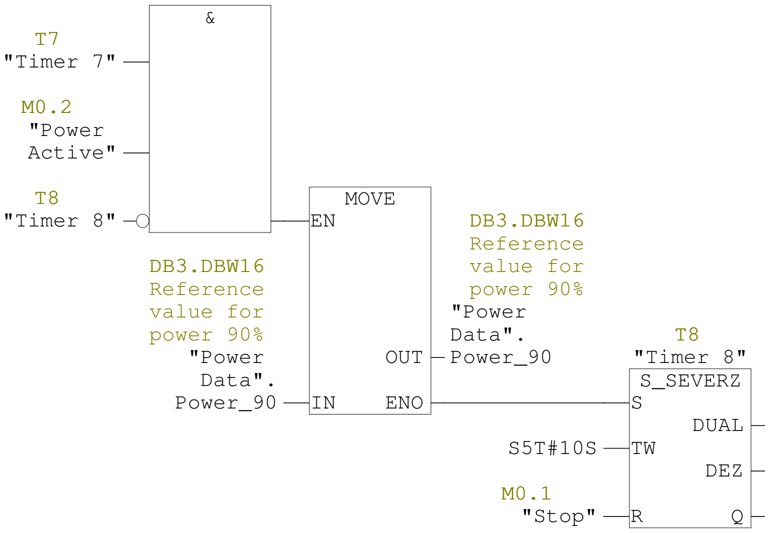
Network: 8 Load 80%

This memory for make a delay time to reach the set value of power (Load 80%). Because to reach the power that we want have a delay time. We can change the time of delay.



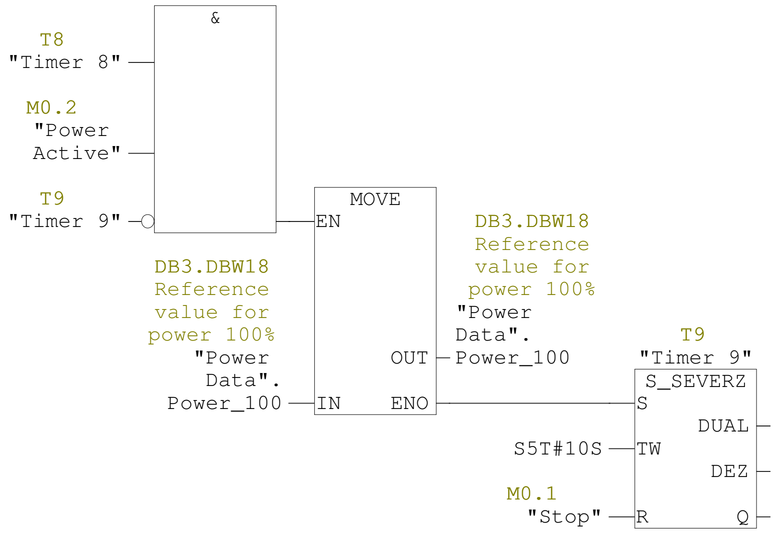
Network: 9 Load 90%

This memory for make a delay time to reach the set value of power (Load 90%). Because to reach the power that we want have a delay time. We can change the time of delay.



Network: 10 Load 100%

This memory for make a delay time to reach the set value of power (Load 100%). Because to reach the power that we want have a delay time. We can change the time of delay.



FC4 - <offline>

"Speed Operation"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/15/2017 02:17:58 PM
Interface: 05/23/2017 01:53:29 PM
Lengths (block/logic/data): 00630 00500 00000

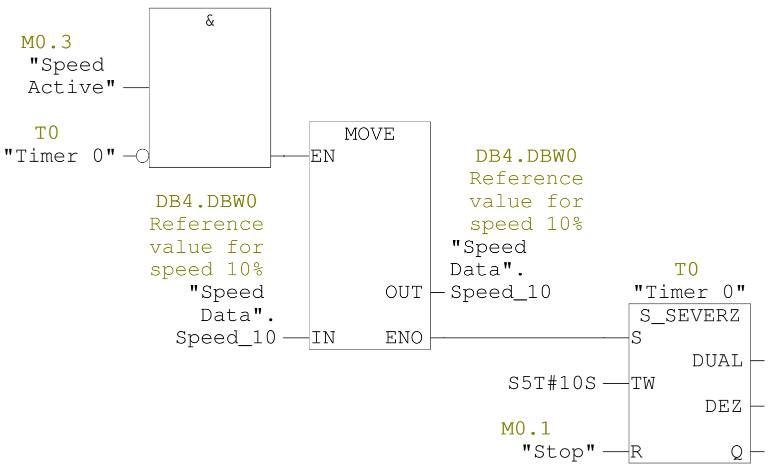
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC4 Speed Operation

This function contains all of speed operation program.

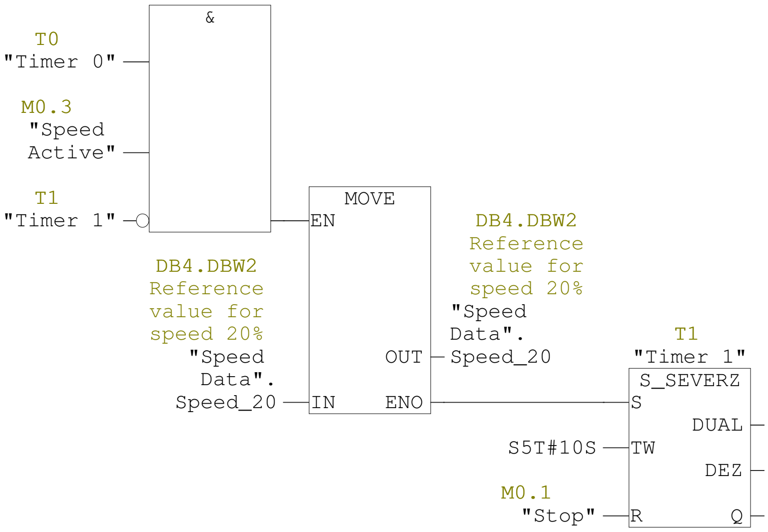
Network: 1 Load 10%

This memory for make a delay time to reach the set value of speed (Load 10%). Because to reach the power that we want have a delay time. We can change the time of delay.



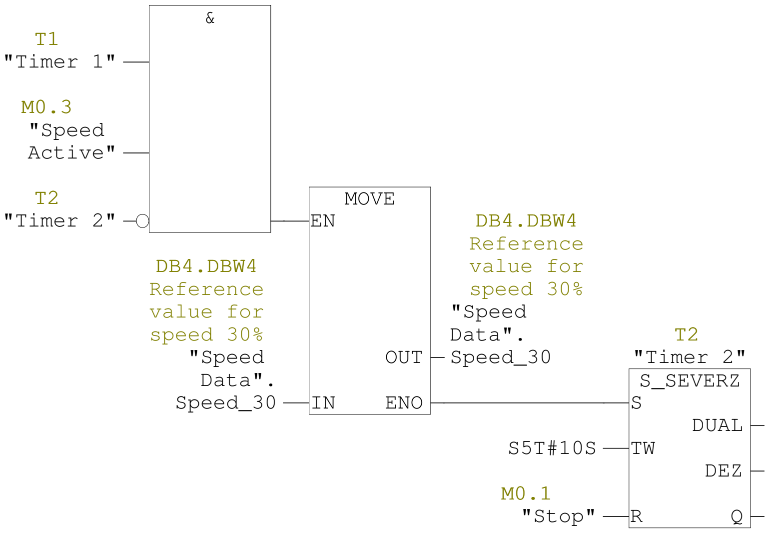
Network: 2 Load 20%

This memory for make a delay time to reach the set value of speed (Load 20%). Because to reach the power that we want have a delay time. We can change the time of delay.



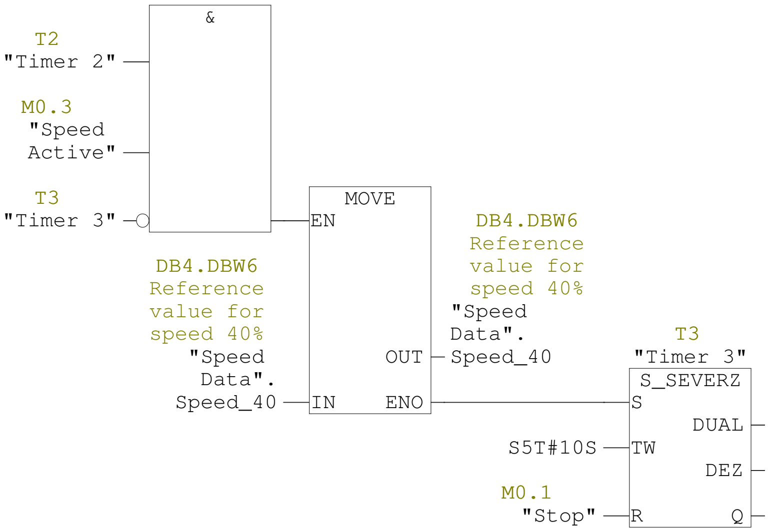
Network: 3 Load 30%

This memory for make a delay time to reach the set value of speed (Load 30%). Because to reach the power that we want have a delay time. We can change the time of delay.



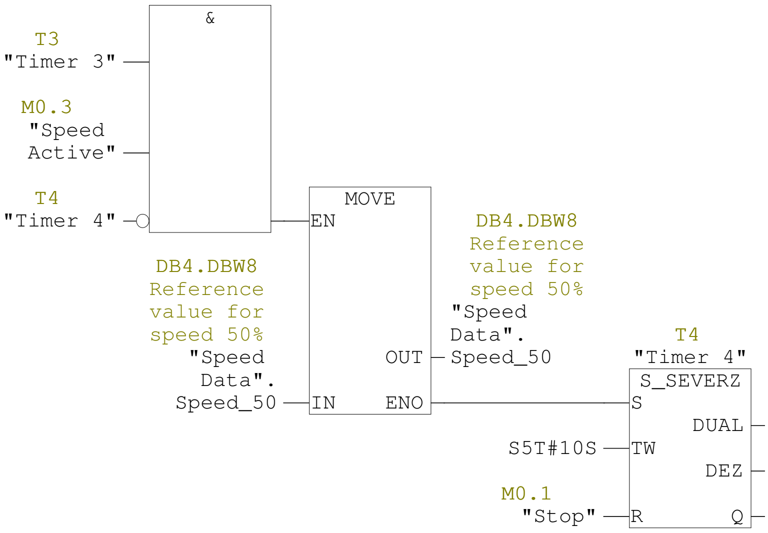
Network: 4 Load 40%

This memory for make a delay time to reach the set value of speed (Load 40%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



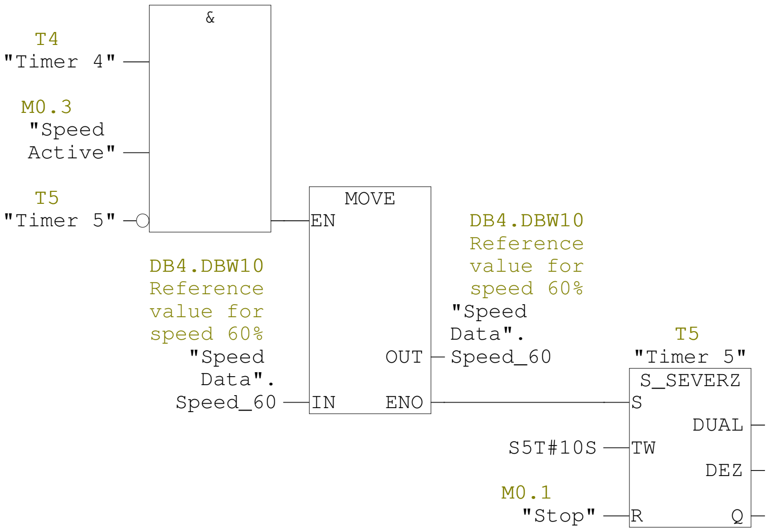
Network: 5 Load 50%

This memory for make a delay time to reach the set value of speed (Load 50%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



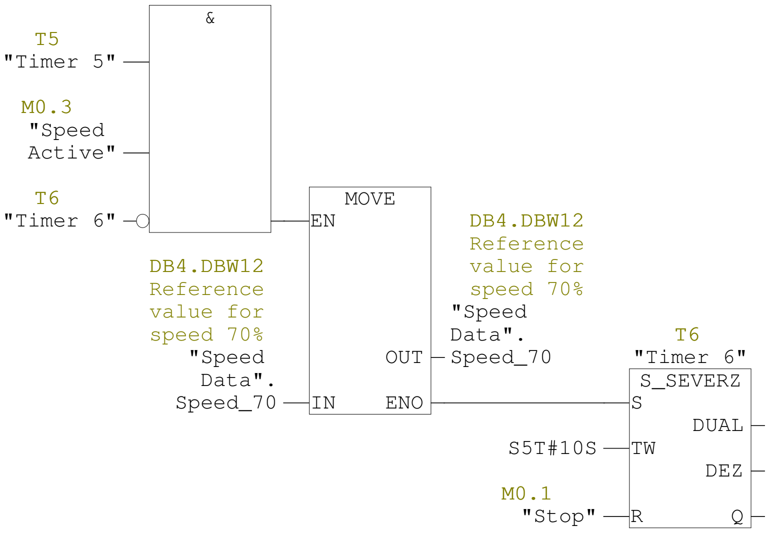
Network: 6 Load 60%

This memory for make a delay time to reach the set value of speed (Load 60%). Because to reach the power that we want have a delay time. We can change the time of delay.



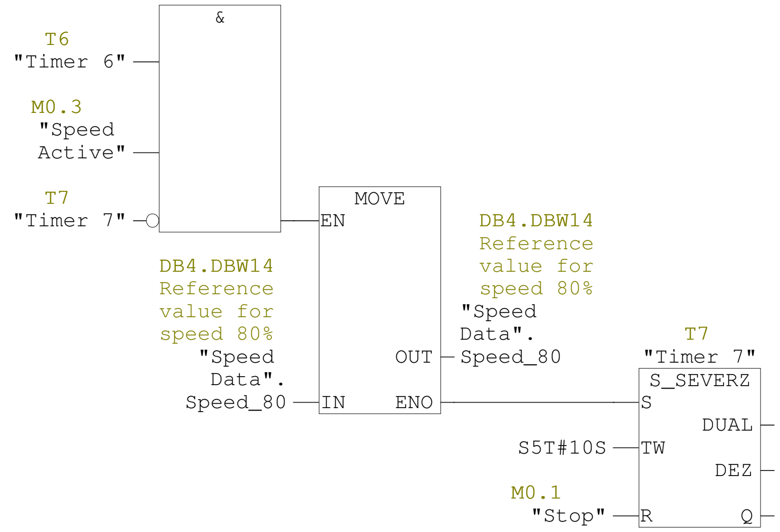
Network: 7 Load 70%

This memory for make a delay time to reach the set value of speed (Load 70%). Because to reach the power that we want have a delay time. We can change the time of delay.



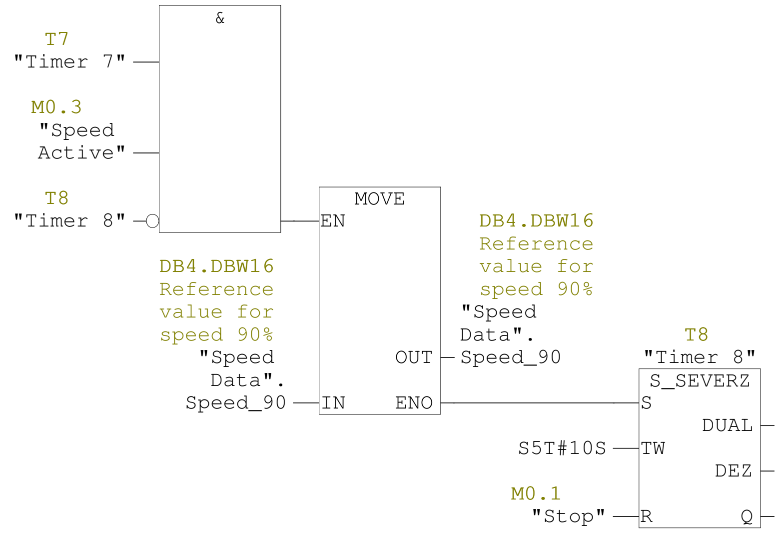
Network: 8 Load 80%

This memory for make a delay time to reach the set value of speed (Load 80%).
Because to reach the power that we want have a delay time. We can change the
time of delay.

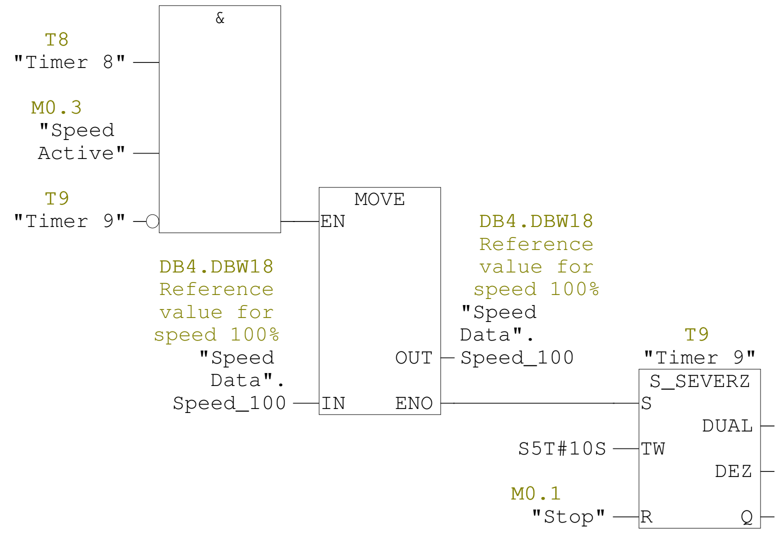


Network: 9 Load 90%

This memory for make a delay time to reach the set value of speed (Load 90%).
Because to reach the power that we want have a delay time. We can change the
time of delay.



Network: 10 Load 100%
This memory for make a delay time to reach the set value of speed (Load 100%). Because to reach the power that we want have a delay time. We can change the time of delay.



FC5 - <offline>

"Calculation"

Name: **Family:**

Author: **Version:** 0.1

Block version: 2

Time stamp Code: 06/15/2017 01:43:39 PM

Interface: 05/23/2017 02:26:54 PM

Lengths (block/logic/data): 00532 00404 00000

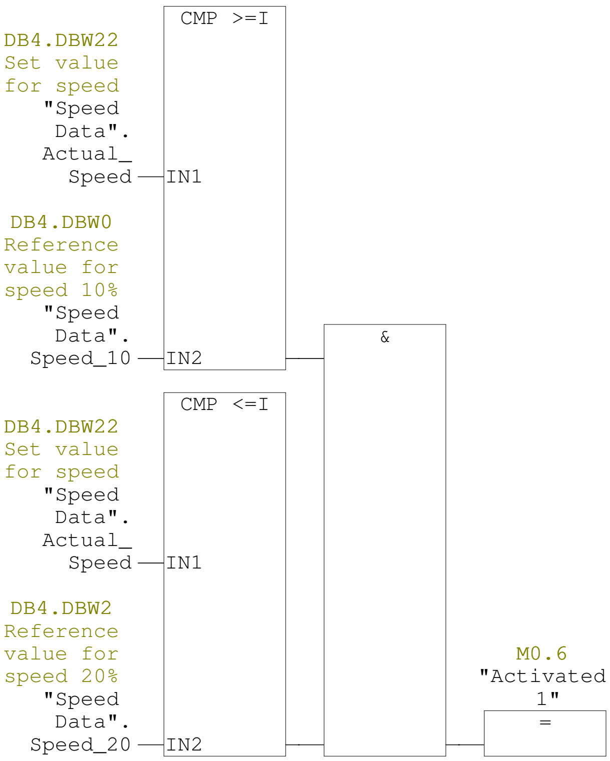
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC5 Calculation

This function contains all of program for calculation to decide the input of set value get into which range. For the example if we set the value in 450 rpm so the range 1 network will be active.

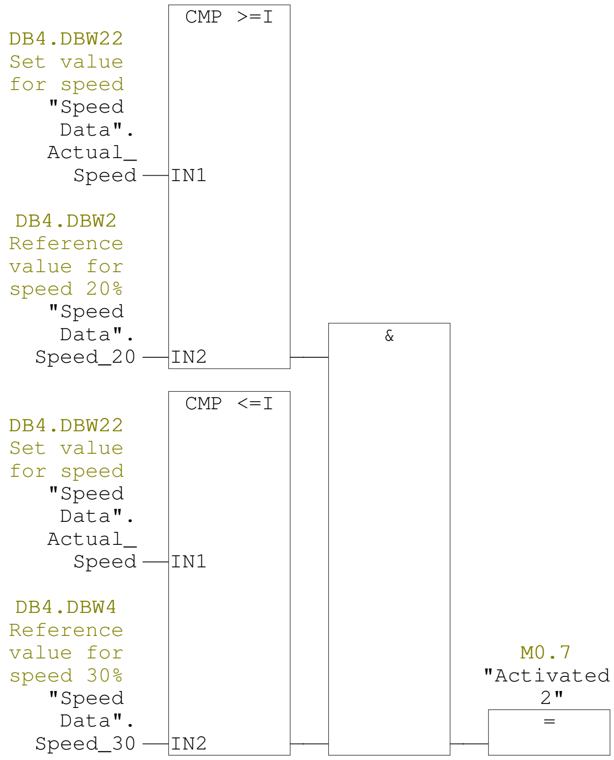
Network: 1 Range 1

This memory will be active if the set value more than 418 rpm (Load 10%) and less than 526 rpm (Load 20%).



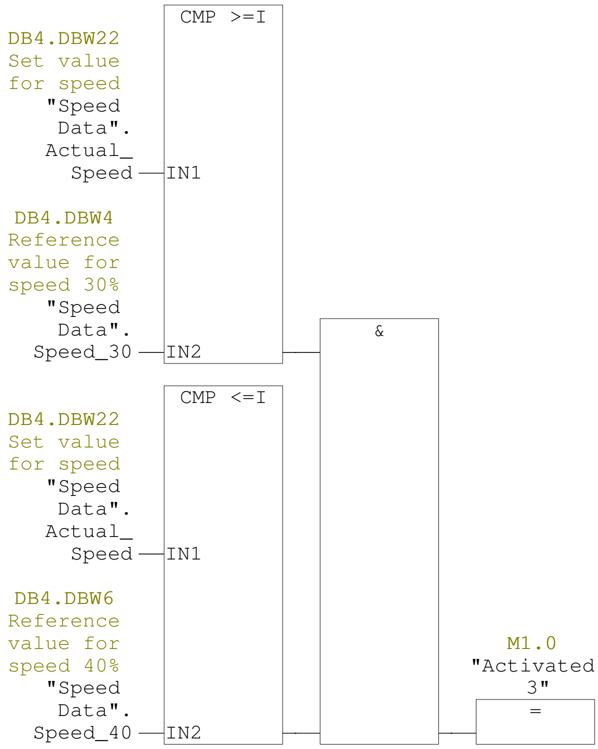
Network: 2 Range 2

This memory will be active if the set value more than 526 rpm (Load 20%) and less than 603 rpm (Load 30%).

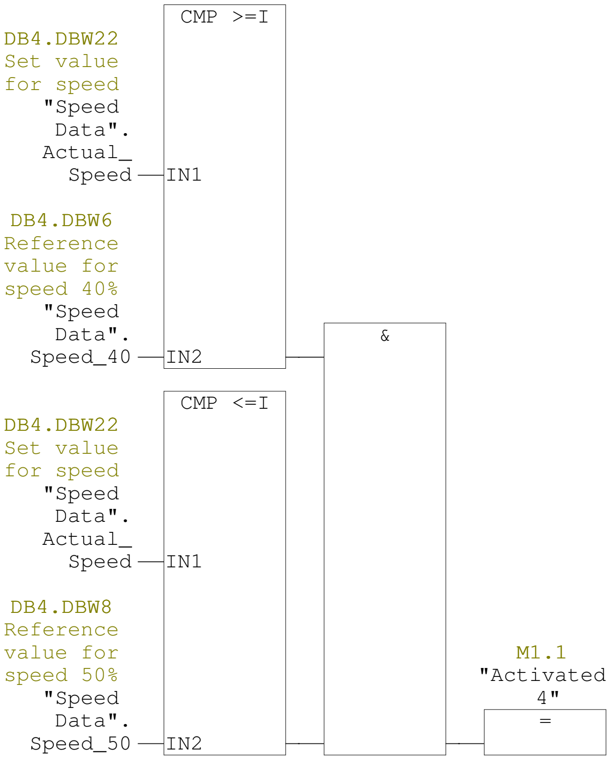


Network: 3 Range 3

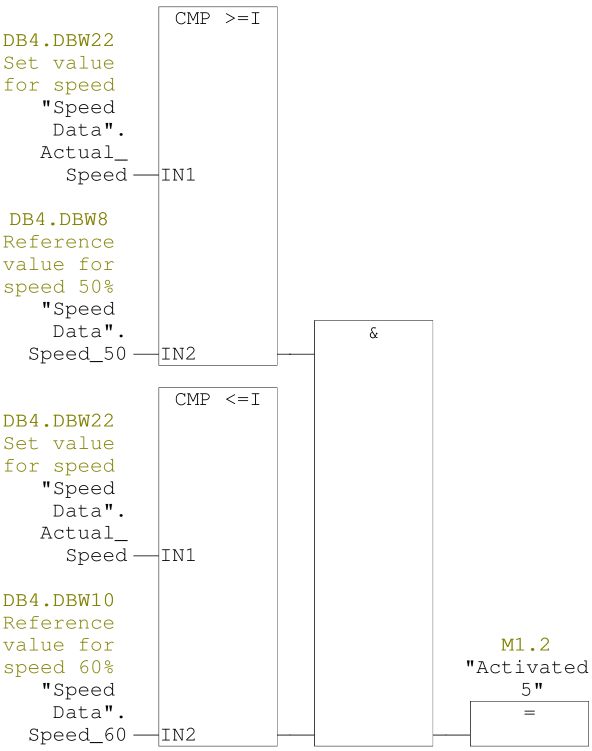
This memory will be active if the set value more than 603 rpm (Load 30%) and less than 663 rpm (Load 40%).



Network: 4 Range 4
This memory will be active if the set value more than 663 rpm (Load 40%) and less than 714 rpm (Load 50%).

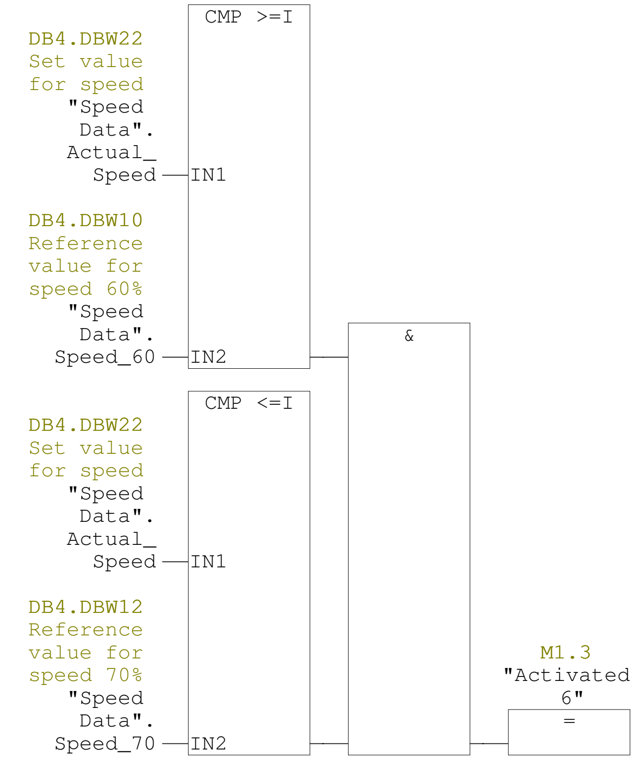


Network: 5 Range 5
This memory will be active if the set value more than 714 rpm (Load 50%) and less than 759 rpm (Load 60%).



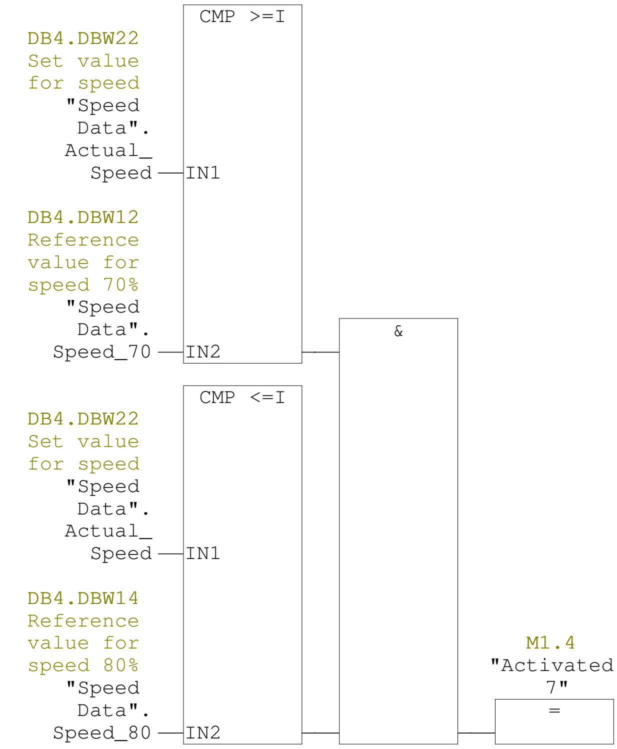
Network: 6 Range 6

This memory will be active if the set value more than 759 rpm (Load 60%) and less than 799 rpm (Load 70%).



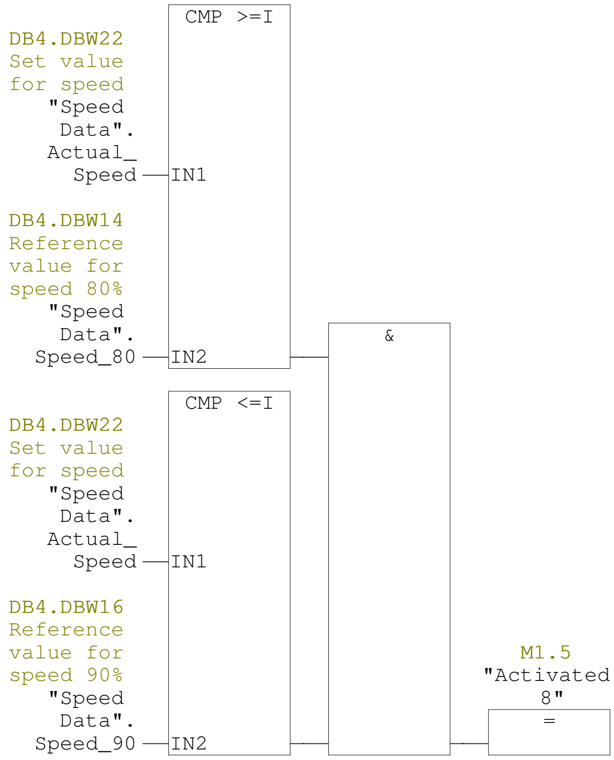
Network: 7 Range 7

This memory will be active if the set value more than 799 rpm (Load 70%) and less than 835 rpm (Load 80%).



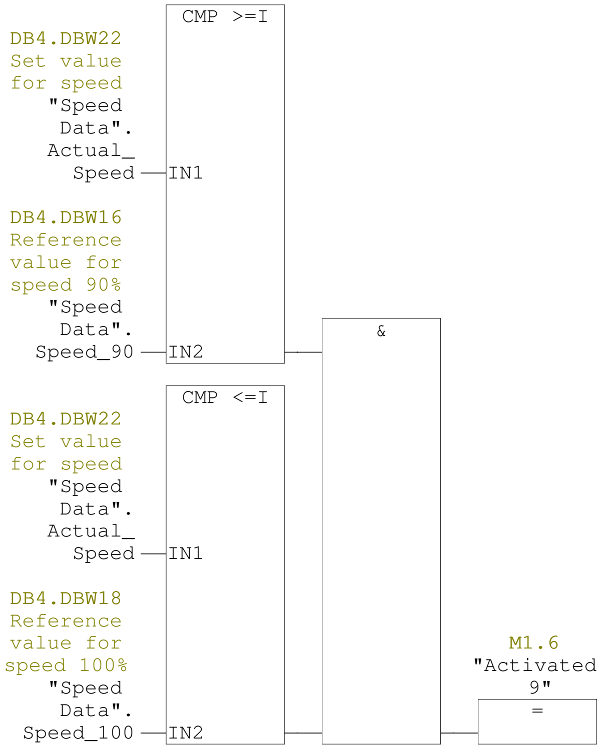
Network: 8 Range 8

This memory will be active if the set value more than 836 rpm (Load 80%) and less than 869 rpm (Load 90%).



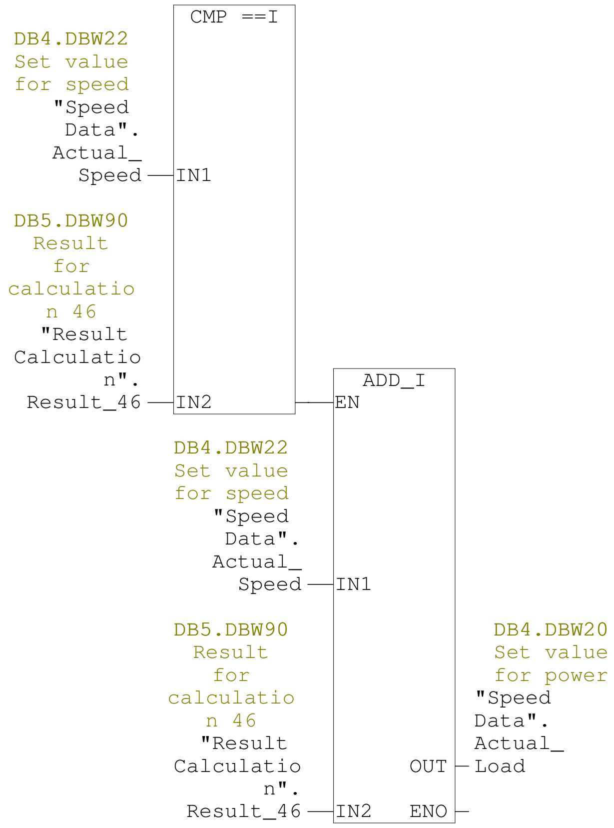
Network: 9 Range 9

This memory will be active if the set value more than 869 rpm (Load 90%) and less than 900 rpm (Load 100%).



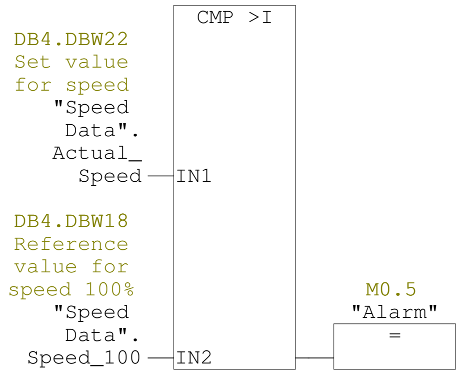
Network: 10 Range 0

This network to set the value come back to 0.



Network: 11 Alarm

This memory will be active if the set value of speed more than maximum of capacity of the engine (900 rpm)



FC6 - <offline>

"Output"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/21/2017 11:24:52 AM
Interface: 06/14/2017 08:43:39 AM
Lengths (block/logic/data): 00122 00026 00000

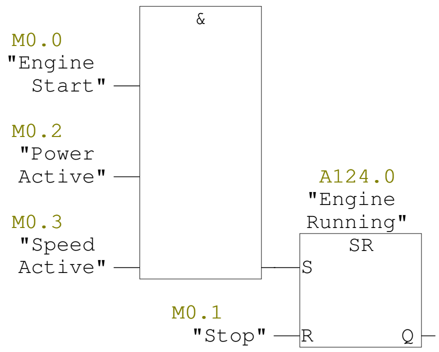
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC6 Output

This function contains all of input program.

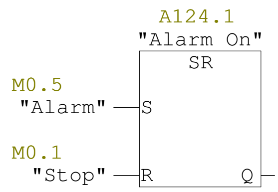
Network: 1 Engine Start

This network for start and stop the engine. The output to check all of memory works to running the engine. So the memory of activate the engine, power, and speed must be activated before. If all of memory above is active, so the engine will work correctly.



Network: 2 Alarm On

This network to make the alarm turn on. The alarm will turn on if we set the value of speed more than the maximum capacity of the engine (900 rpm). If we done to decrease the set value of speed then the alarm will turn off. As long as the alarm is on, the engine does not turn off. Because this alarm only serves to tell that there is over speed now.



FC7 - <offline>

"Visualization of Power"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 11:43:00 AM
Interface: 06/27/2017 11:07:30 AM
Lengths (block/logic/data): 00332 00202 00000

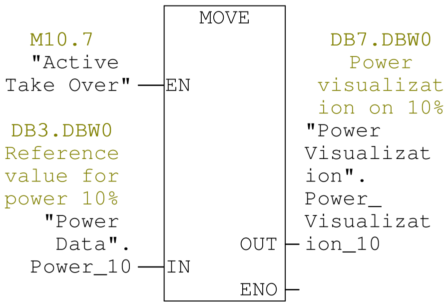
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC7 Visualization of Power

This function contains all of network to make reference value of power move to the visualization.

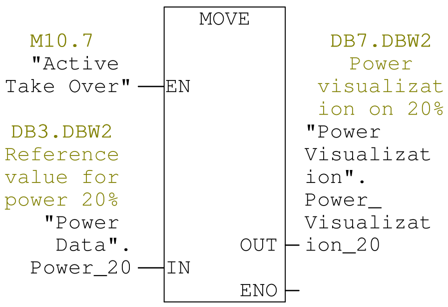
Network: 1 Visualization of Power 10%

This network to make reference value of power 10% move to the visualization.



Network: 2 Visualization of Power 20%

This network to make reference value of power 20% move to the visualization.



Network: 3 Visualization of Power 30%
This network to make reference value of power 30% move to the visualization.

```
                   MOVE
  M10.7         +---------+     DB7.DBW4
 "Active        |         |       Power
Take Over" -----|EN       |     visualizat
                |         |     ion on 30%
 DB3.DBW4       |         |    "Power
Reference       |         |    Visualizat
value for       |         |    ion".
power 30%       |         |    Power_
   "Power       |         |    Visualizat
    Data".      |      OUT|--- ion_30
  Power_30 -----|IN       |
                |      ENO|---
                +---------+
```

Network: 4 Visualization of Power 40%
This network to make reference value of power 40% move to the visualization.

```
                   MOVE
  M10.7         +---------+     DB7.DBW6
 "Active        |         |       Power
Take Over" -----|EN       |     visualizat
                |         |     ion on 40%
 DB3.DBW6       |         |    "Power
Reference       |         |    Visualizat
value for       |         |    ion".
power 40%       |         |    Power_
   "Power       |         |    Visualizat
    Data".      |      OUT|--- ion_40
  Power_40 -----|IN       |
                |      ENO|---
                +---------+
```

Network: 5 Visualization of Power 50%
This network to make reference value of power 50% move to the visualization.

```
                   MOVE
  M10.7         +---------+     DB7.DBW8
 "Active        |         |       Power
Take Over" -----|EN       |     visualizat
                |         |     ion on 50%
 DB3.DBW8       |         |    "Power
Reference       |         |    Visualizat
value for       |         |    ion".
power 50%       |         |    Power_
   "Power       |         |    Visualizat
    Data".      |      OUT|--- ion_50
  Power_50 -----|IN       |
                |      ENO|---
                +---------+
```

Network: 6 Visualization of Power 60%
This network to make reference value of power 60% move to the visualization.

```
                   MOVE
  M10.7         +---------+     DB7.DBW10
 "Active        |         |       Power
Take Over" -----|EN       |     visualizat
                |         |     ion on 60%
DB3.DBW10       |         |    "Power
Reference       |         |    Visualizat
value for       |         |    ion".
power 60%       |         |    Power_
   "Power       |         |    Visualizat
    Data".      |      OUT|--- ion_60
  Power_60 -----|IN       |
                |      ENO|---
                +---------+
```

Network: 7	Visualization of Power 70%

This network to make reference value of power 70% move to the visualization.

```
                 ┌──────────┐
                 │   MOVE   │
    M10.7        │          │    DB7.DBW12
   "Active       │          │      Power
  Take Over" ────┤EN        │   visualizat
                 │          │    ion on 70%
  DB3.DBW12      │          │   "Power
  Reference      │          │   Visualizat
  value for      │          │   ion".
  power 70%      │          │   Power_
     "Power      │          │   Visualizat
      Data".     │       OUT├── ion_70
    Power_70 ────┤IN        │
                 │       ENO├──
                 └──────────┘
```

Network: 8	Visualization of Power 80%

This network to make reference value of power 80% move to the visualization.

```
                 ┌──────────┐
                 │   MOVE   │
    M10.7        │          │    DB7.DBW14
   "Active       │          │      Power
  Take Over" ────┤EN        │   visualizat
                 │          │    ion on 80%
  DB3.DBW14      │          │   "Power
  Reference      │          │   Visualizat
  value for      │          │   ion".
  power 80%      │          │   Power_
     "Power      │          │   Visualizat
      Data".     │       OUT├── ion_80
    Power_80 ────┤IN        │
                 │       ENO├──
                 └──────────┘
```

Network: 9	Visualization of Power 90%

This network to make reference value of power 90% move to the visualization.

```
                 ┌──────────┐
                 │   MOVE   │
    M10.7        │          │    DB7.DBW16
   "Active       │          │      Power
  Take Over" ────┤EN        │   visualizat
                 │          │    ion on 90%
  DB3.DBW16      │          │   "Power
  Reference      │          │   Visualizat
  value for      │          │   ion".
  power 90%      │          │   Power_
     "Power      │          │   Visualizat
      Data".     │       OUT├── ion_90
    Power_90 ────┤IN        │
                 │       ENO├──
                 └──────────┘
```

Network: 10	Visualization of Power 100%

This network to make reference value of power 100% move to the visualization.

```
                 ┌──────────┐
                 │   MOVE   │
    M10.7        │          │    DB7.DBW18
   "Active       │          │      Power
  Take Over" ────┤EN        │   visualizat
                 │          │     ion on
  DB3.DBW18      │          │      100%
  Reference      │          │   "Power
  value for      │          │   Visualizat
  power 100%     │          │   ion".
     "Power      │          │   Power_
      Data".     │          │   Visualizat
   Power_100 ────┤IN     OUT├── ion_100
                 │          │
                 │       ENO├──
                 └──────────┘
```

FC8 - <offline>

"Visualization of Speed"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 11:45:09 AM
Interface: 06/27/2017 11:11:19 AM
Lengths (block/logic/data): 00332 00202 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC8 Visualization of Speed

This function contains all of network to make reference value of speed move to the visualization.

Network: 1 Visualization of Speed 10%

This network to make reference value of speed 10% move to the visualization.

```
                     MOVE
      M10.7                         DB6.DBW0
     "Active                          Speed
  Take Over" ---|EN              visualizat
                                 ion on 10%
   DB4.DBW0                      "Speed
  Reference                      Visualizat
  value for                      ion".
  speed 10%                      Speed_
     "Speed                      Visualizat
      Data".              OUT|---ion_10
    Speed_10 ---|IN
                          ENO|-
```

Network: 2 Visualization of Speed 20%

This network to make reference value of speed 20% move to the visualization.

```
                     MOVE
      M10.7                         DB6.DBW2
     "Active                          Speed
  Take Over" ---|EN              visualizat
                                 ion on 20%
   DB4.DBW2                      "Speed
  Reference                      Visualizat
  value for                      ion".
  speed 20%                      Speed_
     "Speed                      Visualizat
      Data".              OUT|---ion_20
    Speed_20 ---|IN
                          ENO|-
```

Network: 3 Visualization of Speed 30%

This network to make reference value of speed 30% move to the visualization.

```
                    MOVE
  M10.7                         DB6.DBW4
  "Active                         Speed
 Take Over" ---EN               visualizat
                                ion on 30%
 DB4.DBW4                      "Speed
 Reference                     Visualizat
 value for                     ion".
 speed 30%                     Speed_
    "Speed                     Visualizat
     Data".             OUT -- ion_30
  Speed_30 ---IN
                        ENO --
```

Network: 4 Visualization of Speed 40%

This network to make reference value of speed 40% move to the visualization.

```
                    MOVE
  M10.7                         DB6.DBW6
  "Active                         Speed
 Take Over" ---EN               visualizat
                                ion on 40%
 DB4.DBW6                      "Speed
 Reference                     Visualizat
 value for                     ion".
 speed 40%                     Speed_
    "Speed                     Visualizat
     Data".             OUT -- ion_40
  Speed_40 ---IN
                        ENO --
```

Network: 5 Visualization of Speed 50%

This network to make reference value of speed 50% move to the visualization.

```
                    MOVE
  M10.7                         DB6.DBW8
  "Active                         Speed
 Take Over" ---EN               visualizat
                                ion on 50%
 DB4.DBW8                      "Speed
 Reference                     Visualizat
 value for                     ion".
 speed 50%                     Speed_
    "Speed                     Visualizat
     Data".             OUT -- ion_50
  Speed_50 ---IN
                        ENO --
```

Network: 6 Visualization of Speed 60%

This network to make reference value of speed 60% move to the visualization.

```
                    MOVE
  M10.7                         DB6.DBW10
  "Active                         Speed
 Take Over" ---EN               visualizat
                                ion on 60%
 DB4.DBW10                     "Speed
 Reference                     Visualizat
 value for                     ion".
 speed 60%                     Speed_
    "Speed                     Visualizat
     Data".             OUT -- ion_60
  Speed_60 ---IN
                        ENO --
```

Network: 7 Visualization of Speed 70%

This network to make reference value of speed 70% move to the visualization.

```
                  MOVE
    M10.7                      DB6.DBW12
   "Active                       Speed
  Take Over" ──EN              visualizat
                               ion on 70%
  DB4.DBW12                  "Speed
  Reference                  Visualizat
  value for                  ion".
  speed 70%                  Speed_
     "Speed                  Visualizat
      Data".          OUT ── ion_70
    Speed_70 ──IN
                      ENO ──
```

Network: 8 Visualization of Speed 80%

This network to make reference value of speed 80% move to the visualization.

```
                  MOVE
    M10.7                      DB6.DBW14
   "Active                       Speed
  Take Over" ──EN              visualizat
                               ion on 80%
  DB4.DBW14                  "Speed
  Reference                  Visualizat
  value for                  ion".
  speed 80%                  Speed_
     "Speed                  Visualizat
      Data".          OUT ── ion_80
    Speed_80 ──IN
                      ENO ──
```

Network: 9 Visualization of Speed 90%

This network to make reference value of speed 90% move to the visualization.

```
                  MOVE
    M10.7                      DB6.DBW16
   "Active                       Speed
  Take Over" ──EN              visualizat
                               ion on 90%
  DB4.DBW16                  "Speed
  Reference                  Visualizat
  value for                  ion".
  speed 90%                  Speed_
     "Speed                  Visualizat
      Data".          OUT ── ion_90
    Speed_90 ──IN
                      ENO ──
```

Network: 10 Visualization of Speed 100%

This network to make reference value of speed 100% move to the visualization.

```
                  MOVE
    M10.7                      DB6.DBW18
   "Active                       Speed
  Take Over" ──EN              visualizat
                                 ion on
  DB4.DBW18                       100%
  Reference                  "Speed
  value for                  Visualizat
  speed 100%                 ion".
     "Speed                  Speed_
      Data".                 Visualizat
   Speed_100 ──IN     OUT ── ion_100

                      ENO ──
```

FC101 - <offline>

"Range 1"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:22:34 PM
Interface: 06/14/2017 10:11:52 AM
Lengths (block/logic/data): 00348 00226 00000

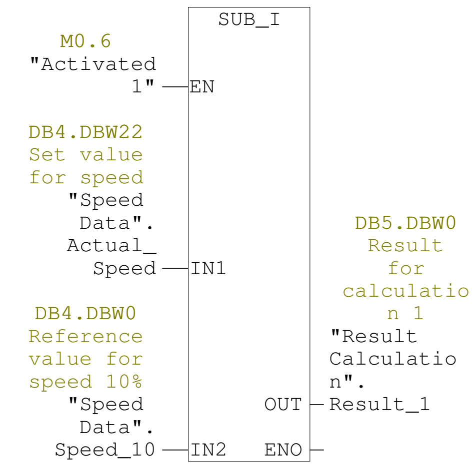
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC101 Range 1

This function contains all of interpolation calculation for range 1.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                   SUB_I
   M0.6
 "Activated
        1" ----EN

 DB4.DBW2
 Reference
 value for
 speed 20%
    "Speed                  DB5.DBW2
     Data".                  Result
  Speed_20 ----IN1            for
                          calculatio
 DB4.DBW0                     n 2
 Reference                 "Result
 value for                 Calculatio
 speed 10%                 n".
    "Speed          OUT --Result_2
     Data".
  Speed_10 ----IN2  ENO --
```

Network: 3 Y2 - Y1

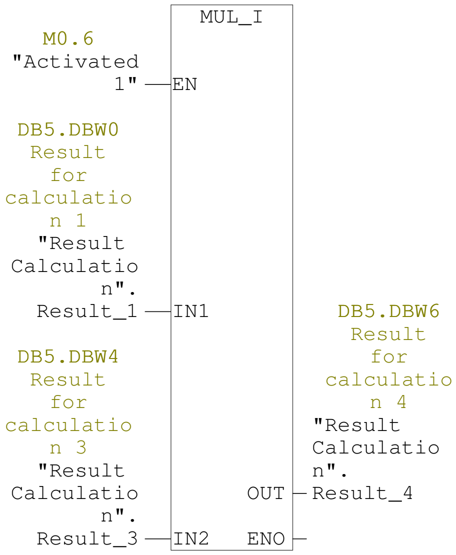
This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                   SUB_I
   M0.6
 "Activated
        1" ----EN

 DB3.DBW2
 Reference
 value for
 power 20%
    "Power                  DB5.DBW4
     Data".                  Result
  Power_20 ----IN1            for
                          calculatio
 DB3.DBW0                     n 3
 Reference                 "Result
 value for                 Calculatio
 power 10%                 n".
    "Power          OUT --Result_3
     Data".
  Power_10 ----IN2  ENO --
```

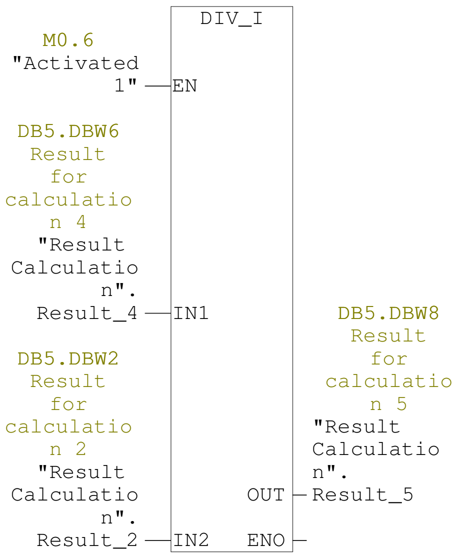
Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.



Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.



Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

ADD_I

M0.6
"Activated
1" — EN

DB3.DBW0
Reference
value for
power 10%
"Power
Data".
Power_10 — IN1

DB5.DBW8
Result
for
calculatio
n 5
"Result
Calculatio
n".
Result_5 — IN2

OUT — DB5.DBW92
Result
for
calculatio
n 47
"Result
Calculatio
n".
Result_47

ENO —

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

MUL_I

M0.6
"Activated
1" — EN

DB5.DBW92
Result
for
calculatio
n 47
"Result
Calculatio
n".
Result_47 — IN1

DB4.DBW24
This is
to set
the
percentage
of load
(torque
of water
brake)
from 0% -
100&
"Speed
Data".Load — IN2

OUT — DB5.DBW94
Result
for
calculatio
n 48
"Result
Calculatio
n".
Result_48

ENO —

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.

```
                    DIV_I
    M0.6
 "Activated
         1" ---EN

 DB5.DBW94
  Result
   for
 calculatio
   n 48
  "Result
 Calculatio
        n".
 Result_48 ---IN1

 DB4.DBW26
  This is
    to
  devided                      DB4.DBW20
  with 100                     Set value
  to make                      for power
 percentage                   "Speed
   into                       Data".
  decimal                     Actual_
   "Speed              OUT -- Load
    Data".
 Percentage ---IN2     ENO --
```

FC102 - <offline>

"Range 2"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:24:40 PM
Interface: 06/14/2017 10:12:13 AM
Lengths (block/logic/data): 00348 00226 00000

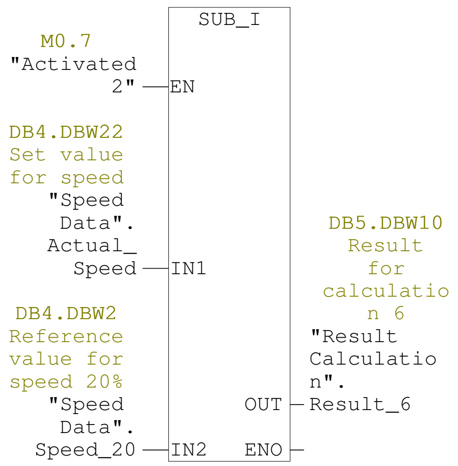
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC102 Range 2

This function contains all of interpolation calculation for range 2.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1
This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                     SUB_I
  M0.7
 "Activated
        2" ---EN

 DB4.DBW4
 Reference
 value for
 speed 30%
    "Speed                      DB5.DBW12
     Data".                       Result
  Speed_30 ---IN1                  for
                                calculatio
 DB4.DBW2                         n 7
 Reference                     "Result
 value for                     Calculatio
 speed 20%                     n".
    "Speed             OUT --- Result_7
     Data".
  Speed_20 ---IN2      ENO ---
```

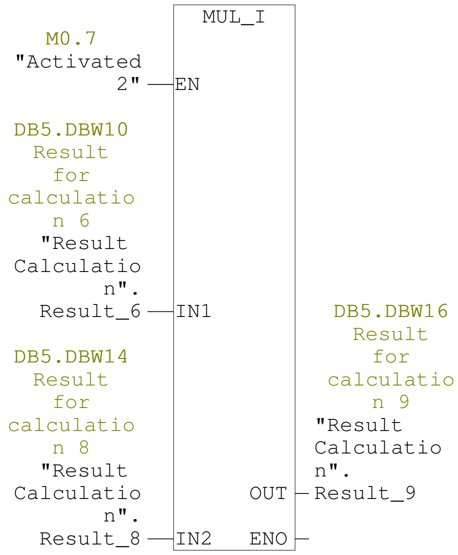
Network: 3 Y2 - Y1
This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                     SUB_I
  M0.7
 "Activated
        2" ---EN

 DB3.DBW4
 Reference
 value for
 power 30%
    "Power                      DB5.DBW14
     Data".                       Result
  Power_30 ---IN1                  for
                                calculatio
 DB3.DBW2                         n 8
 Reference                     "Result
 value for                     Calculatio
 power 20%                     n".
    "Power             OUT --- Result_8
     Data".
  Power_20 ---IN2      ENO ---
```

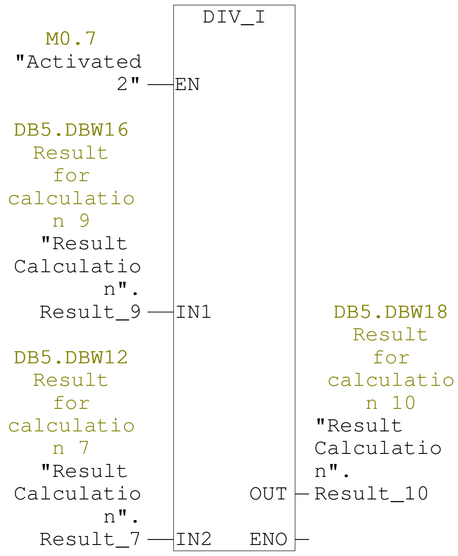
Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.



Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.



Network: 6	Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                         ADD_I
M0.7
"Activated
        2" ----EN

DB3.DBW2
Reference
value for
power 20%
"Power
Data".
Power_20 ------IN1          DB5.DBW96
                            Result
DB5.DBW18                   for
Result                      calculatio
for                         n 49
calculatio                  "Result
n 10                        Calculatio
"Result                     n".
Calculatio            OUT --Result_49
n".
Result_10 -----IN2    ENO --
```

Network: 7	Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

```
                         MUL_I
M0.7
"Activated
        2" ----EN

DB5.DBW96
Result
for
calculatio
n 49
"Result
Calculatio
n".
Result_49 -----IN1

DB4.DBW24
This is
to set                      DB5.DBW98
the                         Result
percentage                  for
of load                     calculatio
(torque                     n 50
of water                    "Result
brake)                      Calculatio
from 0% -                   n".
100&                  OUT --Result_50
"Speed
Data".Load ----IN2    ENO --
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.

DIV_I

M0.7
"Activated 2" — EN

DB5.DBW98
Result for calculation 50
"Result Calculation".Result_50 — IN1

DB4.DBW26
This is to devided with 100 to make percentage into decimal
"Speed Data".Percentage — IN2

OUT — DB4.DBW20
Set value for power
"Speed Data".Actual_Load

ENO —

FC103 - <offline>

"Range 3"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:31:46 PM
Interface: 06/14/2017 10:12:25 AM
Lengths (block/logic/data): 00348 00226 00000

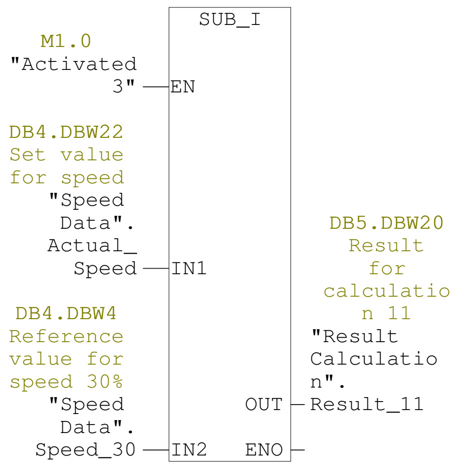
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC103 Range 3

This function contains all of interpolation calculation for range 3.

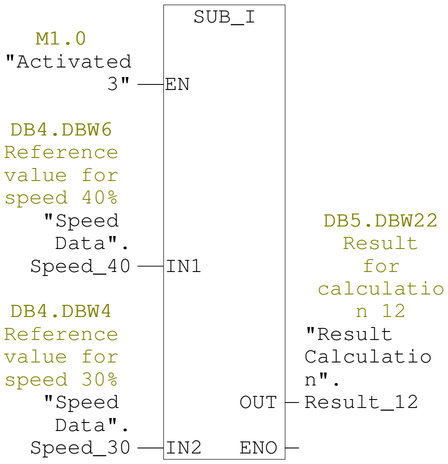
Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



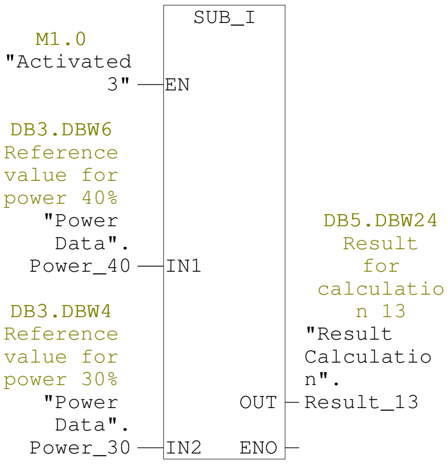
Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.



Network: 3 Y2 - Y1

This network contain the result for calculation Y2 - Y1 part of interpolation calculation.



Network: 4 (X - X1) x (Y2-Y1)
This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.

```
                  MUL_I
    M1.0
 "Activated
        3" --EN

 DB5.DBW20
   Result
    for
 calculatio
   n 11
   "Result
 Calculatio
         n".
  Result_11 --IN1           DB5.DBW26
                              Result
 DB5.DBW24                     for
   Result                  calculatio
    for                       n 14
 calculatio                "Result
   n 13                    Calculatio
   "Result                 n".
 Calculatio         OUT --Result_14
         n".
  Result_13 --IN2   ENO --
```

Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)
This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                  DIV_I
    M1.0
 "Activated
        3" --EN

 DB5.DBW26
   Result
    for
 calculatio
   n 14
   "Result
 Calculatio
         n".
  Result_14 --IN1           DB5.DBW28
                              Result
 DB5.DBW22                     for
   Result                  calculatio
    for                       n 15
 calculatio                "Result
   n 12                    Calculatio
   "Result                 n".
 Calculatio         OUT --Result_15
         n".
  Result_12 --IN2   ENO --
```

Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                        ADD_I
M1.0
"Activated
3" ------------------- EN

DB3.DBW4
Reference
value for
power 30%
"Power
Data".
Power_30 ------------- IN1          DB5.DBW100
                                    Result
DB5.DBW28                           for
Result                              calculation 51
for                                 "Result
calculation 15                      Calculation".
"Result
Calculation".                 OUT - Result_51
Result_15 ------------ IN2    ENO -
```

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

```
                        MUL_I
M1.0
"Activated
3" ------------------- EN

DB5.DBW100
Result
for
calculation 51
"Result
Calculation".
Result_51 ------------ IN1

DB4.DBW24
This is
to set                              DB5.DBW102
the                                 Result
percentage                          for
of load                             calculation 52
(torque                             "Result
of water                            Calculation".
brake)
from 0% -
100&
"Speed                        OUT - Result_52
Data".Load ----------- IN2    ENO -
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.

DIV_I

M1.0
"Activated 3" — EN

DB5.DBW102
Result for calculation 52
"Result Calculation".Result_52 — IN1

DB4.DBW26
This is to devided with 100 to make percentage into decimal
"Speed Data".Percentage — IN2

OUT — DB4.DBW20
Set value for power
"Speed Data".Actual_Load

ENO —

FC104 - <offline>

"Range 4"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:38:29 PM
Interface: 06/14/2017 10:12:46 AM
Lengths (block/logic/data): 00348 00226 00000

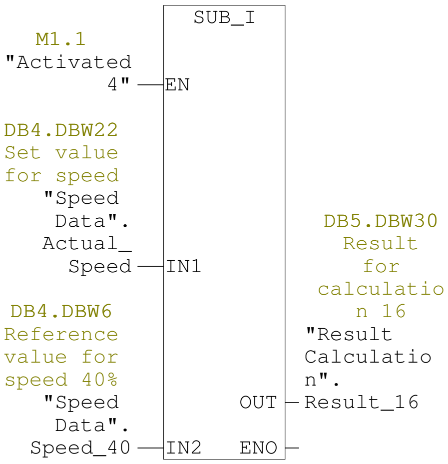
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC104 Range 4

This function contains all of interpolation calculation for range 4.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                   SUB_I
   M1.1
 "Activated
         4" ------EN

 DB4.DBW8
 Reference
 value for
 speed 50%
    "Speed                    DB5.DBW32
     Data".                    Result
  Speed_50 ------IN1            for
                             calculatio
 DB4.DBW6                       n 17
 Reference                   "Result
 value for                   Calculatio
 speed 40%                   n".
    "Speed           OUT ---Result_17
     Data".
  Speed_40 ------IN2  ENO ---
```

Network: 3 Y2 - Y1

This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                   SUB_I
   M1.1
 "Activated
         4" ------EN

 DB3.DBW8
 Reference
 value for
 power 50%
    "Power                    DB5.DBW34
     Data".                    Result
  Power_50 ------IN1            for
                             calculatio
 DB3.DBW6                       n 18
 Reference                   "Result
 value for                   Calculatio
 power 40%                   n".
    "Power           OUT ---Result_18
     Data".
  Power_40 ------IN2  ENO ---
```

Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.

MUL_I

M1.1
"Activated
4" — EN

DB5.DBW30
Result
for
calculatio
n 16
"Result
Calculatio
n".
Result_16 — IN1

DB5.DBW34
Result
for
calculatio
n 18
"Result
Calculatio
n".
Result_18 — IN2

OUT — DB5.DBW36
Result
for
calculatio
n 19
"Result
Calculatio
n".
Result_19

ENO —

Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

DIV_I

M1.1
"Activated
4" — EN

DB5.DBW36
Result
for
calculatio
n 19
"Result
Calculatio
n".
Result_19 — IN1

DB5.DBW32
Result
for
calculatio
n 17
"Result
Calculatio
n".
Result_17 — IN2

OUT — DB5.DBW38
Result
for
calculatio
n 20
"Result
Calculatio
n".
Result_20

ENO —

Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

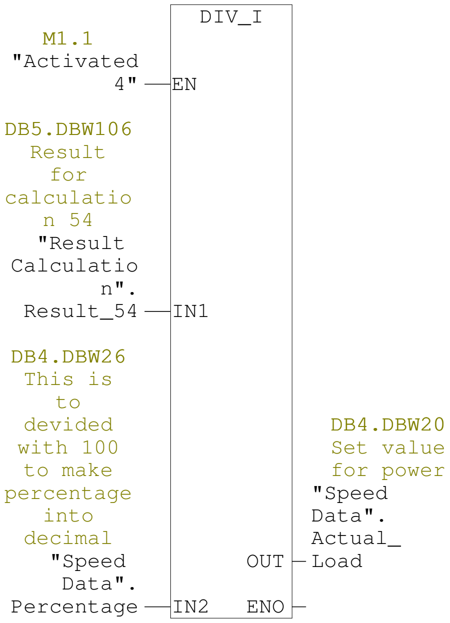
```
                   +--------------+
                   |    ADD_I     |
    M1.1           |              |
 "Activated        |              |
         4" -------|EN            |
                   |              |
  DB3.DBW6         |              |
 Reference         |              |
 value for         |              |
 power 40%         |              |
    "Power         |              |
     Data".        |              |
   Power_40 -------|IN1           |   DB5.DBW104
                   |              |     Result
 DB5.DBW38         |              |      for
   Result          |              |   calculatio
    for            |              |      n 53
 calculatio        |              |  "Result
    n 20           |              |  Calculatio
   "Result         |              |  n".
 Calculatio        |           OUT|--Result_53
        n".        |              |
  Result_20 -------|IN2        ENO|--
                   +--------------+
```

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

```
                   +--------------+
                   |    MUL_I     |
    M1.1           |              |
 "Activated        |              |
         4" -------|EN            |
                   |              |
 DB5.DBW104        |              |
   Result          |              |
    for            |              |
 calculatio        |              |
    n 53           |              |
   "Result         |              |
 Calculatio        |              |
        n".        |              |
  Result_53 -------|IN1           |
                   |              |
 DB4.DBW24         |              |
  This is          |              |
  to set           |              |   DB5.DBW106
    the            |              |     Result
 percentage        |              |      for
  of load          |              |   calculatio
  (torque          |              |      n 54
  of water         |              |  "Result
   brake)          |              |  Calculatio
 from 0% -         |              |  n".
    100&           |           OUT|--Result_54
    "Speed         |              |
 Data".Load -------|IN2        ENO|--
                   +--------------+
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.



FC105 - <offline>

"Range 5"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:40:07 PM
Interface: 06/14/2017 10:12:58 AM
Lengths (block/logic/data): 00348 00226 00000

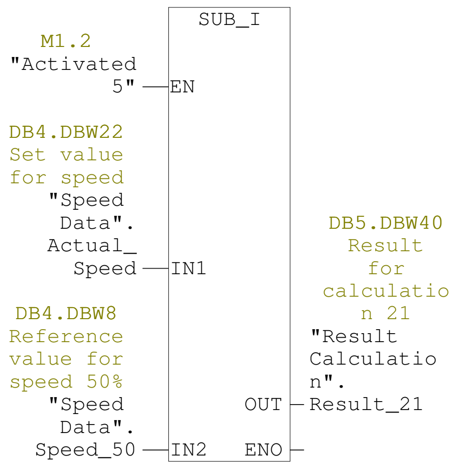
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC105 Range 5

This function contains all of interpolation calculation for range 5.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                 SUB_I
   M1.2
 "Activated
        5" ------EN

DB4.DBW10
Reference
value for
speed 60%                  DB5.DBW42
   "Speed                    Result
    Data".                    for
  Speed_60 ------IN1       calculatio
                              n 22
 DB4.DBW8                  "Result
Reference                  Calculatio
value for                  n".
speed 50%
   "Speed            OUT --Result_22
    Data".
  Speed_50 ------IN2  ENO --
```

Network: 3 Y2 - Y1

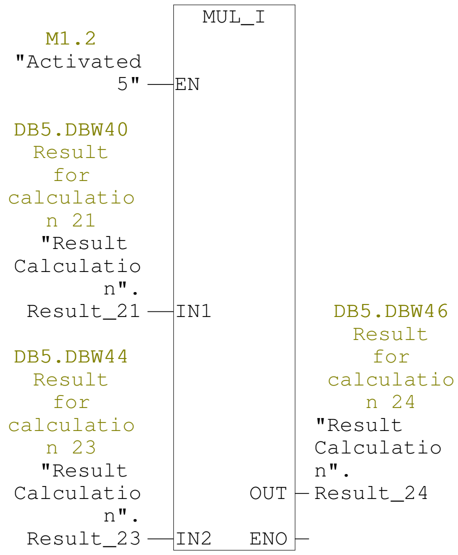
This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                 SUB_I
   M1.2
 "Activated
        5" ------EN

DB3.DBW10
Reference
value for
power 60%                  DB5.DBW44
   "Power                    Result
    Data".                    for
  Power_60 ------IN1       calculatio
                              n 23
 DB3.DBW8                  "Result
Reference                  Calculatio
value for                  n".
power 50%
   "Power            OUT --Result_23
    Data".
  Power_50 ------IN2  ENO --
```

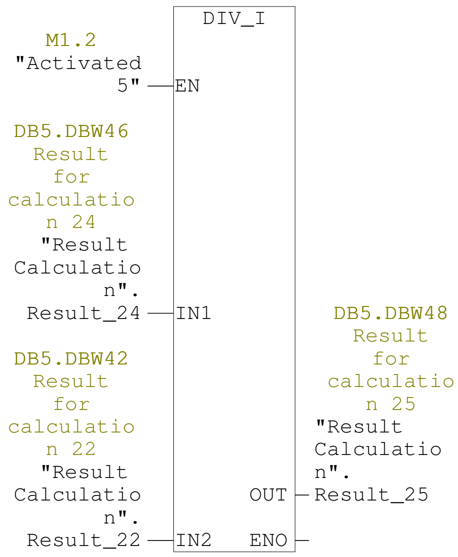
Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.



Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.



Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

ADD_I

M1.2
"Activated 5" — EN

DB3.DBW8
Reference value for power 50%
"Power Data".Power_50 — IN1

DB5.DBW48
Result for calculation 25
"Result Calculation".Result_25 — IN2

OUT — DB5.DBW108
Result for calculation 55
"Result Calculation".Result_55

ENO

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

MUL_I

M1.2
"Activated 5" — EN

DB5.DBW108
Result for calculation 55
"Result Calculation".Result_55 — IN1

DB4.DBW24
This is to set the percentage of load (torque of water brake) from 0% - 100&
"Speed Data".Load — IN2

OUT — DB5.DBW110
Result for calculation 56
"Result Calculation".Result_56

ENO

Network: 8	Percentage

This network to devided with 100 to make percentage into decimal.

DIV_I

M1.2
"Activated
5" — EN

DB5.DBW110
Result
for
calculatio
n 56
"Result
Calculatio
n".
Result_56 — IN1

DB4.DBW26
This is
to
devided
with 100
to make
percentage
into
decimal
"Speed
Data".
Percentage — IN2

OUT — DB4.DBW20
Set value
for power
"Speed
Data".
Actual_
Load

ENO —

FC106 - <offline>

"Range 6"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:42:23 PM
Interface: 06/14/2017 10:13:09 AM
Lengths (block/logic/data): 00348 00226 00000

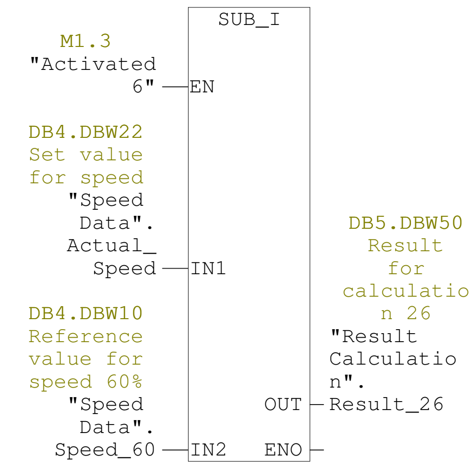
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC106 Range 6

This function contains all of interpolation calculation for range 6.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                 SUB_I
   M1.3
"Activated
      6" ----EN

DB4.DBW12
Reference
value for
speed 70%
   "Speed                    DB5.DBW52
    Data".                    Result
 Speed_70 ----IN1               for
                            calculatio
DB4.DBW10                      n 27
Reference                  "Result
value for                  Calculatio
speed 60%                  n".
   "Speed            OUT --Result_27
    Data".
 Speed_60 ----IN2    ENO --
```

Network: 3 Y2 - Y1

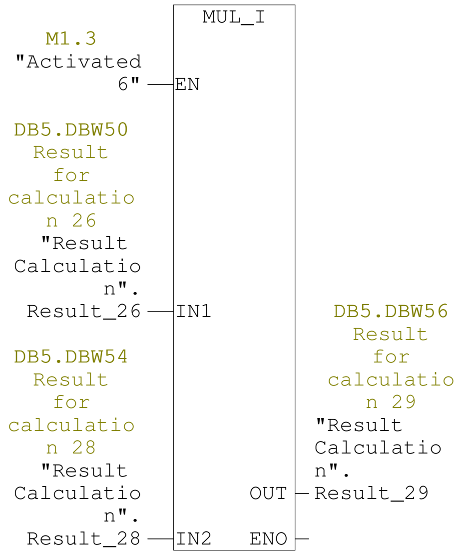
This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                 SUB_I
   M1.3
"Activated
      6" ----EN

DB3.DBW12
Reference
value for
power 70%
   "Power                    DB5.DBW54
    Data".                    Result
 Power_70 ----IN1               for
                            calculatio
DB3.DBW10                      n 28
Reference                  "Result
value for                  Calculatio
power 60%                  n".
   "Power            OUT --Result_28
    Data".
 Power_60 ----IN2    ENO --
```

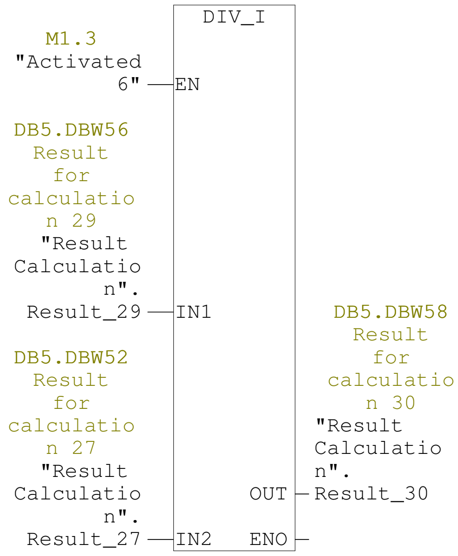
Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.



Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.



Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)
This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                  +--------------+
                  |    ADD_I     |
     M1.3         |              |
 "Activated       |              |
          6" -----|EN            |
                  |              |
   DB3.DBW10      |              |
   Reference      |              |
   value for      |              |
   power 60%      |              |
       "Power     |              |
        Data".    |              |
    Power_60 -----|IN1           |    DB5.DBW112
                  |              |      Result
   DB5.DBW58      |              |       for
    Result        |              |    calculatio
     for          |              |       n 57
  calculatio      |              |   "Result
     n 30         |              |   Calculatio
     "Result      |              |   n".
   Calculatio     |           OUT|---Result_57
          n".     |              |
   Result_30 -----|IN2       ENO |---
                  +--------------+
```

Network: 7 Load of Water Brake
This network to multiplied the result of interpolation with the value of load water brake.

```
                  +--------------+
                  |    MUL_I     |
     M1.3         |              |
 "Activated       |              |
          6" -----|EN            |
                  |              |
  DB5.DBW112      |              |
    Result        |              |
     for          |              |
  calculatio      |              |
     n 57         |              |
     "Result      |              |
   Calculatio     |              |
          n".     |              |
   Result_57 -----|IN1           |
                  |              |
   DB4.DBW24      |              |
    This is       |              |
    to set        |              |    DB5.DBW114
     the          |              |      Result
  percentage      |              |       for
   of load        |              |    calculatio
   (torque        |              |       n 58
   of water       |              |   "Result
    brake)        |              |   Calculatio
   from 0% -      |              |   n".
     100&         |           OUT|---Result_58
      "Speed      |              |
   Data".Load ----|IN2       ENO |---
                  +--------------+
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.

```
                 DIV_I
   M1.3
 "Activated
        6" --EN

 DB5.DBW114
   Result
    for
 calculatio
    n 58
   "Result
 Calculatio
        n".
  Result_58 --IN1

  DB4.DBW26
   This is
     to
   devided                    DB4.DBW20
  with 100                    Set value
   to make                    for power
 percentage                  "Speed
    into                     Data".
   decimal                   Actual_
    "Speed             OUT --Load
     Data".
 Percentage --IN2      ENO --
```

FC107 - <offline>

"Range 7"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:44:04 PM
Interface: 06/14/2017 10:13:21 AM
Lengths (block/logic/data): 00348 00226 00000

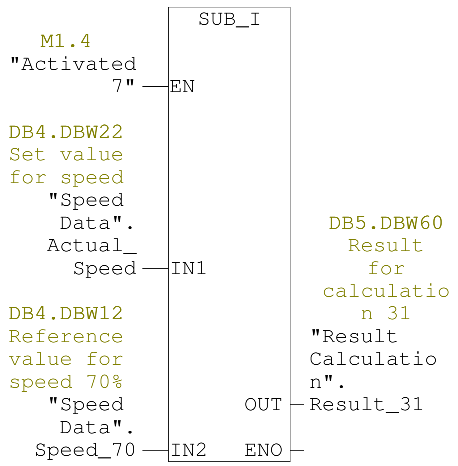
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC107 Range 7

This function contains all of interpolation calculation for range 7.

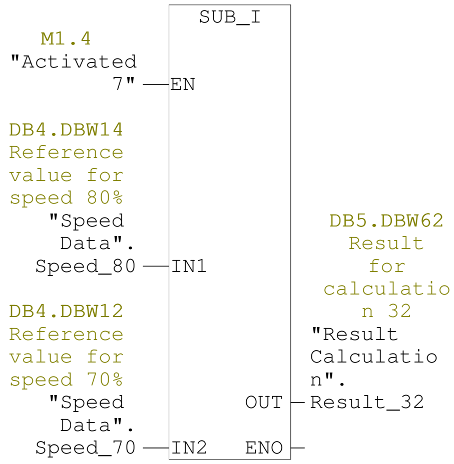
Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



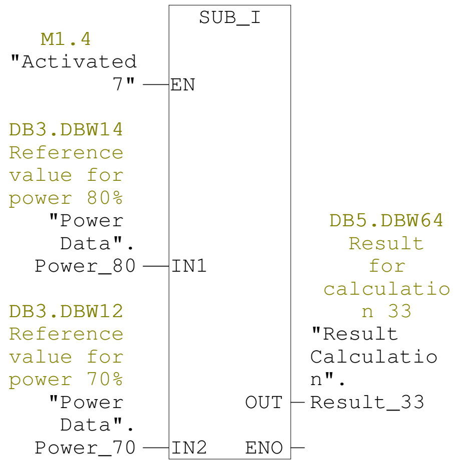
Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.



Network: 3 Y2 - Y1

This network contain the result for calculation Y2 - Y1 part of interpolation calculation.



Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.

```
                      MUL_I
    M1.4
 "Activated
         7" ---EN

 DB5.DBW60
  Result
   for
calculatio
   n 31
  "Result
 Calculatio
       n".
  Result_31 ---IN1              DB5.DBW66
                                 Result
 DB5.DBW64                        for
  Result                       calculatio
   for                            n 34
calculatio                      "Result
   n 33                         Calculatio
  "Result                       n".
 Calculatio            OUT --- Result_34
       n".
  Result_33 ---IN2     ENO ---
```

Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                      DIV_I
    M1.4
 "Activated
         7" ---EN

 DB5.DBW66
  Result
   for
calculatio
   n 34
  "Result
 Calculatio
       n".
  Result_34 ---IN1              DB5.DBW68
                                 Result
 DB5.DBW62                        for
  Result                       calculatio
   for                            n 35
calculatio                      "Result
   n 32                         Calculatio
  "Result                       n".
 Calculatio            OUT --- Result_35
       n".
  Result_32 ---IN2     ENO ---
```

Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

ADD_I

M1.4 "Activated 7" — EN

DB3.DBW12 Reference value for power 70% "Power Data".Power_70 — IN1

DB5.DBW68 Result for calculation 35 "Result Calculation".Result_35 — IN2

OUT — DB5.DBW116 Result for calculation 59 "Result Calculation".Result_59

ENO

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

MUL_I

M1.4 "Activated 7" — EN

DB5.DBW116 Result for calculation 59 "Result Calculation".Result_59 — IN1

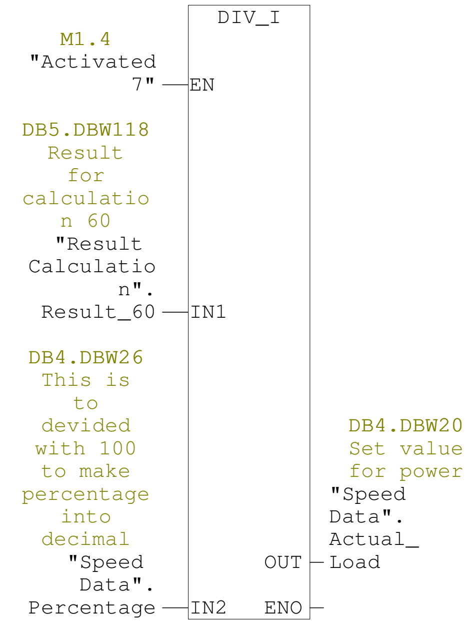
DB4.DBW24 This is to set the percentage of load (torque of water brake) from 0% - 100& "Speed Data".Load — IN2

OUT — DB5.DBW118 Result for calculation 60 "Result Calculation".Result_60

ENO

Network: 8	Percentage

This network to devided with 100 to make percentage into decimal.



FC108 - <offline>

"Range 8"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:01:32 PM
Interface: 06/14/2017 10:13:34 AM
Lengths (block/logic/data): 00348 00226 00000

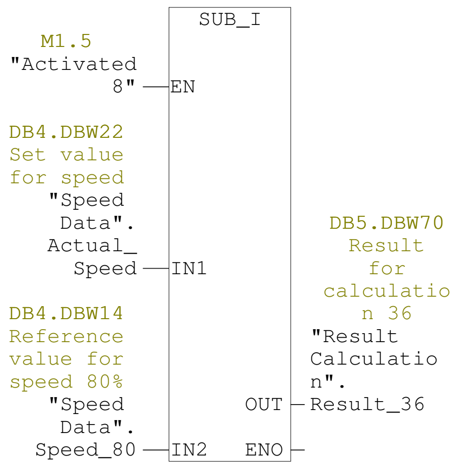
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC108 Range 8

This function contains all of interpolation calculation for range 8.

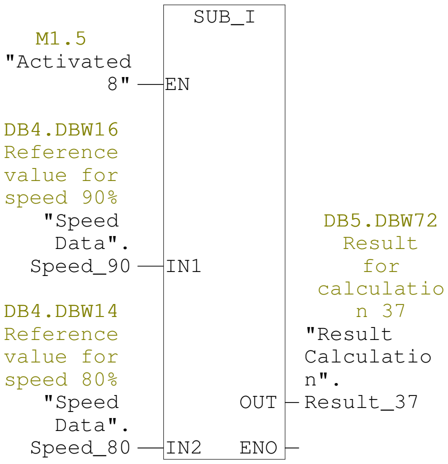
Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



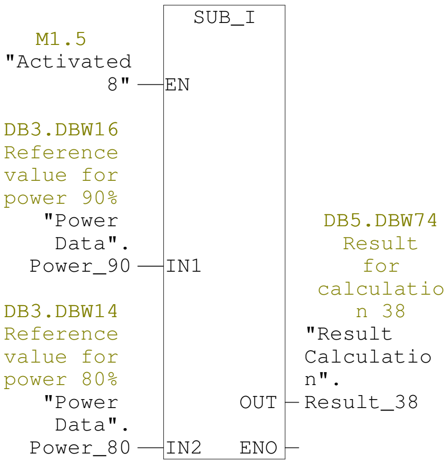
Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.



Network: 3 Y2 - Y1

This network contain the result for calculation Y2 - Y1 part of interpolation calculation.



Network: 4 (X - X1) x (Y2-Y1)
This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.

```
                 MUL_I
      M1.5
  "Activated
         8" ---EN

 DB5.DBW70
   Result
    for
 calculatio
   n 36
   "Result
  Calculatio
        n".
  Result_36 ---IN1            DB5.DBW76
                                Result
 DB5.DBW74                       for
   Result                     calculatio
    for                          n 39
 calculatio                  "Result
   n 38                      Calculatio
   "Result                   n".
  Calculatio         OUT --- Result_39
        n".
  Result_38 ---IN2   ENO ---
```

Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)
This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                 DIV_I
      M1.5
  "Activated
         8" ---EN

 DB5.DBW76
   Result
    for
 calculatio
   n 39
   "Result
  Calculatio
        n".
  Result_39 ---IN1            DB5.DBW78
                                Result
 DB5.DBW72                       for
   Result                     calculatio
    for                          n 40
 calculatio                  "Result
   n 37                      Calculatio
   "Result                   n".
  Calculatio         OUT --- Result_40
        n".
  Result_37 ---IN2   ENO ---
```

Network: 6 Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

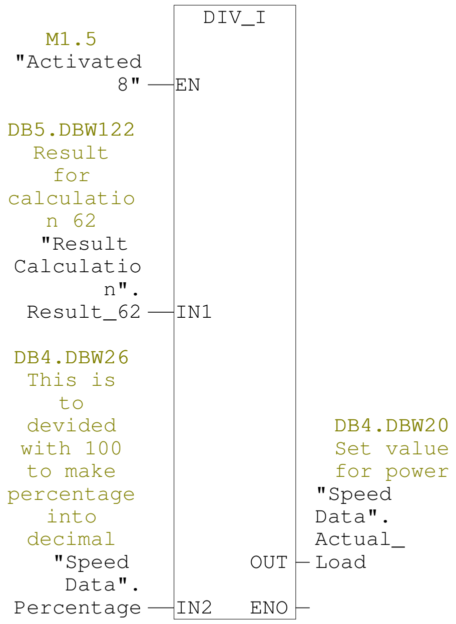
```
              +----------+
              |  ADD_I   |
   M1.5       |          |
 "Activated   |          |
        8" ---|EN        |
              |          |
 DB3.DBW14    |          |
 Reference    |          |
 value for    |          |
 power 80%    |          |
    "Power    |          |
     Data".   |          |
  Power_80 ---|IN1       |     DB5.DBW120
              |          |       Result
 DB5.DBW78    |          |        for
  Result      |          |     calculatio
   for        |          |        n 61
calculatio    |          |    "Result
  n 40        |          |    Calculatio
  "Result     |          |    n".
Calculatio    |       OUT|--- Result_61
       n".    |          |
 Result_40 ---|IN2    ENO|---
              +----------+
```

Network: 7 Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

```
              +----------+
              |  MUL_I   |
   M1.5       |          |
 "Activated   |          |
        8" ---|EN        |
              |          |
DB5.DBW120    |          |
  Result      |          |
   for        |          |
calculatio    |          |
  n 61        |          |
  "Result     |          |
Calculatio    |          |
       n".    |          |
 Result_61 ---|IN1       |
              |          |
 DB4.DBW24    |          |
  This is     |          |
  to set      |          |     DB5.DBW122
   the        |          |       Result
percentage    |          |        for
 of load      |          |     calculatio
 (torque      |          |        n 62
 of water     |          |    "Result
  brake)      |          |    Calculatio
 from 0% -    |          |    n".
   100&       |       OUT|--- Result_62
    "Speed    |          |
Data".Load ---|IN2    ENO|---
              +----------+
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.



FC109 - <offline>

"Range 9"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:05:35 PM
Interface: 06/14/2017 10:14:40 AM
Lengths (block/logic/data): 00348 00226 00000

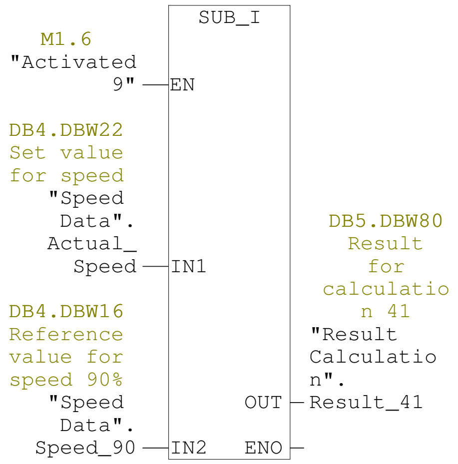
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC109 Range 9

This function contains all of interpolation calculation for range 9.

Network: 1 X - X1

This network contain the result for calculation X - X1 part of interpolation calculation.



Network: 2 X2 - X1

This network contain the result for calculation X2 - X1 part of interpolation calculation.

```
                  SUB_I
    M1.6
 "Activated
        9" ---EN

DB4.DBW18
Reference
value for
speed 100%
  "Speed
   Data".                      DB5.DBW82
 Speed_100 ---IN1               Result
                                  for
DB4.DBW16                     calculatio
Reference                        n 42
value for                     "Result
speed 90%                     Calculatio
  "Speed                      n".
   Data".           OUT --- Result_42
  Speed_90 ---IN2   ENO ---
```

Network: 3 Y2 - Y1

This network contain the result for calculation Y2 - Y1 part of interpolation calculation.

```
                  SUB_I
    M1.6
 "Activated
        9" ---EN

DB3.DBW18
Reference
value for
power 100%
  "Power
   Data".                      DB5.DBW84
 Power_100 ---IN1               Result
                                  for
DB3.DBW16                     calculatio
Reference                        n 43
value for                     "Result
power 90%                     Calculatio
  "Power                      n".
   Data".           OUT --- Result_43
  Power_90 ---IN2   ENO ---
```

Network: 4 (X - X1) x (Y2-Y1)

This network contain the result for calculation (X - X1) x (Y2-Y1) part of interpolation calculation.

MUL_I

M1.6
"Activated 9" — EN

DB5.DBW80
Result for calculation 41
"Result Calculation".
Result_41 — IN1

DB5.DBW84
Result for calculation 43
"Result Calculation".
Result_43 — IN2

OUT — DB5.DBW86
Result for calculation 44
"Result Calculation".
Result_44

ENO —

Network: 5 (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

DIV_I

M1.6
"Activated 9" — EN

DB5.DBW86
Result for calculation 44
"Result Calculation".
Result_44 — IN1

DB5.DBW82
Result for calculation 42
"Result Calculation".
Result_42 — IN2

OUT — DB5.DBW88
Result for calculation 45
"Result Calculation".
Result_45

ENO —

Network: 6	Y1 + (X - X1) x (Y2-Y1) / (X2-X1)

This network contain the result for calculation Y1 + (X - X1) x (Y2-Y1) / (X2-X1) part of interpolation calculation.

```
                         ADD_I
   M1.6
"Activated
        9" ---EN

DB3.DBW16
Reference
value for
power 90%
   "Power
    Data".
  Power_90 ---IN1             DB5.DBW124
                                Result
DB5.DBW88                        for
  Result                      calculatio
   for                           n 63
calculatio                    "Result
   n 45                       Calculatio
  "Result                     n".
Calculatio              OUT --Result_63
        n".
 Result_45 ---IN2       ENO --
```

Network: 7	Load of Water Brake

This network to multiplied the result of interpolation with the value of load water brake.

```
                         MUL_I
   M1.6
"Activated
        9" ---EN

DB5.DBW124
  Result
   for
calculatio
   n 63
  "Result
Calculatio
        n".
 Result_63 ---IN1

DB4.DBW24
 This is
  to set                      DB5.DBW126
   the                          Result
percentage                       for
 of load                      calculatio
 (torque                         n 64
 of water                     "Result
  brake)                      Calculatio
from 0% -                     n".
   100&                 OUT --Result_64
   "Speed
Data".Load ---IN2       ENO --
```

Network: 8 Percentage
This network to devided with 100 to make percentage into decimal.

DIV_I

- EN: M1.6 "Activated 9"
- IN1: DB5.DBW126 Result for calculation 64 "Result Calculation".Result_64
- IN2: DB4.DBW26 This is to devided with 100 to make percentage into decimal "Speed Data".Percentage
- OUT: DB4.DBW20 Set value for power "Speed Data".Actual_Load
- ENO

DB3 - <offline> - Declaration view

"Power Data" This data block is use to save the reference value of power and calculation.
Global data block DB 3

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:17:18 PM
Interface: 06/27/2017 10:43:03 AM
Lengths (block/logic/data): 00148 00020 00000

Block: DB3

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Power_10	INT	96	Reference value for power 10%
+2.0	Power_20	INT	192	Reference value for power 20%
+4.0	Power_30	INT	288	Reference value for power 30%
+6.0	Power_40	INT	384	Reference value for power 40%
+8.0	Power_50	INT	480	Reference value for power 50%
+10.0	Power_60	INT	576	Reference value for power 60%
+12.0	Power_70	INT	672	Reference value for power 70%
+14.0	Power_80	INT	768	Reference value for power 80%
+16.0	Power_90	INT	864	Reference value for power 90%
+18.0	Power_100	INT	960	Reference value for power 100%
=20.0		END_STRUCT		

DB4 - <offline> - Declaration view

"Speed Data" This data block is use to save the reference value of speed and calculation.
Global data block DB 4
Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:18:03 PM
Interface: 06/27/2017 12:25:51 PM
Lengths (block/logic/data): 00168 00028 00000

Block: DB4

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Speed_10	INT	418	Reference value for speed 10%
+2.0	Speed_20	INT	526	Reference value for speed 20%
+4.0	Speed_30	INT	603	Reference value for speed 30%
+6.0	Speed_40	INT	663	Reference value for speed 40%
+8.0	Speed_50	INT	714	Reference value for speed 50%
+10.0	Speed_60	INT	759	Reference value for speed 60%
+12.0	Speed_70	INT	799	Reference value for speed 70%
+14.0	Speed_80	INT	836	Reference value for speed 80%
+16.0	Speed_90	INT	869	Reference value for speed 90%
+18.0	Speed_100	INT	900	Reference value for speed 100%
+20.0	Actual_Load	INT	0	Set value for power
+22.0	Actual_Speed	INT	0	Set value for speed
+24.0	Load	INT	100	This is to set the percentage of load (torque of water brake) from 0% - 100&
+26.0	Percentage	INT	100	This is to devided with 100 to make percentage into decimal
=28.0		END_STRUCT		

DB5 - <offline> - Declaration view

"Result Calculation" This data block to save the value of result interpolation calculation
Global data block DB 5
Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 12:15:18 PM
Interface: 06/27/2017 12:15:18 PM
Lengths (block/logic/data): 00344 00128 00000

Block: DB5

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Result_1	INT	0	Result for calculation 1
+2.0	Result_2	INT	0	Result for calculation 2
+4.0	Result_3	INT	0	Result for calculation 3
+6.0	Result_4	INT	0	Result for calculation 4
+8.0	Result_5	INT	0	Result for calculation 5
+10.0	Result_6	INT	0	Result for calculation 6
+12.0	Result_7	INT	0	Result for calculation 7
+14.0	Result_8	INT	0	Result for calculation 8
+16.0	Result_9	INT	0	Result for calculation 9
+18.0	Result_10	INT	0	Result for calculation 10
+20.0	Result_11	INT	0	Result for calculation 11
+22.0	Result_12	INT	0	Result for calculation 12
+24.0	Result_13	INT	0	Result for calculation 13
+26.0	Result_14	INT	0	Result for calculation 14
+28.0	Result_15	INT	0	Result for calculation 15
+30.0	Result_16	INT	0	Result for calculation 16
+32.0	Result_17	INT	0	Result for calculation 17
+34.0	Result_18	INT	0	Result for calculation 18
+36.0	Result_19	INT	0	Result for calculation 19
+38.0	Result_20	INT	0	Result for calculation 20
+40.0	Result_21	INT	0	Result for calculation 21
+42.0	Result_22	INT	0	Result for calculation 22
+44.0	Result_23	INT	0	Result for calculation 23
+46.0	Result_24	INT	0	Result for calculation 24
+48.0	Result_25	INT	0	Result for calculation 25
+50.0	Result_26	INT	0	Result for calculation 26
+52.0	Result_27	INT	0	Result for calculation 27
+54.0	Result_28	INT	0	Result for calculation 28
+56.0	Result_29	INT	0	Result for calculation 29
+58.0	Result_30	INT	0	Result for calculation 30
+60.0	Result_31	INT	0	Result for calculation 31
+62.0	Result_32	INT	0	Result for calculation 32
+64.0	Result_33	INT	0	Result for calculation 33
+66.0	Result_34	INT	0	Result for calculation 34
+68.0	Result_35	INT	0	Result for calculation 35
+70.0	Result_36	INT	0	Result for calculation 36
+72.0	Result_37	INT	0	Result for calculation 37
+74.0	Result_38	INT	0	Result for calculation 38
+76.0	Result_39	INT	0	Result for calculation 39
+78.0	Result_40	INT	0	Result for calculation 40
+80.0	Result_41	INT	0	Result for calculation 41
+82.0	Result_42	INT	0	Result for calculation 42
+84.0	Result_43	INT	0	Result for calculation 43
+86.0	Result_44	INT	0	Result for calculation 44
+88.0	Result_45	INT	0	Result for calculation 45
+90.0	Result_46	INT	0	Result for calculation 46
+92.0	Result_47	INT	0	Result for calculation 47
+94.0	Result_48	INT	0	Result for calculation 48
+96.0	Result_49	INT	0	Result for calculation 49
+98.0	Result_50	INT	0	Result for calculation 50
+100.0	Result_51	INT	0	Result for calculation 51
+102.0	Result_52	INT	0	Result for calculation 52
+104.0	Result_53	INT	0	Result for calculation 53
+106.0	Result_54	INT	0	Result for calculation 54
+108.0	Result_55	INT	0	Result for calculation 55

Address	Name	Type	Initial value	Comment
+110.0	Result_56	INT	0	Result for calculation 56
+112.0	Result_57	INT	0	Result for calculation 57
+114.0	Result_58	INT	0	Result for calculation 58
+116.0	Result_59	INT	0	Result for calculation 59
+118.0	Result_60	INT	0	Result for calculation 60
+120.0	Result_61	INT	0	Result for calculation 61
+122.0	Result_62	INT	0	Result for calculation 62
+124.0	Result_63	INT	0	Result for calculation 63
+126.0	Result_64	INT	0	Result for calculation 64
=128.0		END_STRUCT		

DB6 - <offline> - Declaration view

"Speed Visualization" This data block to save value for visualization of the speed.
Global data block DB 6

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:19:22 PM
Interface: 06/27/2017 10:39:20 AM
Lengths (block/logic/data): 00148 00020 00000

Block: DB6

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Speed_Visualization_10	INT	418	Speed visualization on 10%
+2.0	Speed_Visualization_20	INT	526	Speed visualization on 20%
+4.0	Speed_Visualization_30	INT	603	Speed visualization on 30%
+6.0	Speed_Visualization_40	INT	663	Speed visualization on 40%
+8.0	Speed_Visualization_50	INT	714	Speed visualization on 50%
+10.0	Speed_Visualization_60	INT	759	Speed visualization on 60%
+12.0	Speed_Visualization_70	INT	799	Speed visualization on 70%
+14.0	Speed_Visualization_80	INT	836	Speed visualization on 80%
+16.0	Speed_Visualization_90	INT	869	Speed visualization on 90%
+18.0	Speed_Visualization_100	INT	900	Speed visualization on 100%
=20.0		END_STRUCT		

DB7 - <offline> - Declaration view

"Power Visualization" This data block to save value for visualization of the power.
Global data block DB 7
Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 06/27/2017 01:21:13 PM
Interface: 06/27/2017 11:18:09 AM
Lengths (block/logic/data): 00152 00022 00000

Block: DB7

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Power_Visualization_10	INT	96	Power visualization on 10%
+2.0	Power_Visualization_20	INT	192	Power visualization on 20%
+4.0	Power_Visualization_30	INT	288	Power visualization on 30%
+6.0	Power_Visualization_40	INT	384	Power visualization on 40%
+8.0	Power_Visualization_50	INT	480	Power visualization on 50%
+10.0	Power_Visualization_60	INT	576	Power visualization on 60%
+12.0	Power_Visualization_70	INT	672	Power visualization on 70%
+14.0	Power_Visualization_80	INT	768	Power visualization on 80%
+16.0	Power_Visualization_90	INT	864	Power visualization on 90%
+18.0	Power_Visualization_100	INT	960	Power visualization on 100%
+20.0	Take_Over	BOOL	FALSE	To Take Over
=22.0		END_STRUCT		