LAB University of Applied Sciences

Industrial IoT Based Belt Conveyor Systems

LAB University of Applied Sciences
Bachelor of Engineering, Mechanical Engineering and Production Technology
2021
Zakhar Afanasev

Abstract

<table>
<tr><td rowspan="2">Author(s)
Zakhar Afanasev</td><td>Publication type
Thesis, UAS</td><td>Completion year
2021</td></tr>
<tr><td>Number of pages
50</td><td></td></tr>
<tr><td colspan="3">Title of the thesis
Industrial IoT Based Belt Conveyor Systems</td></tr>
<tr><td colspan="3">Degree
Bachelor of Engineering</td></tr>
<tr><td colspan="3">Name, title and organisation of the thesis supervisor
Jouni Könönen, Senior Lecturer, LAB University of Applied Sciences</td></tr>
<tr><td colspan="3">Name, title and organisation of the client</td></tr>
<tr><td colspan="3">Abstract
The goal of this paper was to develop a conveyor belt system for early stages of Industry 4.0. The objective of the current work is to create a network at which feedback from the system could be sensed and prepare an asset in a cloud service to store and to analyse the feedback data.

The development process was based on the analysis of automation pyramid for Industry 4.0, IoT architecture, determination of common functional possibilities and technical specifications for conveyor belt systems.

The result is presented as an implemented instruction for each of the devices that create the conveyor belt system, a PLC program for operating and detailed selection process of the monitoring equipment as well as communication protocols. The documentation provides knowledge for future studies and developments of the conveyor belt systems.</td></tr>
<tr><td colspan="3">Keywords
Conveyor belt system, Industry 4.0, IoT</td></tr>
</table>

Contents

1 Introduction 1
1.1 Research motivation and objectives 1
1.2 Framework 2
2 Background information 4
2.1 Belt conveyor 4
2.2 Programmable logic controller 5
2.3 Discrete and analog signals 6
2.4 Programming terminal 7
2.5 Data communication protocols 9
2.6 AC Motor 9
2.7 AC Drive 11
2.8 Torque transmission technologies 12
2.9 Industry 4.0 13
3 Initial project hardware 14
3.1 Computing network 14
3.2 Driving hardware 15
3.3 Cloud network 16
4 Network development 18
4.1 Automation pyramid 18
4.2 IIoT network architecture 19
4.3 Project network and topology 20
4.4 Selection of monitoring equipment 22
4.4.1 Bearings inspection 23
4.4.2 Surround monitoring equipment 26
4.4.3 Productivity monitoring 28
4.5 Selection of data communication protocol 30
5 Integration process 32
5.1 Industrial drive 32
5.2 Computing unit 36
5.3 MindSphere asset configuration 46
6 Summary 48
References 49

Appendices

Appendix 1. Overview of the main PLC program

1 Introduction

Conveyor belt systems have become an essential part of a wide variety of industries, right from the dawn of industrial systems in 1905, being a key element that can transport the rising volumes of good produced from one location to another. Compared to manual handling, conveyor belts are quick and efficient as well as reducing the necessity for human attendance in the transportation process (Wikipedia 2021a).

Throughout the usage, conveyor belt systems have always been a part of the modernizing cycle to fulfil the endlessly growing interest in the higher rate of transporting goods. The transporting rate is influenced by factors such as the level of automatization and reliability of the system. With the introduction of computerized technologies and programming languages during the third industrial revolution, conveyor belt systems were able to keep up with the demand. The introduced technologies were able to automate the transportation process and drastically decrease human involvement in the process.

The fourth industrial revolution offers new tools and automation technologies. The technologies enable large-scale machine-to-machine and the internet of things communications. The communications open new routes of information exchange between all nodes involved in cyber-physical systems or CPS. By collecting valuable live feedback from CPS systems, the revolution can provide a predictive maintenance schedule, that allows owners to perform cost-effective maintenance. Further, the feedback can be used for creating a virtual copy of the real system to test solutions and experiment with the solutions, which can lead to new design improvements for the system (Oscar Carlsson, 2017,169).

1.1 Research motivation and objectives

The current thesis aims to contribute to the developed solutions that lead to decreasing human presence close to the dangerous operating areas. One of the ways to cut down the time of the presence is to make maintenance inspections to be done by sensing equipment rather than a human.

The goal can be achieved by creating monitor and drive structures for a conveyor belt system with the new technologies brought forward by the fourth industrial revolution. The monitor and drive structures must provide live inspection feedback as well as the transporting rate of the system. The information is further safely transmitted to cloud services for further analytic work to be carried out. The monitor and drive structures have to be able to complete monitoring tasks quickly, reliably and safely when compared to manual inspections.

The ability of the owner to observe conditions of the conveyor belt system remotely is beneficial for the goal of the thesis. The remote access sufficiently decreases the time required for regular visual inspections of the equipment done by a human as well as notify a human about problematic areas of the system in advance and tell if human presence is needed. If the time of human presence in a dangerous operating area is decreased so is the amount of harmful and life-threatening accidents related in the area. Such a strategy, implementing this type of the system, covers not only the goal of decreasing the share of human participation in the operating process but also give a rise to overall performance level and storing valuable feedback from sensors to the cloud services for further analytic work.

The prime interest can be described as striving to solve modern practical automation problems of the proposed subject, face obstacles related to the work, test personal capabilities and enjoy time spent on doing creative work.

For this paper to have an impact and contribute to the research field it must match the required level of objectivity and validity. Validity and objectivity could be affected by research material used and human nature to make mistakes. In order to minimize the influence of the effect on the work, the parameters are supervised by the supervisor with knowledge about the topic and a language teacher, assigned for the task.

The created monitor and the drive systems are applicable and meaningful for conveyor belt systems. If not, it runs on other automated motorized transportation systems used in a wide range of applications, executing similar types of hardware, e.g. chain conveyor or overhead conveyor type systems.

The interference of the proposed work with several spheres proves the relevance of the current study. The outcomes of the paper might amount to a tiny amount of knowledge that will be displayed to the scientific world for future developments to be done in the same sphere.

1.2 Framework

The theoretical framework defines boundaries that define what the current study is going to cover and what it aims to achieve. The limitations are useful to focus on the area of the work and prevent unnecessary tasks from being done.

Existing conveyor belt systems, available technologies related to secure data exchange and well-established cloud services are the reference points of the current work. Their working principles and operations are investigated for design considerations during the system designing process. The paper is going to cover a selection of data exchange protocols between drive nodes of the system, selection of sensing equipment to create a working monitoring system, the integration process of each of the hardware components to create a fully operational conveyor belt system and preparation an asset to be set in the cloud service.

Wiring diagrams of the sensing and driving hardware will not be covered by the study. Besides, the election processes of drive hardware as well as materials required for building the system, are removed from this paper.

Any conclusions and decisions made by the viewer, in terms of the abandoned topics, must be evaluated and researched by themselves or further developed.

2 Background information

2.1 Belt conveyor

A belt conveyor is an essential participant in executing automated distribution and material handling, not limited by type of goods being handled. The system has great adjustability, which can be changed to satisfy the demands of different industries. When compared to the other transportation and distribution systems, the conveyor belt is the least expensive in terms of installation and maintenance costs. Those factors make the system the most widely accepted mechanical mode for distribution and transporting goods throughout different industries.

The structure of conveyor belt systems is simple and consists of two or more drum assemblies, a conveying belt that rotates around them, the frame of the system and a motor that drives the pulley. Although the complexity of the structure is growing as the length of the conveyor belts increase to adhere to modern requirements. Figure 1 presents the working principle and the simplest arrangement of the system (D.V Subba Rao 2020, 3).

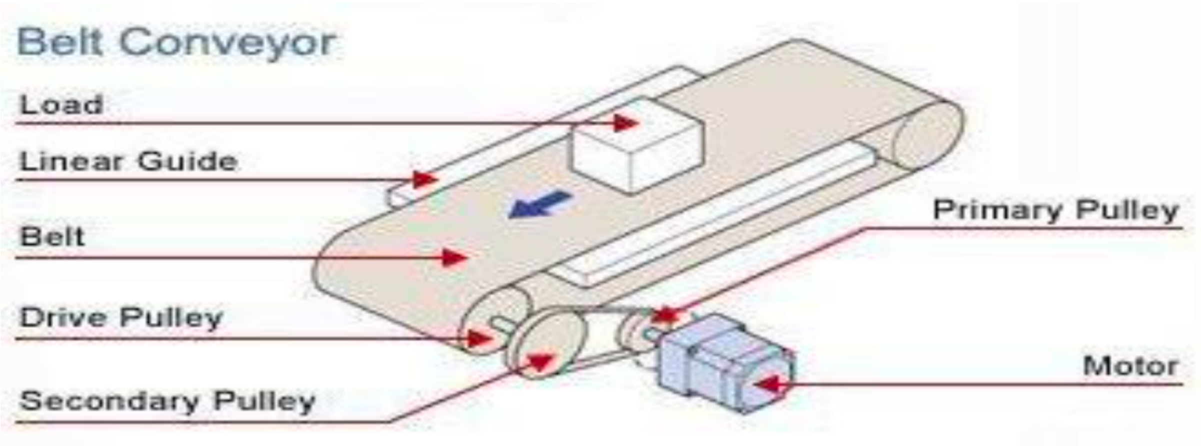


Figure 1. Working principle and arrangement of a conveyor belt system (mogroup.com)

Even though a conveyor belt system has numerous advantages, it is not an ideal medium for handling goods. Key characteristics of the system are listed and sorted in the up following table:

Advantages	Disadvantages
High reliability and durability	Limited to structural design
High adjustability	Low universality

Low initial costs	Requires loading and unloading equipment
Low running cost	Limited transmitting rate of a line
Possibility of some vertical movement	Difficulties to make 90° turn with one line
Possibility of transportation for a long distance	Requires complex belt tension system and frame

Table 1. Advantages and disadvantages of belt conveyor systems

Based on that information, a conclusion can be made – conveyor belt systems are able to match the requirements of the majority of production industries and secure their place as a leader among the systems for transporting goods.

2.2 Programmable logic controller

Most modern systems are automated and controlled by a programmable logic controller or PLC. A PLC is an industrial computer that was designed to work only with industrial applications in harsh arrangement conditions and was introduced to replace hard-wired relay logic systems. PLCs have become a reliable and compact tool for technicians to make logic operations and has allowed them to configure programs without any wiring efforts.

The basic principle of the computer is to produce real-time output results in response to conditions of input signals within a limited time. The computer is commonly used in automated assembly lines, robots as well as any industrial activities that require to be automated.

A typical PLC consists of:

- A processor unit (CPU) reads inputs, executes the control program and sends outputs
- A power supply unit (PSU)
- A memory unit which stores data from inputs and program to be executed
- An input and output interface (I/O interface) where the controller does data exchange
- A communication interface to receive and transmit data on the network

For developing a program and later downloading it to PLCs, a programming terminal is required. Figure 2 gives a visual representation of components that PLCs consist of.

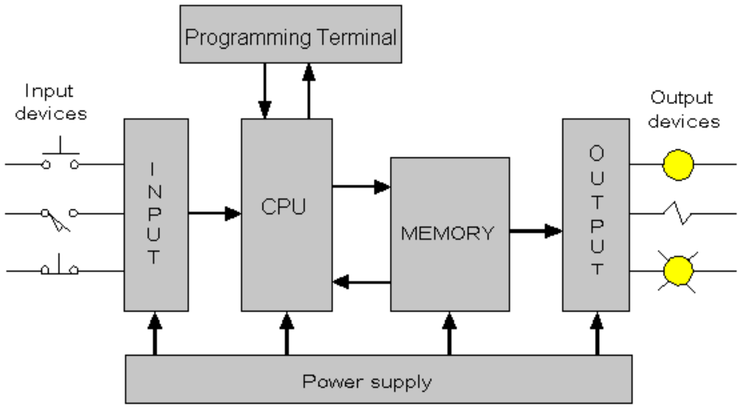


Figure 2. PLC components (plctechnician.com)

2.3 Discrete and analog signals

Through the I/O interface, a PLC gains information about the condition of the devices connected to it. There are two main types of signals that PLCs can read and send through the I/O interface: digital and analog. Although both signals use voltage and current to send information of their condition, the way the interface interprets the condition for the CPU differs.

Discrete or digital signals can only have two conditions, on or off value. A simple example of devices that provide a discrete signal are switches. Digital signals are sent by using voltage or current, where the range of low and high level of the signals is assigned as logical 0 or 1. As such, PLCs might use 10V DC input with values above 8V DC to be a logical 1, where values below 2V DC to be logical 0. However, the behaviour of logical value in between the two ranges can vary (Amrutha Varshini, 2021). The two states of the signal create a waveform that is shown in Figure 3.

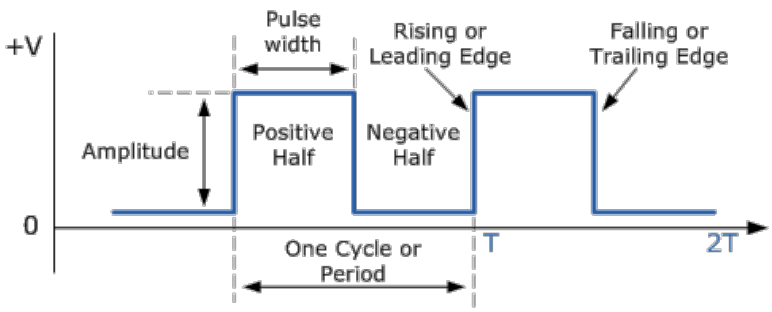


Figure 3. The waveform of a discrete signal (technlab.blogspot.com)

Analog signals have more than two conditions of a value. An analog signal has a range of voltage or current it works with, where the minimum and maximum values are set. The range between them is used to get discrete values. The limitation of the range is utilized as the reference point for scaling the discrete values to get a real value for the CPU. For instance, an analog signal is an audio signal, where the voltage of the signal depends on the pressure of the sound wave. Most of the I/O interfaces use a range of 0 to 10V or 4-20mA to scale analog signals. Then the signal is converted into an integer range of 0 to 32767. Further, the PLCs take this range and transpose it into the desired units, so the program or operant can read it (Amrutha Varshini, 2021). An analog signal has a sine waveform shown in Figure 4.

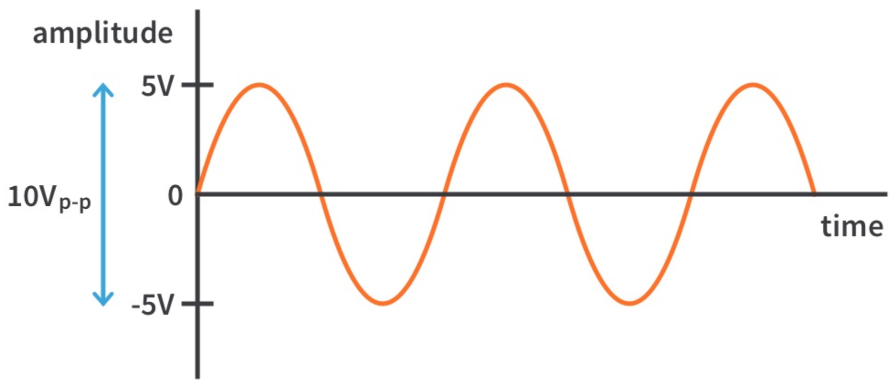


Figure 4. An analog signal waveform (circuitbread.com)

2.4 Programming terminal

Since PLCs require a program to be written and downloaded inside the memory, a programming terminal usually takes the form of special programming software on a personal computer. The vast majority of the PLC manufacturers develop their programming software for their controllers that provide extra features such as hardware diagnostics, simulations and software debugging.

Modern programming software supports a wide variety of programming languages used to create a PLC program. The Two main types of programming languages that are used for development: textual programming language and graphical language of which is the most common type. Graphical language includes ladder diagram, function block diagram and sequential function chart. Although, ladder and function block diagrams are widely accepted tools for creating PLC programs when the sequential functional chart is used for teaching new programmers (Wikipedia, 2021c).

Ladder logic is a programming language that evolved from circuit diagrams of relay logic hardware. Because of the similarities, ladder logic is simple to understand and became commonly accepted with the introduction of PLCs. The logic can replicate simple functions such as logical AND or logical OR functions. Ladder logic programs have inputs named checkers and an output named an actuator. Usually, the output signal is shown with round brackets. Figure 5 illustrates the basic functions created with a ladder logic diagram.

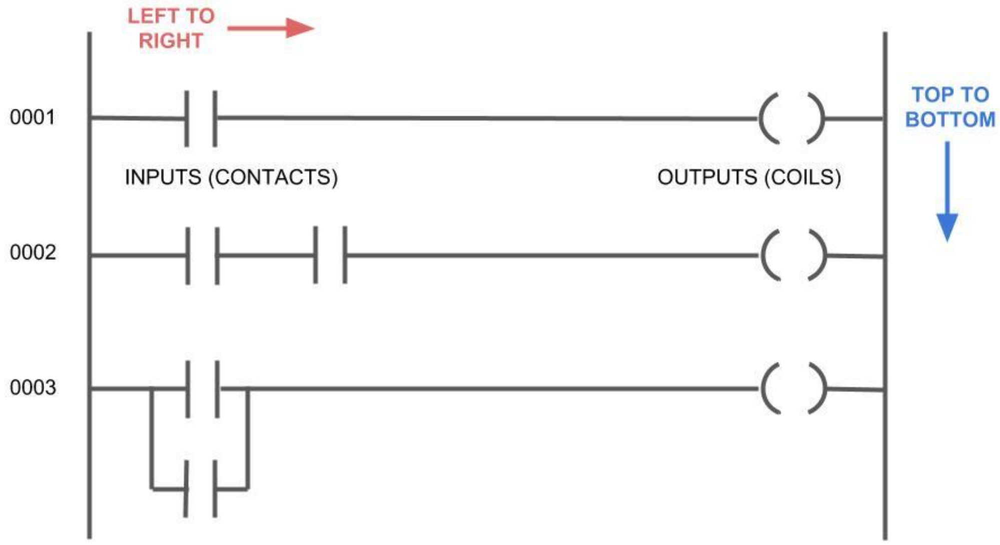


Figure 5. Ladder logic diagram functions (learnrobotics.org)

The function block diagram is a programming language that can describe the function between input variables and output variables. A complex function can be described through a set of elementary blocks, grouped to create a complex function block. Figure 6 presents a basic function block diagram.

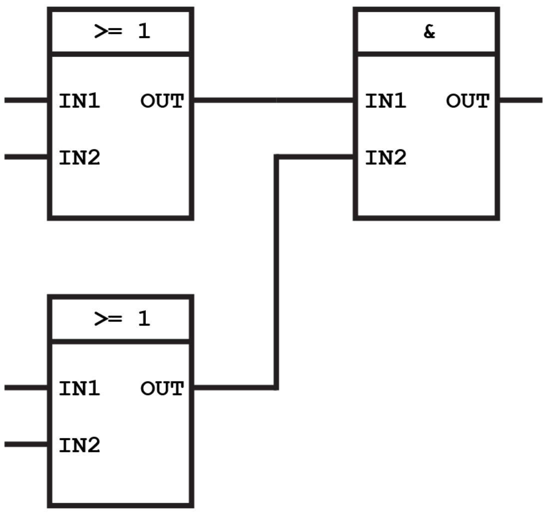


Figure 6. Function block diagram program (plcacademy.com)

However, the ladder logic diagram is commonly used in tandem with function block diagrams to create PLC programs, since the modern level of automatization requires complex functions to be used for creating an output signal.

2.5 Data communication protocols

Modern automatization systems consist of numerous complex operating machines. For them to cooperate in one network, communication protocols were invented. Communication protocols represent a set of rules that allow a communication network to transmit information between the participants. The protocols include not only the information exchange rules but error recovery methods, semantics and synchronization of the communications. Data communication protocols are typically developed into a technical standard for simplicity. The most common data communication protocols that can be encountered in modern PLCs are EhterNet/IP, Profibus, Modbus and ProfiNet communication protocols (Wikipedia 2021c).

2.6 AC Motor

Most present-day industries use AC motors as the main drive source for automated systems. An AC motor is an electric motor driven by an AC current. The construction of a typical AC motor is basic, consists of two main parts, a stator that produced a rotating magnetic field, and a rotor connected to the output shaft for transmitting the torque. Figure 7 presents a simple construction of an AC motor.

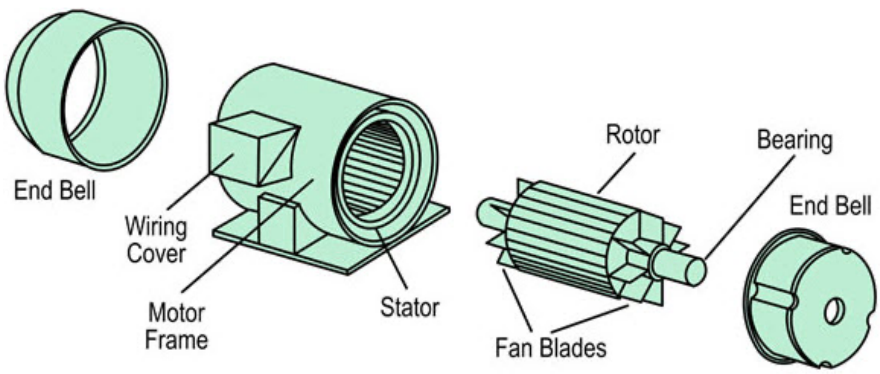


Figure 7. Simple construction of AC motor (elprocus.com)

AC motors are divided into two main categories: induction AC motor and synchronous AC motor. The main difference between the two is the operating principle. The induction AC motor generates torque by relying on a speed difference between the stator’s rotating magnetic field and the rotor shaft. The motor operating principle for the induction AC motor is called the slip-induction principle. The synchronous AC motor principle doesn’t rely on the slip principle, the motor produces torque by synchronizing the rotor shaft rotation with the rotation of the stator's magnetic field. Although the synchronous AC motor has the advantage of speed per frequency, the induction AC motor is the most common type used for automatization purposes because the induction motor has lower manufacturing costs (ELPROCUS, 2021).

For conveyor belt systems a 3-phase induction AC motor is commonly used. The stator has three-phase windings which are overlapped with each other at a 120-degree phase shift. There are three windings inside the stator that have six ends, two for each. One winding consists of two wires: one is a phase wire through which current flows and creates a magnetic field and the other one is a neutral wire, which is a return path to complete the circuit. The rotor is attracted by the magnetic field, created by the stator, and rotates towards it, before the rotation happens the acting winding change happens (ELPROCUS, 2021). Figure 8 shows a basic principle of 3-phase induction AC motor.

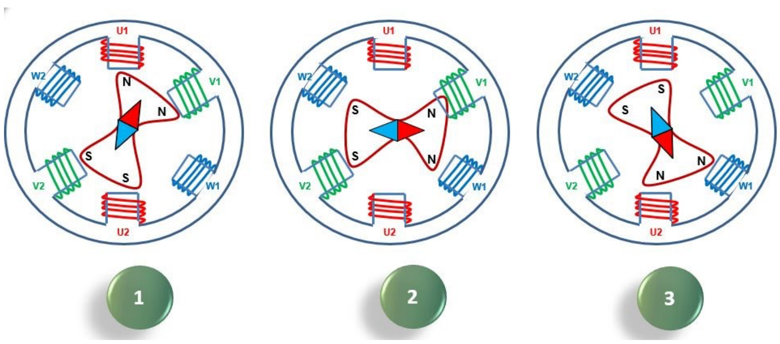


Figure 8. Basic principle of induction AC motor (flaktgroup.com)

As was previously mentioned the neutral wire is used to complete the circuit, since the 3-phase system has three neutral wires which have an effect on the performance of the system. There are two main types of neutral wiring connection: a star connection and a delta connection.

In a star connection, the phase wires are connected to a common point and the neutral wire is taking from the common point. The connection affects basic values of the system such as current and voltage. In a star connection, the phase wire current is equal to the line current and the line voltage is equal to the square root of phase voltage multiplied by three. The voltage has an effect on the rotational speed of the motor and reduces the speed (CurcuitGlobe, 2021). Figure 9 shows a simple wiring diagram of a star connected AC motor.

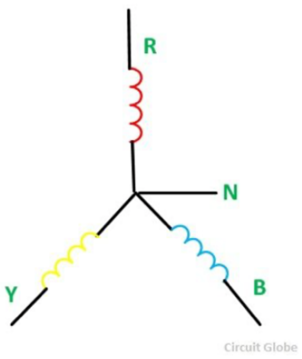


Figure 9. Star connection (circuitglobe.com)

In a delta connection, the phase wires are connected forming a closed loop, illuminating the neutral point of the system. In this connection, each line receives less current when compared to the phase current. Although, the line voltage is equal to the phase voltage. Because of this, the rotation speed of the motor is higher due to each of the phases getting the total line voltage (CurcuitGlobe, 2021). Figure 10 gives a better look at the delta connection.

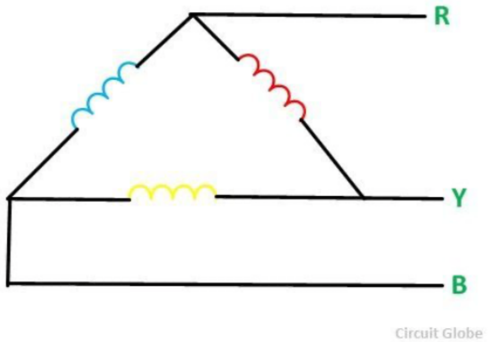


Figure 10. Delta connection (circuitglobe.com)

2.7 AC Drive

An AC drive or VFD (short for variable frequency drive) is an electro-mechanical device that regulates the speed and torque of an AC motor by adjusting motor input frequency and voltage. Since electric motors consume about 25% of the world's electrical energy, an AC drive is used to control the system which makes an impact on the overall system efficiency, when compared to other throttling control tools.

An AC drive is wired between the main grid and an AC motor. The drive consumes the grid energy, then converts it to DC electricity, filtering the electric noise through the use of capacitors and amplifies DC electricity back to AC with adjusted parameters to be feed to the AC motor (ABB, 2016). Figure 11 illustrates the basic working principle of an AC drive.

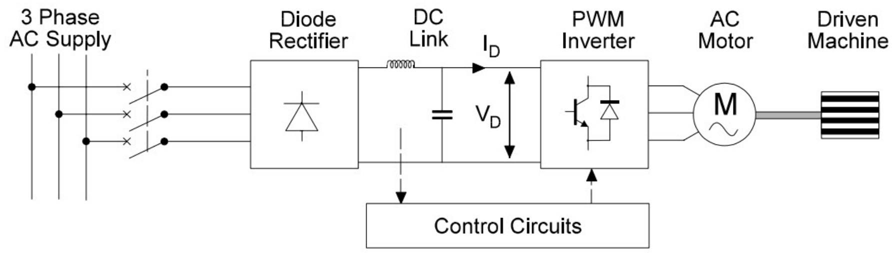


Figure 11. AC drive system working principle (laptrinhx.com)

The control circuit usually is linked with a user interface, where technicians write values that describes required motor behaviour. The information of the behaviour is further read by the control circuit. The circuit has an effect on the inverter output characteristics and the inverter amplifies the voltage, current and frequency of the phases to match the users' settings. Modern AC drives are able to restore some of the electricity from braking action, increasing the efficiency of the system and reducing the impact on the surrounding (ABB, 2016).

2.8 Torque transmission technologies

There are two main technologies that are used to deliver torque from an electric motor to a conveyor belt: direct and indirect technology.
Direct torque transmission is a technology based on transmitting the torque when an electric motor and drive pulley are assembled in one product. The technology is used only for food-related conveying applications since it outperforms the competitor in hygiene-related comparisons.

However, the most commonly used technology in all industrial applications is an indirect transmission. Indirect transmission is a technology based on transmitting the torque when an electric motor is located outside of a drum. The location allows for quick and cheap swaps of the motor if maintenance is required. The technology enables the flexibility to select a motor for any purposes and power since there is no size limitation (Davide Barbanti, 2017). Figure 12 displays the difference between the two technologies.

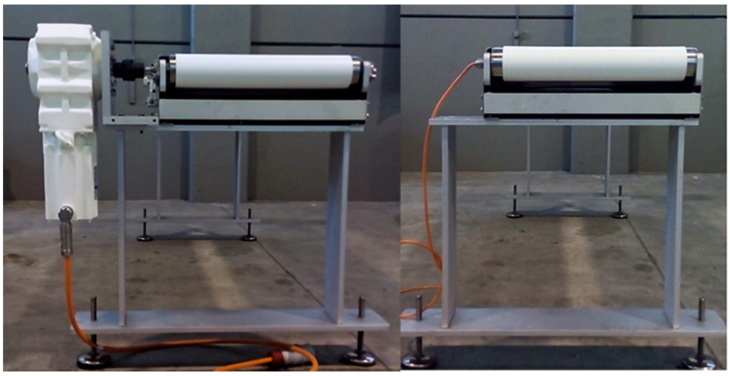


Figure 4. Motor gear (left) and drum motor (right) technologies (motioncontroltips.com)

2.9 Industry 4.0

The fourth industrial revolution is the digital transformation of manufacturing and production-related industries. That way it creates a new stage in the organization and control of the industrial value chain. The revolution is characterized by even further automating systems compared to the third industrial revolution, connecting physical and digital worlds through cyber-physical systems and shifting from a central industrial control system to one where smart products define the production steps, creating a smart factory.

The goal of the revolution is to integrate cyber-physical systems and humans. To make that possible, four main design principles were created:

- Interconnection – the ability of machines, devices, sensors and people to connect and communicate with each other via the Internet of Things
- Information transparency – provide operators with significant information, collected from all the points in the manufacturing and production processes to make a decision.
- Technical assistance – cyber-physical systems assist humans with decision-making and problem-solving, as well as the ability of the systems to help with difficult or unsafe tasks for humans
- Decentralized decisions – the ability of cyber-physical systems to make a decision on their own based on the collected information and to perform their tasks as autonomously as possible

As can be seen from the design principles, the production and manufacturing industries are in the early stage of making progress in the revolution. Although, new technologies are advancing and introducing every year that move the industries toward the four principles mentioned above (G. Lodewijks & Xiaoli Jiang & Wenfeng Li, 2016, 1; Wikipedia. 2021d.)

3 Initial project hardware

3.1 Computing network

The Computing network of the current project consists of a Siemens SIMATIC S7-1516-3 PN/DN CPU module to which AI, AQ, DI and DQ extension modules are attached. The CPU module has built-in support for the most common communication protocols widely used in the industries.

For programming Siemens CPUs, TIA Portal is currently the most commonly used software as the main programming terminal. TIA Portal was designed and developed by Siemens AG to work exclusively with Siemens's industrial programming hardware such as PLCs or automated drives. The software is able to install programming libraries of third-party developers if the libraries are optimized by the parties. TIA Portal supports a majority of the programming languages and tools used to program PLCs. It is worth mentioning that TIA Portal has plenty of support documentation available on Siemens's site and online forums created to assist technicians with the problem-solving process.

The computing network was recreated in TIA Portal V15. Figure 12 and Figure 13 were made to give a better view of the hardware components that form the network.

Device overview

Module	Rack	Slot	I address	Q address	Type	Article no.	Firmware
	0	100					
PM 190W 120/230VAC	0	0			PM 190W 120/230...	6EP1333-4BA00	
Thesis Station	0	1			CPU 1516-3 PN/DP	6ES7 516-3AN01-0AB0	V2.5
PROFINET interface_1	0	1 X1			PROFINET interface		
PROFINET Master connec...	0	1 X2			PROFINET interface		
DP interface_1	0	1 X3			DP interface		
AI 8xU/I/RTD/TC ST_1	0	2	0...15		AI 8xU/I/RTD/TC ST	6ES7 531-7KF00-0AB0	V2.1
AQ 4xU/I ST_1	0	3		0...7	AQ 4xU/I ST	6ES7 532-5HD00-0AB0	V2.1
DI 32x24VDC HF_1	0	4	16...19		DI 32x24VDC HF	6ES7 521-1BL00-0AB0	V2.1
DQ 32x24VDC/0.5A HF_1	0	5		8...11	DQ 32x24VDC/0.5...	6ES7 522-1BL01-0AB0	V1.0

Figure 12. Overview of computing network's modules

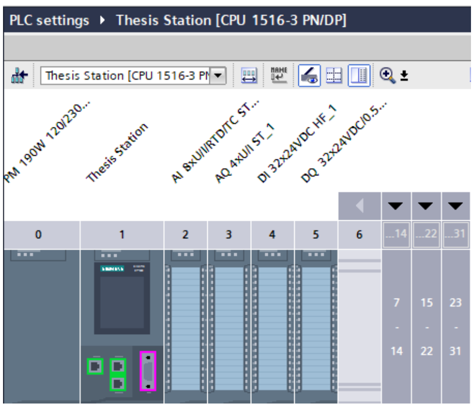


Figure 13. Siemens SIMATIC S7-1516-3 PN/DP device view

3.2 Driving hardware

For the current study the driving system consists of an electric AC motor and an AC drive. The AC motor is used to transmit torque to the driven drum when AC drive controls the output torque and speed of the motor. By using the system, users have the flexibility to adjust the output characteristics of the motor depending on the application being used. The adjustments of the output can be done by just changing the related settings in the AC drive.

In the project the electric AC motor is an ABB M3AA 132M-4 AC motor. The selected AC motor is reliable and has an efficiency class of IE3. The IE3 efficiency class is the highest rating that an induction AC electric motors can be rated. The drive is rated at 7,5kW output power and connects to a 600V electric grid. The motor's properties are further used for the integration process.

Figure 14. ABB M3AA 132M-4 induction motor (abb.com)

As for the AC drive, the choice is ABB ACS880-M04 Industrial drive. The drive was designed for AC motors that use power voltage ranging between 380V and 690V. It offers a power range from 550 W to 250 kW and is able to regenerate some of the power back though braking actions. Built-in I/O interface board and electric ports for connecting modules show potential for expanding the purposes of the device for newer automation demands. The drive's software is supported by the PLC manufacturer and beneficial for the integration process. A pre-programmed monitoring system collects and analyses electrical values of the drive system in real-time and can be shared with a user via Bluetooth. ABB supporting manuals, guidelines and forums can be effortlessly found on the internet, which are valuable for troubleshooting steps.

Figure 15. Device view of ACS880-M04 (abb.com)

3.3 Cloud network

For automated lines to be remotely examined, feedback information must be securely sent and stored on cloud-based systems. Further, inside the cloud systems analytic apps and visual tools could be created or installed to deliver a better overview for the user. Based on the analytic data, the user decides if the automated workflow of a system requires human involvement.

MindSphere takes the role of cloud service for the study. It was developed by Siemens and supports every automation product designed by the same company. The service is aimed only at Industrial IoT applications and management. MindSphere has a subscription-based pricing model, however users are free to choose between plans or create their own value plan. MindSphere is equipped with development tools, application programming interfaces and an app store. In the end, the service becomes a flexible software, that can be tuned according to applications and is a reliable participant of modern automated systems. Documentation papers and instructions, for merging with the hardware are available on the internet, making it easier to learn.

To allow the conveyor belt system to exchange data with the cloud, MindSphere Nano IoT gateway was used. The device is the intermediary between the cloud and the hardware. It supports the transmission of encrypted data and protects the system from cyber-attacks. MindConnect Nano has a high rate of transmission speed, up to 250Mb/sec. As well as having a built-in memory to store the monitored data if internet goes down, so users won’t lose the valuable feedback from the system.

Figure 16. MindConnect Nano device view (siemens.com)

4 Network development

4.1 Automation pyramid

Automation pyramid is a concept of representing the layers of automation within a factory. The pyramid includes five layers, where each of the layer is based on the one located below and transforms information to the one located above (Jakes Mantle, 2019). For better visualisation of the automation pyramid Figure 17 is presented.

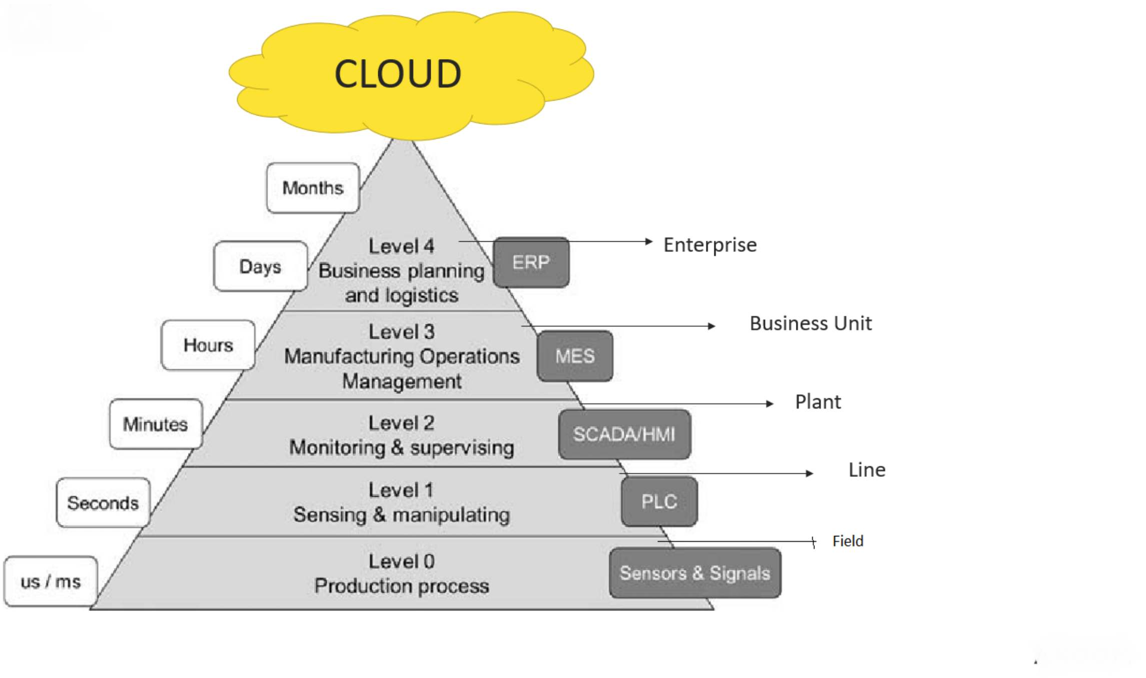


Figure 17. The structure of automation pyramid during Industry 4.0 (medium.com)

As can be seen on Figure 17, the base level of the pyramid is the field level. The level includes production devices, sensors and actuators that can be found in the most production floors. The field level is the space where physical work and monitoring of the work take place.

The next layer is the control level. At this level, devices that manipulate the field layer, which do the physical work are located. The control devices take information from input sources to make decision on output signals to complete the task.

The supervisory layer in the automation pyramid is designated to SCADA. SCADA stands for supervisory control and data acquisition. On this level, HMI or Human Machine Interface is added to control and monitor workflow of the control level through sensing data from the previous layers.

The third level in the pyramid is called the planning layer. The layer utilizes a manufacturing execution system. The system is used to monitor the whole process of manufacturing in a factory, from raw materials to finished products.

The last level of automated pyramid is called the management level. On this level, a factory uses enterprise resource planning or ERP for short. ERP helps a factory gain information from sales of a product, purchasing raw materials and logistics. By implementing ERP, factories control and make each step relating to the business aspect more efficient.

On the top of the automated pyramid through the introduction of the industry 4.0, is the cloud level. Cloud based implementation makes it possible to transform data from any of the layers directly to other applications or for simulations to gather feedback from data close to the reality. This is the next step in achieving the next level of factory's efficiency and operation flow (Jakes Mantle, 2019).

4.2 IIoT network architecture

Industrial IoT network design consists of four architecture layers: IoT device layer, IoT getaway layer, IoT edge IT and IoT cloud. The designing process of the architecture is a crucial process for users to get the feedback about the performance of the industrial operations. The knowledge about the IIoT network is an essential when creating a new system that is aimed to contribute to the digitalisation process of companies (Alexsoft, 2020).

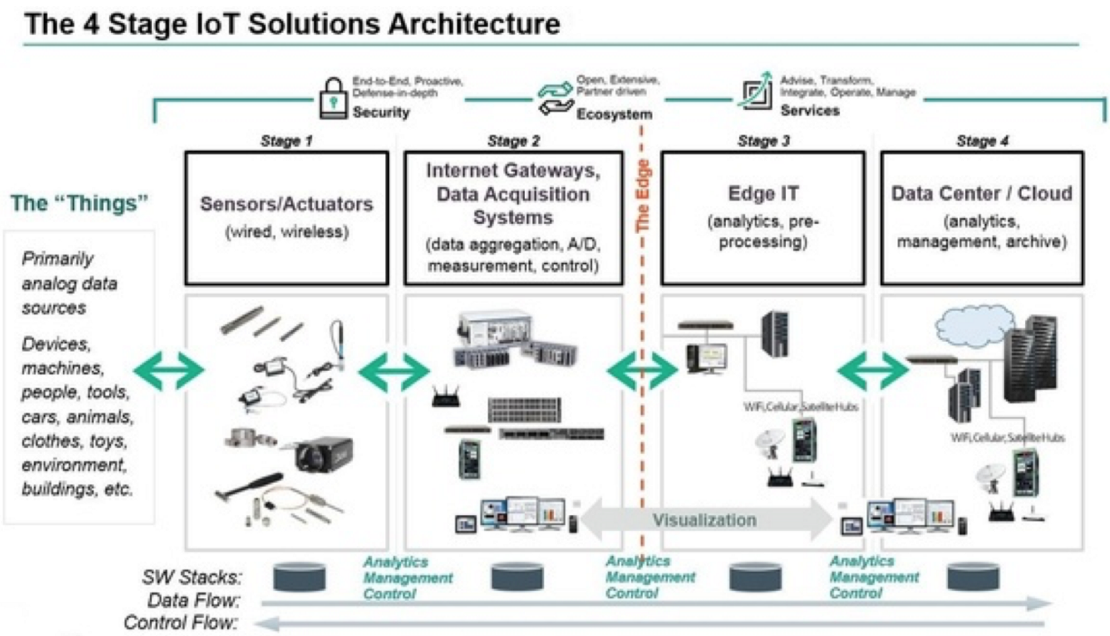


Figure 18. IIoT network architecture stages (medium.datadriveninvestor.com)

The first stage is based on the ability of the sensors to convert the information from the field devices into data for analytical work. The sensors supply the raw material to which the management crew or automated systems can make a decision or get insight toward production line automatization.

The second stage is the layer in which data is first analysed by the data acquisition systems. The system processes the large amount of raw information collected by the sensors and compress it to the optimal size for further analytic work. Data acquisition systems assign timing to the data as well as shows how data varies throughout the operating process.

In the third stage the data from previous level is visualised. The stage interferes with some of the human design-making process. As such, it is required for the data to be visualised before it goes to the real analytic work. In this stage the first conclusions can be made regarding the systems performance and operating processes.

The last stage of IIoT network architecture is a cloud service. Inside the data centre, in-depth processing happens as well as revision of the feedback. Here, the collaboration of IT and production management comes together to analyse and perform changes to systems to match the factory’s standards or expectations from systems. Therefore, the data from other systems can be included at this level for comparison to make more precise designs and decisions (Alexsoft, 2020).

4.3 Project network and topology

Physical and logic organisation network of the devices were created based on the principles of the automation pyramid and IIoT network architecture, to make an automated conveyor belt system for IIoT purposes.

Physical network of devices is based on the automation pyramid and it helps users clearly understand the arrangement of the network participants. The topology views communicating devices as nodes and connection between them as links; hence users can observe the physical layer of a system before it is built. That way changes in the physical layer of the system could be easily merged on paper, rather than in real life. The following Figure 19 gives a better look at the participants of the project as well as the physical connections linked between them.

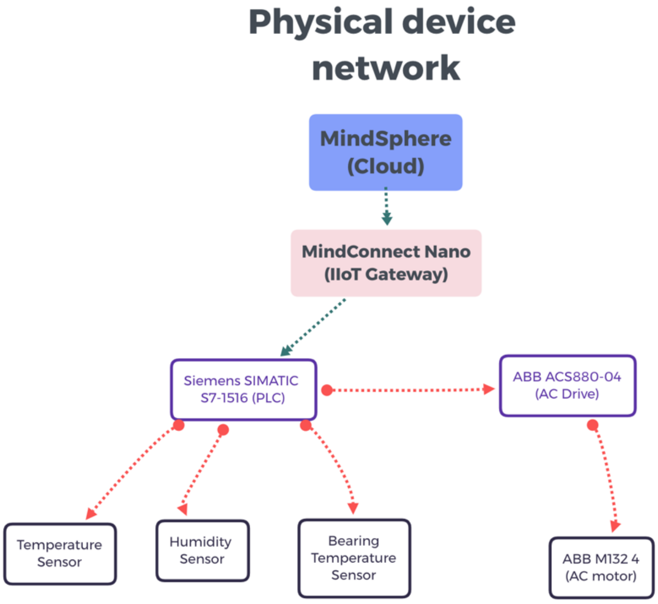


Figure 19. Physical topology of the conveyor belt system

As can be seen in Figure 19, most of the physical links are attached to the computing unit of the project, which becomes the critical node of the system. The network executes the CPU as its data hub and processing power for most of the information. If the computing node fails, the system will not be able to operate as it was intended to. Although the physical layer is enough for the goals of the current project, the network is easily expandable to fit the needs of a user. Limitations are set only by the computing unit's capacity to process information without errors and hardware failures.

Logical topology is based on IIoT architecture and on the arrangement of devices inside a network highlighting the ways the devices communicate with each other. The logical topology view shows how signals are acting inside the network; hence users can observe and edit feedback signals before it is built. As a frame, the topology uses the physical layer view and further data exchange links are drawn. Figure 20 visualizes the logical topology of the study. The numbers of the nodes represent participants in the same order as in Figure 19.

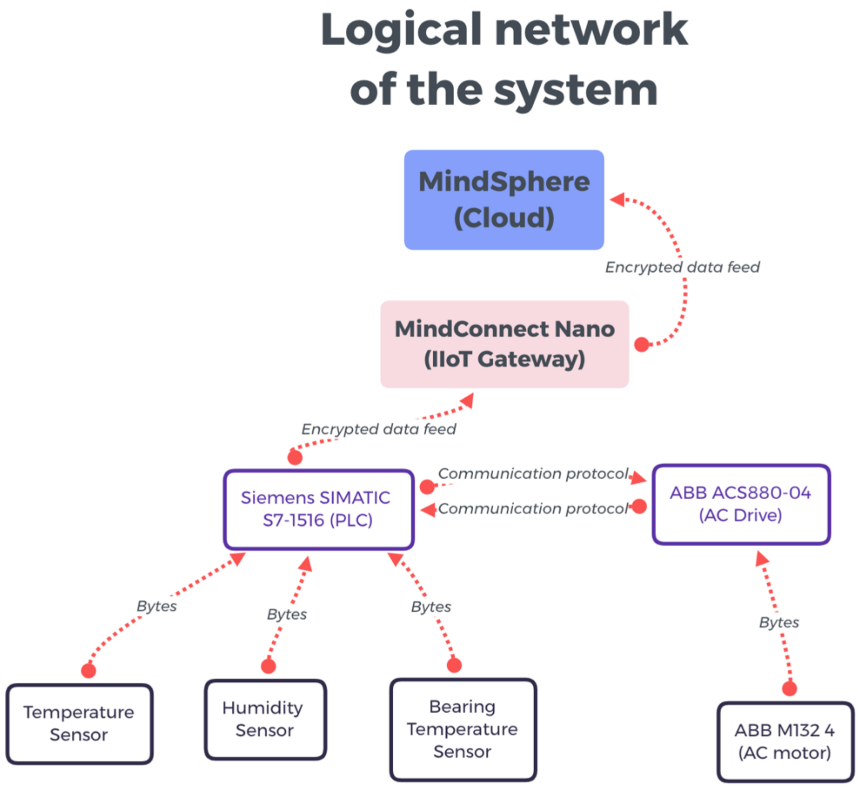


Figure 20. The logical topology of the study

The figure clearly shows that the PLC has the most influence, that making it the information optimizer hub of the network. Sensing nodes are the monitoring system for the conveyor belt system, to which the computing device has access to gain feedback information sensed through output bytes. The PLC and the AC drive are constantly in a feedback loop enabled by the communication protocol and both of them represent the control system of the conveyor belt. The loop allows them to influence each other for work optimizations. The drive has full control over the AC motor's actions through information bytes. The hub further shares information collected from all the monitoring and control nodes and sends it to MindSphere through the MindConnect Nano IIoT gateway.

4.4 Selection of monitoring equipment

In the modern world, many types of sensors are available for purchase and understanding the application type and basic operating principle is an important aspect of the study. Finding the right sensor for the application starts with identifying the operating conditions in which the senso will be exposed to for a specific application.

4.4.1 Bearings inspection

One of the maintenance inspection tasks of conveyor belt systems is making sure that drum bearings are in a good condition. A way to tell that a bearing needs to be replaced or maintained is through a change in operating temperature. The difference in operating temperature can occur if extra lubrication is needed or bearing balls been damaged. Monitoring the temperature of drum bearings helps with better organizing of the maintenance schedule and by accessing the information remotely, it decreases the time of human interactions close to the moving parts.

One limitation for selecting the right sensor is the location of a sensor. By installing a probe tip inside a bearing housing, the sensor will measure a precise temperature value. Placement of the sensor allows for a quick swap to be done if a failure of the equipment occurs. Figure 21 shows the installation drawing of the bearing housing mounting.

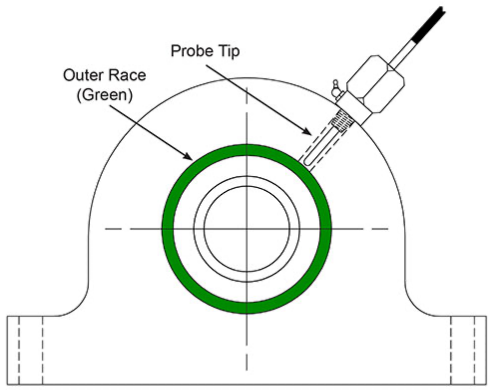


Figure 21. Bearing housing mounting (www.go4b.co.uk)

There are two groups of temperature sensing device that are used for this type of mounting: thermocouple sensor and resistance temperature detectors (or RTD for short). A thermocouple consists of two different electric conductors that produce temperature-dependent voltage, as a result of the Seebeck effect, that is further measured and interpreted in output temperature values. Resistant electric conductors consist of a sensing element that changes its resistance according to the applied temperature which then measures and interpretes as a temperature value (Omega, 2019). Figure 22 represents the basic principle of the sensor types.

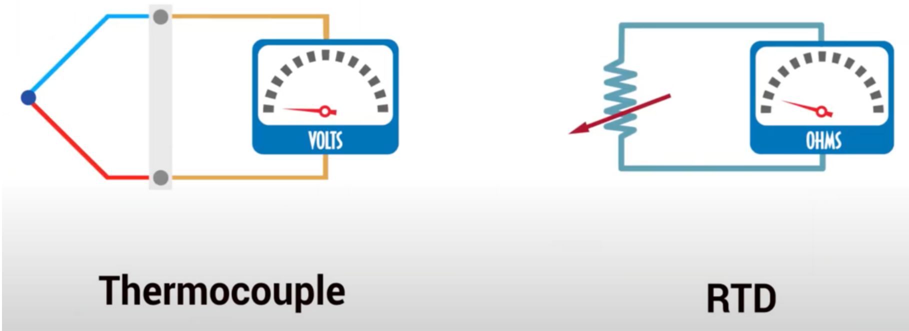


Figure 22. Basic operating principle of thermocouple and RTD (omega.com)

The next step is a comparison by defining qualities, comparing each of them in one table and giving a general description of the qualities. The results can be seen in Table 2.

Quality	**Thermocouple**	**RTD**
Temperature range	✓	
Cost	✓	
Installation cost	✓	✓
Sensitivity	✓	✓
Accuracy		✓
Stability		✓
Linearity		✓

Table 2. Thermocouple and RTD quality comparison

Temperature range

The range of a sensor is the lower and upper values that can be measured. Thermocouples have a greater ability to work at high temperatures of up to 2500ºC, while RTDs work in a range below 400ºC.

Cost

Thermocouples are cheaper to manufacture since the sensing technology is simple to replicate. RTDs cost more than thermocouples, with the same properties.

Installation cost

Both of them are simple in the installation process and require the same amount of work, although the RTDs have the possibility to use cheaper wiring cables.

Sensitivity

This is a period when a sensor changes the output state after the input parameter has changed. Both of the sensors quickly notice the temperature change of the system, but the thermocouple relies on a physical response which usually acts faster.

Accuracy

This is a property that is equal to a minimum difference in values of the output parameter and the actual value. RTDs are more accurate, with the typical accuracy of 0.1ºC compared to the average 1ºC accuracy of thermocouples.

Stability

It is a property that represents sensor reading to stay constant throughout time. RTD probe readings stay stable and repeatable for a long time. Thermocouple probe readings suffer from drifting of the value due to chemical changes occurring at the sensor tip (Joseph J. Carr & John M. Brown, 1998).

Linearity

It is a graph expression that reflects how distant is measuring the curve of a sensor to the ideal mearing curve and so influences the output result. The temperature-resistant RTDs has an almost linear sensing curve while thermocouples have an “S”-shaped measuring curve (Joseph J. Carr & John M. Brown, 1998).

Summing up the obtained data, RTD has more attractive and justified properties for the project. The sensor type must deliver reliable and stable output measurements throughout the whole operating time.

After researching to find a sensor on the internet to buy, two sensors were chosen to be the part of the system for sensing the bearing temperature: SITRANS TS300 and SITRANS TS200. They both have almost identical sensing characteristics. However, SITRANS TS300 can output HART signals and has greater sanitary standards that allow the sensor to be used for the pharmaceutical industry and food industry. Table 3 gives a general comparison of the properties.

Property	SITRANS TS200	SITRANS TS300
Application	Bearing temperature and surface measurement	Bearing temperature and surface measurement
Output signal	Digital 4-20 mA	HART
Minimal respond time	2 … 6 seconds	5 seconds
Temperature limits	-30 … +400ºC	-30 … +300ºC

Table 3. Comparison SITRANS TS200 and TS300

SINTRANS TS200 was selected as the choice for monitoring the temperature of the drum bearings.

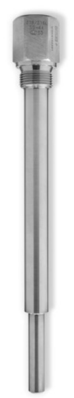


Figure 23. Siemens SINTRANS TS200. (siemens.com)

4.4.2 Surround monitoring equipment

Surrounding temperature and humidity have a crucial impact on the reliability of the system. Monitoring the area where the conveyor belt system operates can give a prediction regarding the life-spawn of the components. Supporting the constant and best operating temperature as well as ensuring correct humidity of the surroundings is beneficial for any systems reliability. Prolonged exposure to moisture initiates rapid rust occurrence which impacts the whole productivity of the system. Holding the same temperature of the surrounding helps rotating parts with heat dissipation. Dissipating heat at a constant rate decreases the chances of thermal expansion fatigue-related failures from occurring.

The modern market allows for the purchasing of sensors that have temperature and humidity sensors integrated on a single board. The market can offer a wide variety of sensors to be purchased cheap but most of them use the same type of sensor, hence arises lack of diversity in sensor types. Based on that, the selection process of the sensor will only be formed through a comparison of the products characteristics.

By searching the Siemens store for the type of sensors, these products were selected: Siemens QFA2071, QFA3171 and QFA4171. In Table 4 properties of each product are listed and compared.

Property	**QFA2071**	**QFA3171**	**QFA4171**
Description	Room sensor for temperature and humidity	Room sensor for temperature and humidity for demanding requirements	Room sensor for temperature and humidity with calibration certificate
Measuring range, temperature	0…50ºC, -35…35ºC, -40…70ºC	0…50ºC, -35…35ºC, -40…70ºC	0…50ºC, -35…35ºC, -40…70ºC
Measuring range, humidity	0…95% r.h.	0…100% r.h.	0…100% r.h.
Accuracy	At 0...95 % r.h. and 23 ºC: ±5 %, At 30...70 % r.h. and 23 ºC: ±3 %	At 15…35 °C: ±0.6 K, 23 % r.h., 0…100 % r.h.: ±2 % r.h.	At 15…35 °C: ±0.6 K, 23 % r.h., 0…100 % r.h.: ±2 % r.h.
Output signal	DC 4...20 mA	DC 4...20 mA	DC 4...20 mA
Time constant	Humidity <20 s, Temperature <8.5 min	Humidity: 20 s, Temperature: 510 s	Humidity: 20 s, Temperature: 510 s
Degree of protection	Not specified	IP65	IP65

Table 4. Comparison QFA2071, QFA3171 and QFA4171.

As a choice for monitoring the temperature and humidity of the operating area of the conveyor belt system, **QFA3171** was selected. The sensor has the right output signal for the computing unit, as well as excellent accuracy and the measuring range meets the demand.

Figure 24. Siemens QFA3171 (siemens.com)

4.4.3 Productivity monitoring

Conveyor belt productivity could be measured by the number of goods being transported through it. Most conveyor belt systems are used for transportation of countable goods for example boxes, bottles and manufactured parts. Monitoring the value could shed a light on the real productivity of the system.

There are several groups of sensors that able to detect if an object has entered the conveyor belt such as photoelectric switch, magnetic sensors and ultrasonic sensors. An ultrasonic sensor consists of an emitter of sound and a receiver. If an item passes through it and blocks the sound going toward the emitter it registers the item. A photoelectric switch represents an optical sensor that detects interruption of a beam of light and registers an object. A magnetic sensor detects a sudden change in the magnetic field by an object interacting with it hence noticing a product on the belt. For the selection process of the sensing types, Table 5 was created and lists the main qualities for the project

Quality	**Ultrasonic sensor**	**Optic sensor**	**Magnetic sensor**
Durability	✓	✓	

Works in dusty surrounding	✓		✓
Senses non-metallic objects	✓	✓	
Senses reflective objects	✓		✓
Sensing range		✓	
Reliability		✓	
Cost		✓	✓

Table 5. Quality comparison of ultrasonic, optic and magnetic sensors

As the result, an optical proximity sensor was selected for the project. It has met properties that are commonly used in most conveyor belt application. It gives a reliable output signal if working with non-reflective surfaces as well as being cost-efficient.

After the market was scanned, three options of products for counting objects on the conveyor belt system: Siemens SIMATIC PXO100 M18, SIMATIC PXO200 18S and SIMATIC PXO300 K21. In Table 6 their properties are specified.

Properties	**PXO100 M18**	**PXO200 M18S**	**PXO300 K21**
Operating mode	Thru-beam sensor	Thru-beam sensor	Diffuse sensor
Sensing range up to	20cm…30cm	4m…6m	40cm…50cm
Light type	Infrared light	Visible red light	Visible red light
Output	PNP/NPN	PNP	PNP/NPN

Table 6. Comparison of PXO100 M18, PXO200 M18S and PXO300 K21

Based on the provided properties **PXO200 M18S** was selected to be the sensor for the productivity of the conveyor belt system. It has an excellent sensing range and the operating mode of the sensor can detect the passing object accurately, which increases the reliability of the obtained information from the sensor.

Figure 25. Siemens SIMATIC PXO200 M18S (siemens.com)

4.5 Selection of data communication protocol

There are two major types of information exchange protocols used in Siemens CPU products: PROFIBUS and PROFINET protocols. Selection of the data protocol is an important step to do since the ABB ASC880-M04 drive does not have a built-in data exchange protocol port and an expansion module has to be used.

To select the best communication protocol for the system characteristics of each must be compared and so Table 7 was created.

Characteristics	**PROFIBUS**	**PROFINET**
Application Profiles	Same	Same
Physical layer	RS-485	Ethernet
Speed	12 Mbit/s	up to 1Gbit/s
Address space	126	Unlimited
Technology	Master/Slave	Provider/Consumer
Connection with other buses	Limited	Supports majority
Wireless connectivity	No	IEEE 802.11, 15.1
Machine-to-Machine	No	Yes
Grow in popularity	Slight	Constantly increasing

Table 7. Characteristics comparison of PROFIBUS and PROFINET protocols

Based on the comparison above PROFINET was selected to be the data exchange protocol used for the conveyor belt system. PROFINET is able to match most of the modern needs for information sharing between nodes and has the potential to grow as a popular choice among newer versions of hardware (Carl Henning, 2021). The lightning speed exchange of the protocol and unlimited address space for new nodes opens up ways to increase the complexity of systems and widely used Ethernet cable illuminates the needs of specific connecting hardware for flexibility of topology systems in the modern industry.

For the drive to be connected to the system through PROFINET it must have an adapter. ABB has only one product that supports ACS880-M04 industrial drive and PROFINET connectivity. The module is the ABB FENA-11 Ethernet adapter module. The company provides GSDML files that are key elements for the further implementation process of the system in the next chapter.

Figure 26. ABB FENA-11 Ethernet adapter module (abb.com)

5 Integration process

The following section will cover the integrating procedure for the nodes to be linked to the others according to the topology previously shown.

5.1 Industrial drive

Corresponding to the topology, the drive must have control over the AC motor and be able to be accessed through PROFINET protocol. The following subsection demonstrates steps for the hardware to be set-up, ready for operating and communicating with the rest of the system.

For ABB ACS880-M04 to override ABB M3AA 132M 4 electric motor, electric data for the motor must be set in the ABB drive's settings. The motor's parameters can be found in the datasheets or on the side of the motor's body. Figure 27 provides the motor's characteristics for each type of connection to the electric grid.

Electrical Data:

Conn	Temp Class	Freq	Voltage	Power	Speed	Current	Power Factor	Efficiency
Y	--	50 Hz	690 V	5.50 kW	1465 r/min	6.30 A	0.790	89.00 %
D	--	50 Hz	400 V	5.50 kW	1465 r/min	10.90 A	0.790	89.00 %
D	--	60 Hz	460 V	5.50 kW	1769 r/min	9.50 A	0.790	89.70 %

Figure 27. Electric characteristics of AVV M3AA 132M 4 (abb.com)

For the project, a star connection would be used when connecting to the grid. However, the connection type can be changed according to the application's needs. Technical and installation documentation can be found on the ABB site dedicated to the ACS880-M04 and the ABB M3AA 132M 4. After the motor is properly wired, setting for the motor can be found inside the driver's parameter list (ABB 2016). There are parameters, addresses and their values recorded in Table 8.

Drive parameter	Value
99.03 Motor type	Asynchronous motor
99.04 Motor control	DTC
99.06 Motor current	6.30 A
99.07 Motor nominal voltage	690.0 V

99.08 Motor nominal frequency	50.00 Hz
99.09 Motor nominal speed	1465 rpm
99.10 Motor nominal power	73.75 hp or 5.50 kW
23.12 Acceleration time	2.000 s
23.13 Deceleration time	7.000 s

Table 8. Parameters addresses and values

The values of acceleration and deceleration time can be tuned according to the application requirements, while others may need to be changed if different AC motor or a different type of electrical connection is used. After the settings are saved inside the drive's program, the motor is ready for operation.

For ABB ACS880-M04 industrial drive to be manipulated a PROFINET module must be installed. The physical installation only requires locating the slot on the interface board of the drive (ABB, 2018). Figure 28 is a drawing of control interfaces installed on the drive's motherboard.

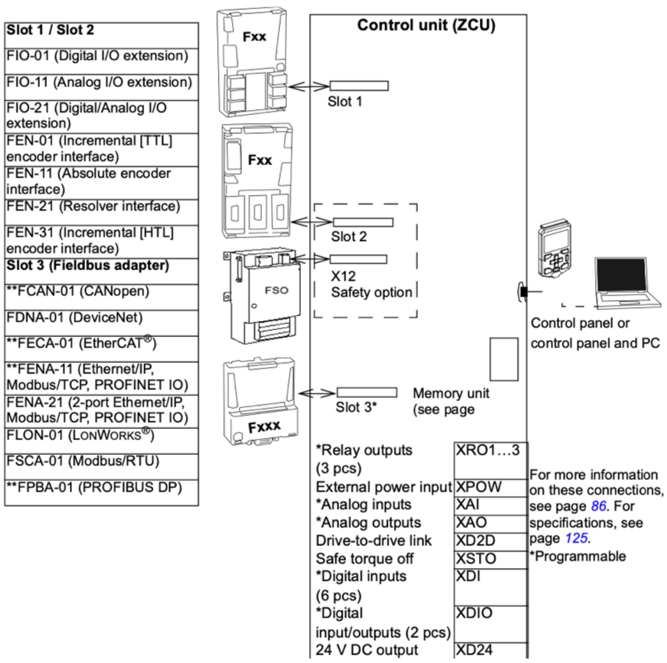


Figure 28. Control panel of ABB ACS880-M04 industrial drive

As can be seen in Figure 28, the FENA-21 PROFINET adapter fits only Slot 3 and must be installed in that particular slot to enable the protocol connectivity. After the physical installation, the drive's Fieldbus mode has to be switched on and the connection settings need to be configured. Fieldbus adapter parameters addresses and input values are listed in Table 9 for installation convenience (ABB, 2018).

Drive parameter	**Value**	**Description**
50.01 FBA A enable	**1** = Option slot 1	Enables communication between the drive and the Fieldbus adapter module
50.02 FBA A comm loss function	**1** = Fault	Enables Fieldbus A communication fault monitoring
50.03 FBA A comm loss t out	3.0 s	Defines the Fieldbus A communication break supervision time
50.04 FBA A ref1 type	**4** = Speed	Selects the Fieldbus A reference 1 type and scaling
50.07 FBA A act1 type	**0** = Auto	Selects the actual value type and scaling according to the currently active Ref1 mode defined in parameter 50.04
51.01 FBA A type	**128** = ETHERNET	Displays the type of the Fieldbus adapter module
51.02 Protocol/Profile	**10** = PNIO Pdrive	Selects the PROFINET IO protocol and the PROFIdrive profile

51.03 Commrate	**0** = Auto	Ethernet communication rate is negotiated automatically by the device
51.04 IP configuration	**0** = Static IP	
52.01 FBA data in1	**4** = SW 16bit	Status word
52.02 FBA data in2	**5** = Acf1 16 bit	Actual value 1
52.03 FBA data in3	01.14	Output power
52.05 FBA data in5	01.11	DC Voltage
53.01 FBA data out1	**1** = CW 16bit	Control word
53.02 FBA data out2	**2** = Ref1 16 bit	Reference 1 (speed)
53.03 FBA data out3	22.26	Constant speed 1
53.05 FBA data out5	22.27	Constant speed 2
51.27 FBA A par refresh	**1** = Refresh	Validates the FENA-21 configuration parameter settings
20.01 Ext1 commands	**12** = Fieldbus A	Selects the Fieldbus A interface as the source of the start and stop commands for external control location 1
22.11 Speed ref1 source	**4** = FB A ref1	Selects the Fieldbus A reference 1 as the source for speed reference 1

Table 9. Configuration of the ACS880-M04 parameters for PROFINET connectivity

After the FENA-21 parameters were saved on the drive, ACS880-M04 is set to be operated through the PROFINET with the FENA-21 adapter. However, some of the parameters' values can be changed according to the user's demands. A Full description of the parameters and documentation are available on the Internet as well as on the ABB website.

5.2 Computing unit

The upcoming chapter shows the steps for configuring the Siemens S7-1514 CPU to receive feedback from the monitoring system and the industrial drive. The process includes, installing the CPU function blocks for a user to be able to adjust the system's behaviour remotely from a configuring station and creating feedback monitoring tables. TIA Portal V15 is used as the main software used for configuring the Siemens S7-1514 CPU.

The first step is to configure the computing device with the software. That is an essential part of the chapter, since the software could run diagnostics to highlight the compatibility issues of the system. The configuration starts by creating a rail, searching through the hardware catalogue and dragging the objects onto the rail. Figure 29 shows the prepared hardware allowing the user to write a code for the system.

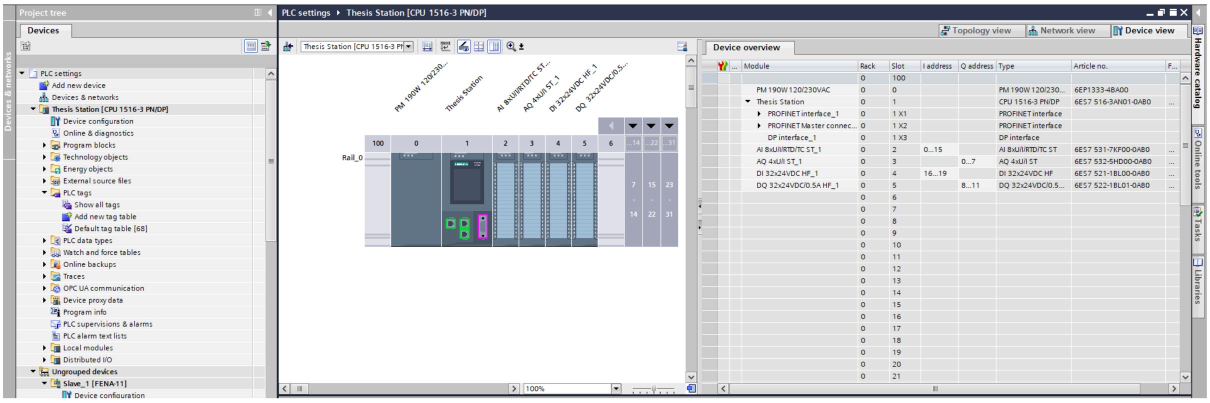


Figure 29. Prepared hardware in TIA Portal V15

Temperature and humidity values are sensed with analogue sensors, selected in the previous chapter. The analogue sensors output signals can be interpreted by using voltage or current readings. The selected sensors have an output voltage ranging from 0V…10V and the voltage range is used for comparing the signals to real values. Although, the CPU AI module's channels must be set to sense the voltage change. The settings for each of the analogue input channels can be modified to different measuring ranges. The analogue sensors selected in the previous chapter have a measuring range from 0V to 10V.

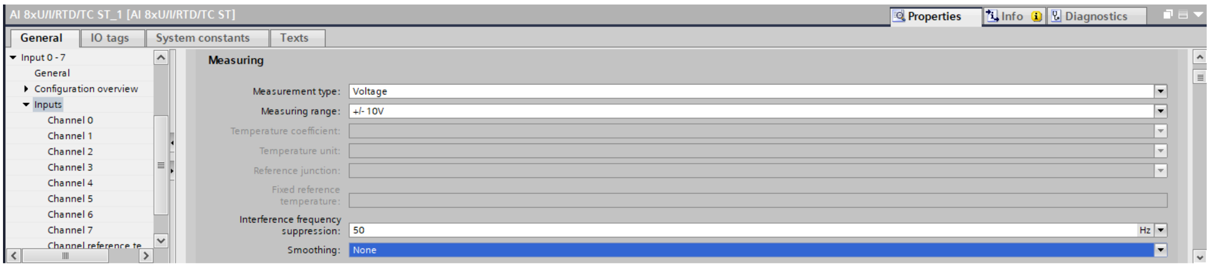


Figure 30. Analog module measuring properties

The next stage of the integration is to write logic to normalize the values from the analogue signals. The project uses three analogue signals. For further implementations, and to simplify the main program, a function block was created. The function block is made only for sensors that output signals in the voltage range and the block is capable of normalizing and scaling values from an analogue sensor to have a measurable value as well as show an error byte if the measuring data is out of range. Figure 31 shows the logic of the function block.

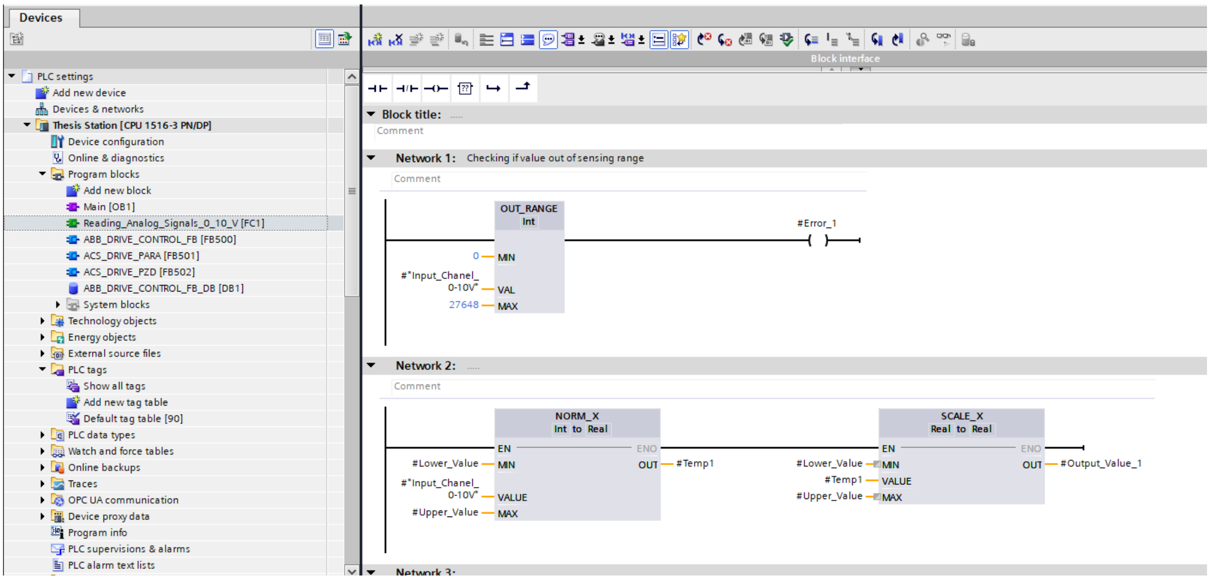


Figure 31. Function block for reading 0…10V analogue signal

For a program to understand if values are out of range, the maximum readable value must be set manually. The range values can be found in the PLC's technical documentation. Different analogue modules have different values for measuring range, as well as if a sensor sends signals via non-voltage dependent signals. In this case, the user must set the value according to the sensor type. Figure 32 is a table of values for different measuring ranges for the analogue module. The error message is made for convenience and easily highlights the issue that needs to be resolved. As a result, it decreases the time spent on a sensor after replacement.

Table C- 4 Voltage measuring ranges ±10 V, ±5 V, ±2.5 V, ±1 V,

Values		Voltage measuring range				Range
dec	hex	±10 V	±5 V	±2.5 V	±1 V	
32767	7FFF	>11.759 V	>5.879 V	>2.940 V	> 1.176 V	Overflow
32511	7EFF	11.759 V	5.879 V	2.940 V	1.176 V	Overshoot range
27649	6C01					
27648	6C00	10 V	5 V	2.5 V	1 V	Rated range
20736	5100	7.5 V	3.75 V	1.875 V	0.75 V	
1	1	361.7 µV	180.8 µV	90.4 µV	36.17 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7.5 V	-3.75 V	-1.875 V	-0.75 V	
-27648	9400	-10 V	-5 V	-2.5 V	-1 V	
-27649	93FF					Undershoot range
-32512	8100	-11.759 V	-5.879 V	-2.940 V	-1.176 V	
-32768	8000	< -11.759 V	< -5.879 V	< -2.940 V	< -1.176 V	Underflow

Figure 32. Values of the measuring range for the analogue module

Further, the block for reading analog signals is used for the main program for each of the sensors inside the monitoring system. The real values from the sensors are stored inside the memory of the computing unit. The real values represent the feedback of the system which will be stored and analyzed in the MindSphere cloud service.

In the main program is possible to set the emergency stop signal if any of the temperatures or humidity exceeded the limits, set by a user. Figure 33 shows the logic of the program applied to the humidity value. The same approach is further applied to the rest of the analogue signals.

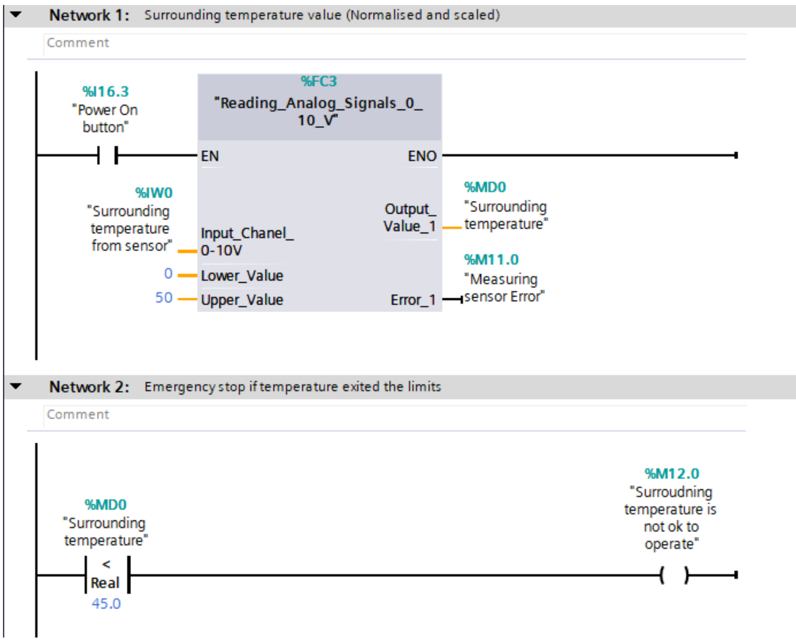


Figure 33. The logic for the humidity sensor

The performance of the conveyor belt system is monitored through a digital sensor. To evaluate the performance success of the system the main program must count the number of goods passed during a certain period of time. For the project, the productivity will be calculated each hour, starting from the beginning of the conveyor operation.

To calculate the time-dependent value, clock signals must be unlocked inside the CPU settings. The clock pulses are used in the main program to reset the counter used for tracking the number of objects. The amount of clock pulses required to do the reset equals the time in seconds. Figure 35 shows the logic that is used to analyze the performance rate of the system.

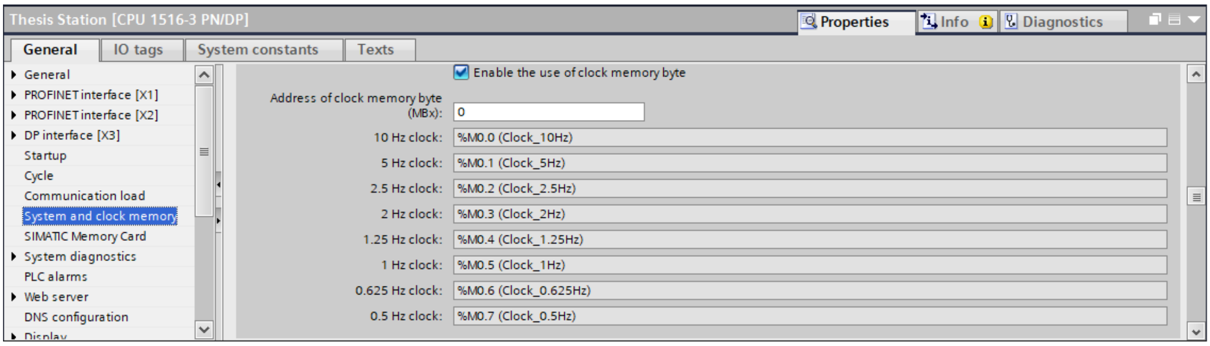


Figure 34. Clock signals settings

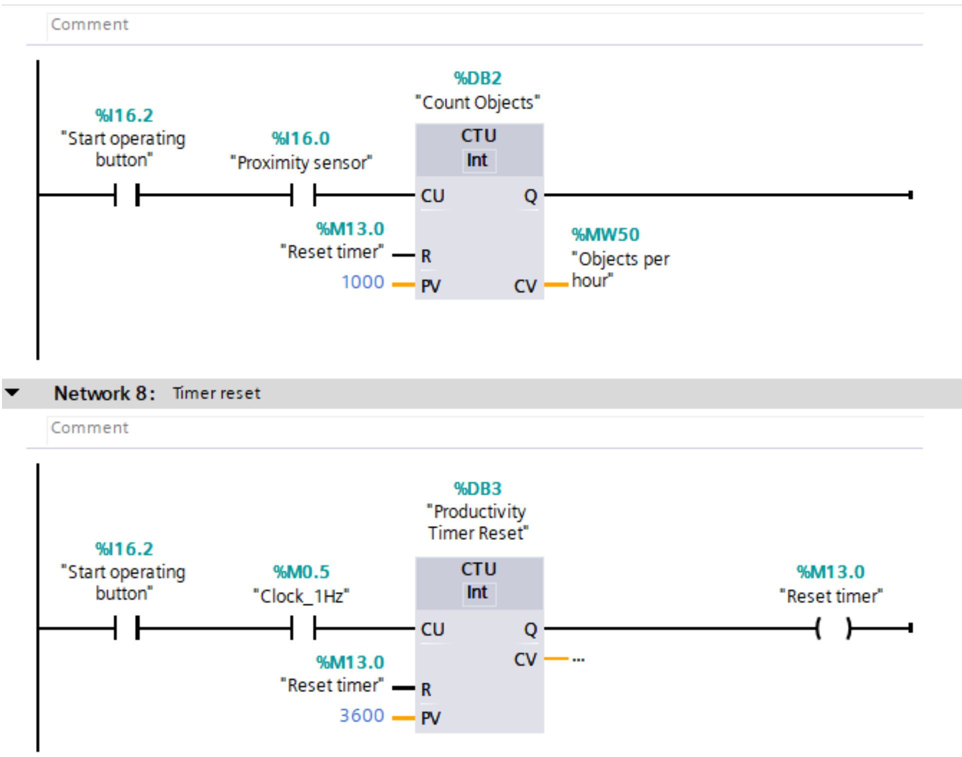


Figure 35. Logic to calculate the performance rate of the system per hour

For the CPU to control the industrial drive via PROFINET, the ABB PROFINET drive function blocks must be installed in TIA Portal. The blocks are available on the ABB site for the FENA-11 product. The blocks allow a user to monitor and adjust the operating behaviour of the drive remotely in the same software used for programming the computing unit (Siemens, 2016). Figure 36 shows the full library of the function blocks needed to proceed with the project. The blocks must be copied from the library to the list of the program blocks to be used in the project.

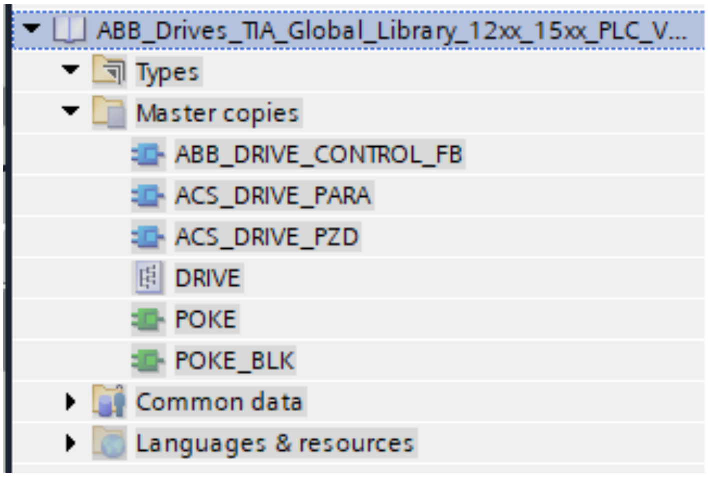


Figure 36. ABB function blocks library for TIA Portal

Further, the FENA-11 adapter must be added in the topology view and wired via a PROFINET cable. The IP addresses for the drive and the CPU must match to go on-line. The settings must be put and synced manually. Figure 37 illustrates the connected devices.

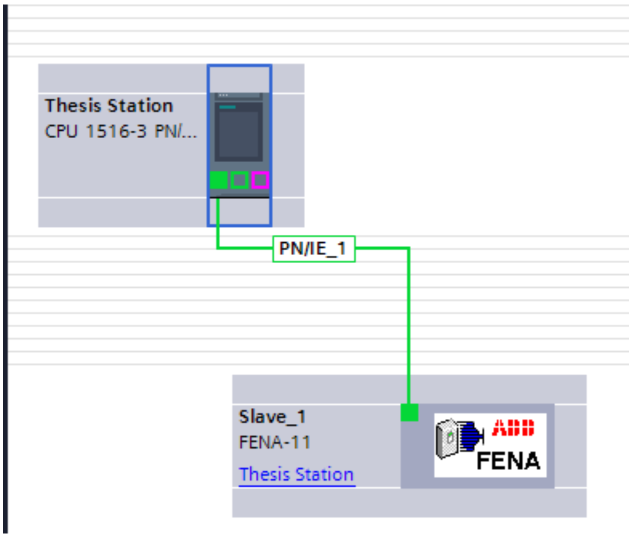


Figure 37. Connected devices via PROFINET cable in TIA Portal

The next stage is to create a program that uses the ABB function blocks to operate the drive. The main drive block is named FB500 and by simply dragging it to the network window it is ready to be programmed. Figure 38 shows the programmed drive block for the system.

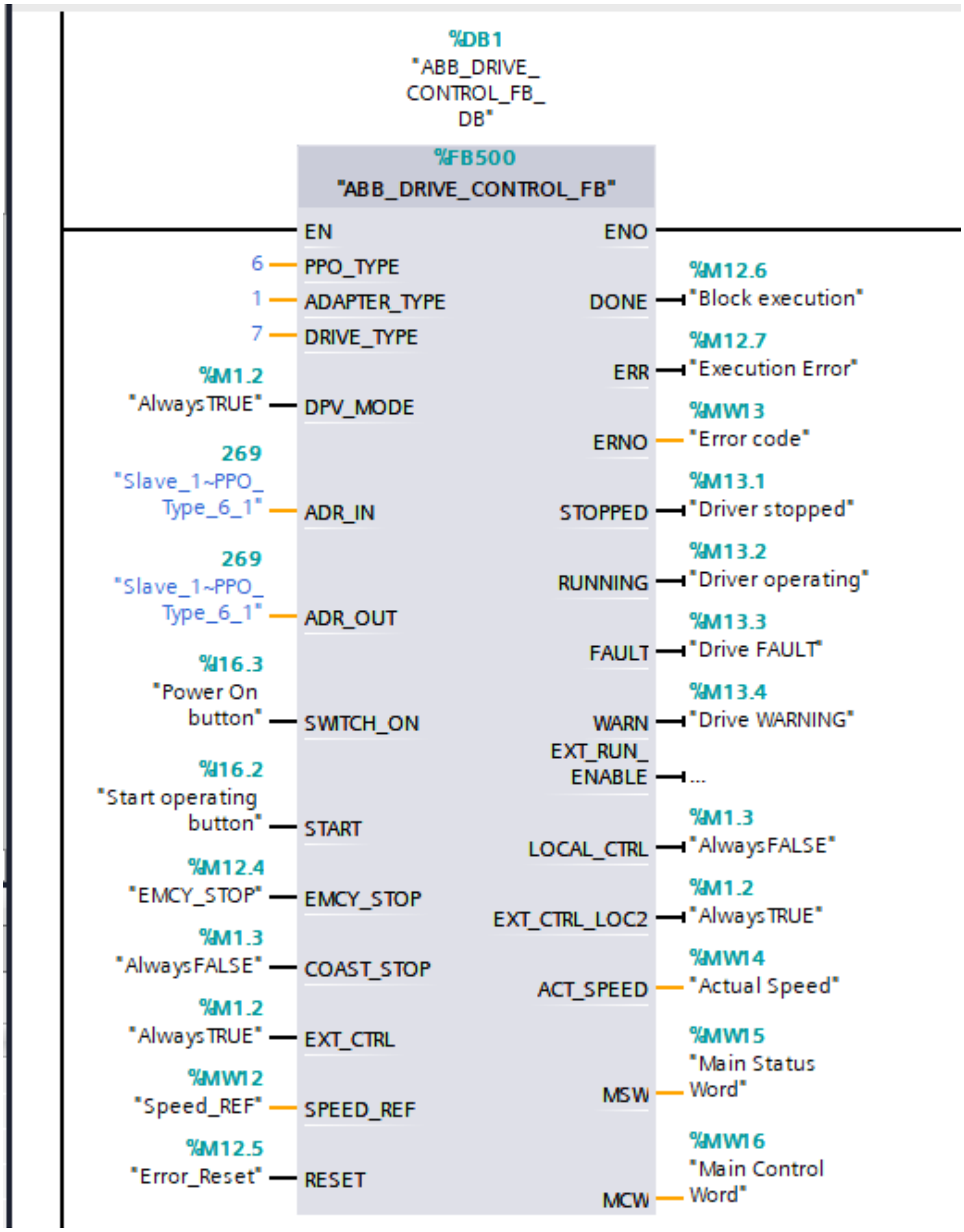


Figure 38. Programmed FB500 inside TIA Portal

Below is a table of block variables with comments for each of the input and the output signals. The output error messages help to monitor the conditions of the drive. If an error occurs during drive's operating time, a user is able to see the exact location of the error and can create a way to solve it without physically interacting with the drive (Siemens, 2016). That results in a reduced time of maintenance close to the dangerous operating area.

Block variables and data types for FB500

Block variable	Data type	Comment
EN	BOOL	Enabling block. FALSE = block code is not executed. TRUE or unconnected = block code is executed.
PPO_TYPE	INT	The PPO type. 1, 2, 3, 4, 5 or 6; 0 = not allowed.
ADAPTER_TYPE	INT	PROFIBUS module type: FPBA-01 PROFIBUS DP module connected in the drive. 1=FPBA (or FENA), 2=RPBA (or RETA).
DRIVE_TYPE	INT	Drive type: ACS800=1, ACSM1=2, ACS350=3, ACS355=4, ACS550=5, ACS850=6, ACS880=7, ACS580=8, ACS380=9.
DPV_MODE	BOOL	FALSE=DP-V0[1], TRUE=DP-V1[2] (or PROFINET).
ADR_IN	HW_IO	Hardware ID of the module from which the data is to be read. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**).
ADR_OUT	HW_IO	Hardware ID of the module to which the data is to be written. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**).
SWITCH_ON	BOOL	FALSE=Drive control switched off, TRUE=Drive control switched on. After an EMERGENCY STOP a new rising edge of SWITCH_ON is needed before next start. SWITCH_ON also needs to be active (TRUE) for resetting drive faults.
START	BOOL	FALSE=Ramp stop with deceleration time according to drive parameter, TRUE=Start. Drive start via fieldbus requires parameter setting in the drive.
EMCY_STOP	BOOL	FALSE=Emergency stop according to emergency stop deceleration time set in drive parameter, TRUE=Normal operation.
COAST_STOP	BOOL	FALSE=Normal operation, TRUE=Coast stop (drive releases control of the motor).
EXT_CTRL	BOOL	Selection of external control location EXT2. FALSE=EXT1, TRUE=EXT2. Shifting to EXT2 via fieldbus requires parameter setting in the drive.

SPEED_REF	INT	Speed reference value: -20000 to 20000. See chapter "Drive configuration" for scaling. Setting speed reference via fieldbus requires parameter setting in the drive.
RESET	BOOL	FALSE=No operation, TRUE =Reset drive fault.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	INT	Error code when ERR=TRUE, see SIMATIC online help for SFC14 or SFC15.
STOPPED	BOOL	FALSE=Drive is not stopped, TRUE=Drive is stopped.
RUNNING	BOOL	FALSE=Drive is not running, TRUE=Drive is running and following the speed reference value.
FAULT	BOOL	FALSE=No drive fault active, TRUE=Drive fault active.
WARN	BOOL	FALSE=No drive warning active, TRUE=Drive warning active.
EXT_RUN_ENABLE	BOOL	FALSE=No external run enable signal received in the drive, TRUE=External run enable signal received in the drive.
LOCAL_CTRL	BOOL	FALSE=Remote control (normal mode), TRUE=Local control (e.g. drive control panel or pc tool in local mode)
EXT_CTRL_LOC2	BOOL	Actual control place, FALSE=EXT1, TRUE=EXT2.
ACT_SPEED	INT	Drive actual speed: -20000 to 20000. See chapter "Drive configuration" for scaling.
MSW	WORD	Drive main status word. See actual fieldbus adapter manual for detailed description.
MCW	WORD	Drive main control word. See actual fieldbus adapter manual for detailed description.

Figure 39. Table of block variables, input and output signals with the description

For the FB500 system, a drive block was created for the logic with an emergency stop function for the system to reduce the risk of components being damaged if the limits of humidity, surrounding temperature or bearing operating temperature exceeded the values set by the user.

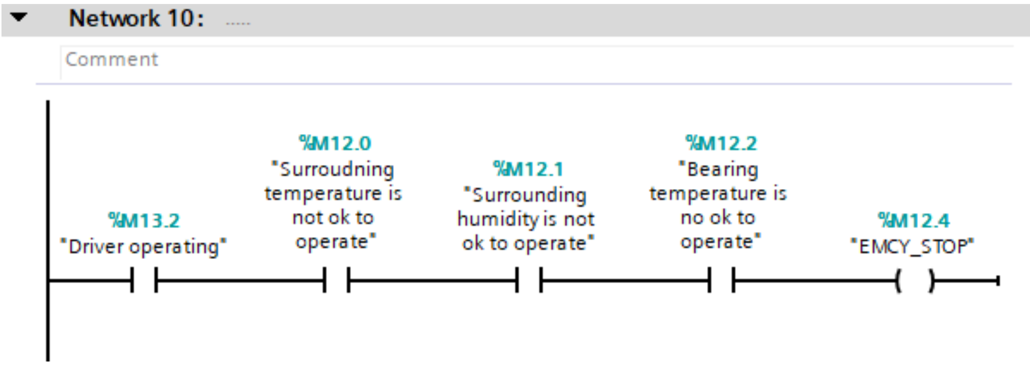


Figure 40. Emergency stop logic for the drive

Although after the step the drive is integrated, for the project it is essential to decrease the time of a user's physical manipulations or time around the machinery close to the rotating objects. Reading and writing the parameters of the drive remotely can contribute to the goal of the thesis. To allow a user to make the changes in the parameters FB501 block from the ABB drive library was used (Siemens, 2016). Figure 41 represents a modified block that enables the edits to be made remotely.

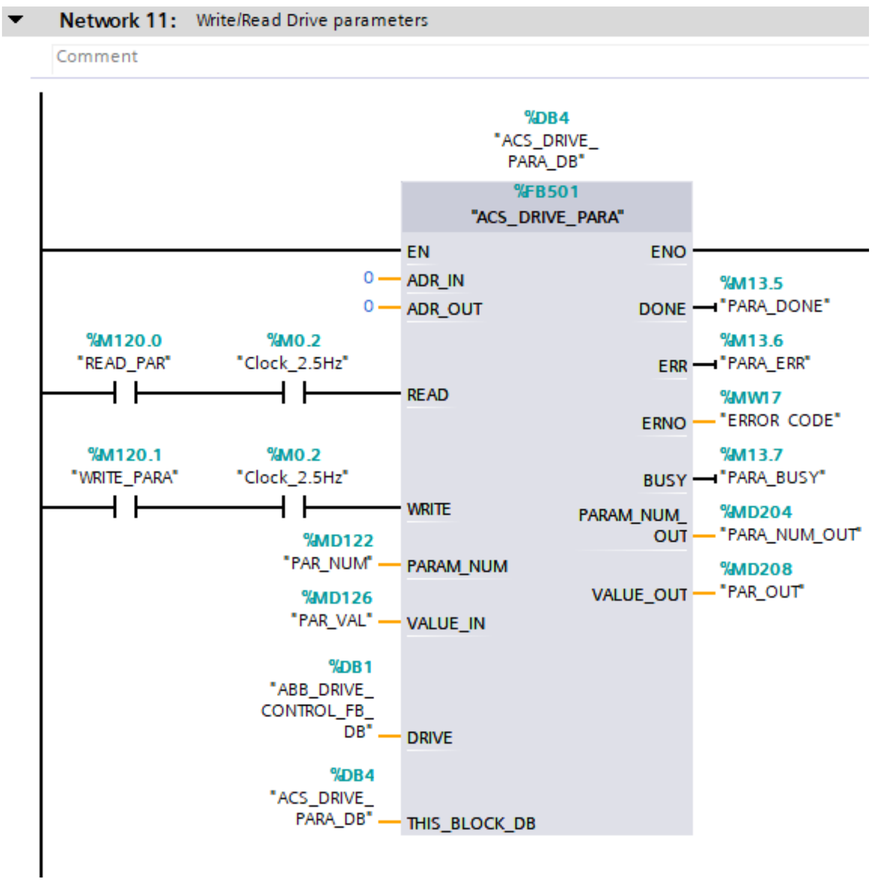


Figure 41. Set function block to remotely access the drive's parameters

Below a table of data, variables used to create the function block input and output signals were created with comments for each of the signals.

Block variables and data types for FB501

Block variable	Data type	Comment
ADR_IN	HW_IO	Hardware ID of the module from which the data is to be read. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**). • DP-V0: Hardware ID of the module must be provided and *DPV_Mode* flag should be *False* in FB500 block. • DP-V1 (or PROFINET): Hardware ID or 0 and *DPV_Mode* flag must be *True* in FB500 block.
ADR_OUT	HW_IO	Hardware ID of the module to which the data is to be written. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**). • DP-V0: Hardware ID of the module must be provided and *DPV_Mode* flag should be *False* in FB500 block. • DP-V1 (or PROFINET): Hardware ID or 0 and *DPV_Mode* flag must be *True* in FB500 block
READ	BOOL	Read the parameter value 0 -> 1 (executed on positive edge).
WRITE	BOOL	Write the parameter value 0 -> 1 (executed on positive edge).
PARAM_NUM	DINT	Read/written parameter: 3 numbers = group, 2 numbers = Index. For example, Par 20.06 = 2006.
VALUE_IN	DINT	Parameter value to be written.
DRIVE	DB_ANY	Instance Data Block. The drive variable is used for identifying to which drive FB501 *ACS_DRIVE_PARA* belongs. The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* must correspond to the variable FB501 ACS_DRIVE_PARA.
THIS_BLOCK_DB	DB_ANY	Instance data block of FB501. **Note:** POKE (FC1) and POKE_BLK (FC2) are used in FB501 block logic to exchange data between FB500 and FB501.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	WORD	Error code when ERR=TRUE.
BUSY	BOOL	FALSE=No operation active, TRUE=Operation active.
PARAM_NUM_OUT	DINT	Handled parameter number: 3 numbers = group, 2 numbers = index; for example, Par 20.06 = 2006.
VALUE_OUT	DINT	Read parameter value.

Figure 42. Table of data variables used in ABB FB501

After the function blocks are set the drive is ready to be operational. The condition of the drive can be monitored remotely, and parameters now can be adjusted through TIA Portal software. As the result, the physical interaction with the devices is reduced to maintenance work only.

5.3 MindSphere asset configuration

Before sending the feedback from the monitoring system to the cloud, the asset for the project must be configured and synced to the cloud service. The step enables secure data transfer between the system and the MindSphere cloud service.

Firstly, the asset must be onboarded in the cloud. The process requires cloud asset configuration and hardware configuration. The cloud configuration starts with accessing the cloud app launchpad and selecting the Asset Manager app. A new asset with the type of the asset to be MindConnect Nano must be created in the app. Further, a piece of basic information must be typed in the window, according to the user's personal information. After the step, the MindConnect Nano is configured to the cloud. To connect the device with the cloud to go on-line, information about the internet router IP address must be manually typed in the asset settings. The router's IP address is used for creating an onboarding key. The onboarding key is used for hardware configuration. The key must be downloaded and transferred to the MindConnect Nano device via a USB stick. Successful onboarding is indicated by a green light in the connection status block.

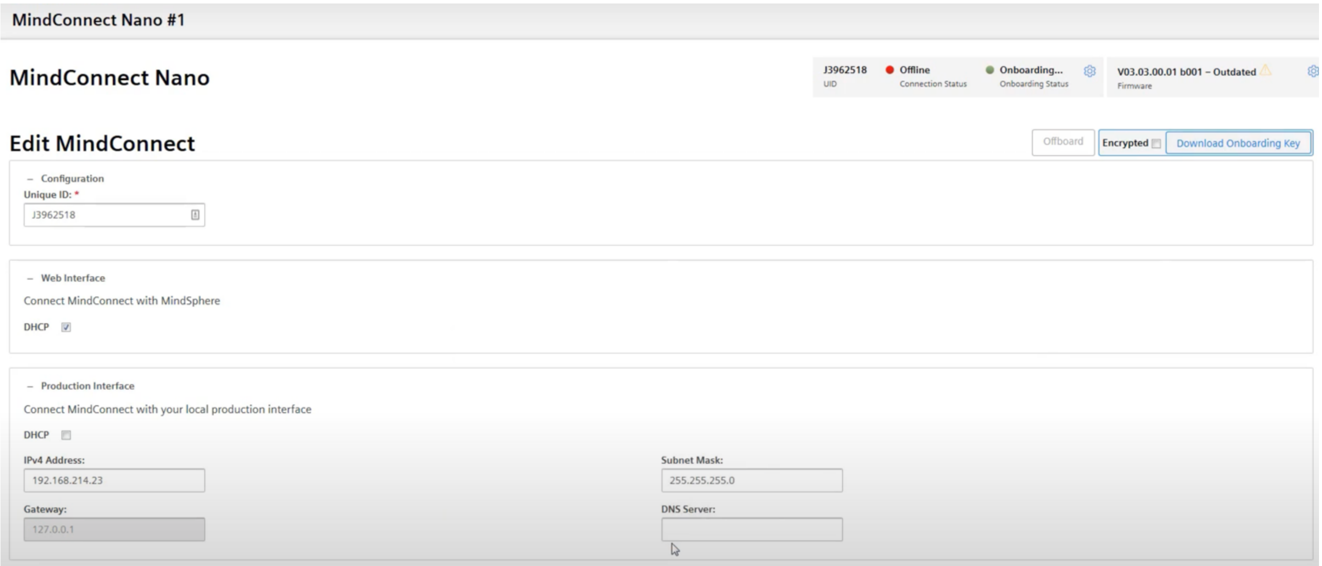
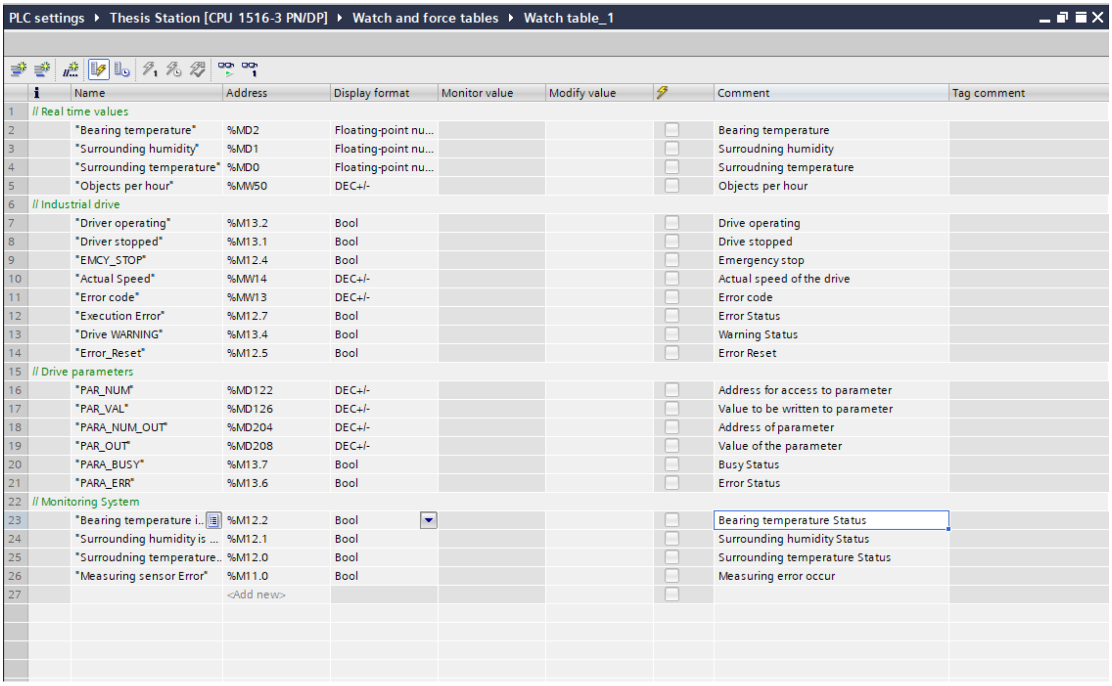


Figure 43. Configured MindConnect Nano asset ready to onboard

The next step in the integrating process is to set up the CPU unit. The unit must be connected to the MindConnect Nano through an ethernet cable. IP addresses of the connecting ports must be identical. The information regarding the IP address can be found in the cloud service, in the section relating to Ethernet for Data Acquisition. The same procedure must be done in TIA Portal for the Ethernet port used on the CPU. After the step is done, the Siemens S7-1500 CPU is ready to share the data with the MindSphere cloud service through MindConnect Nano.

Secondly, a watch table must be created in the TIA Portal software. The watch table consists of values that will be further transported to the MindSphere cloud service for further analytic work. Figure 30 shows a list of values that are monitored and sent to the cloud.



PLC settings ▸ Thesis Station [CPU 1516-3 PN/DP] ▸ Watch and force tables ▸ Watch table_1

	i	Name	Address	Display format	Monitor value	Modify value	⚡	Comment	Tag comment
1	// Real time values								
2		"Bearing temperature"	%MD2	Floating-point nu...				Bearing temperature	
3		"Surrounding humidity"	%MD1	Floating-point nu...				Surroudning humidity	
4		"Surrounding temperature"	%MD0	Floating-point nu...				Surroudning temperature	
5		"Objects per hour"	%MW50	DEC+/-				Objects per hour	
6	// Industrial drive								
7		"Driver operating"	%M13.2	Bool				Drive operating	
8		"Driver stopped"	%M13.1	Bool				Drive stopped	
9		"EMCY_STOP"	%M12.4	Bool				Emergency stop	
10		"Actual Speed"	%MW14	DEC+/-				Actual speed of the drive	
11		"Error code"	%MW13	DEC+/-				Error code	
12		"Execution Error"	%M12.7	Bool				Error Status	
13		"Drive WARNING"	%M13.4	Bool				Warning Status	
14		"Error_Reset"	%M12.5	Bool				Error Reset	
15	// Drive parameters								
16		"PAR_NUM"	%MD122	DEC+/-				Address for access to parameter	
17		"PAR_VAL"	%MD126	DEC+/-				Value to be written to parameter	
18		"PARA_NUM_OUT"	%MD204	DEC+/-				Address of parameter	
19		"PAR_OUT"	%MD208	DEC+/-				Value of the parameter	
20		"PARA_BUSY"	%M13.7	Bool				Busy Status	
21		"PARA_ERR"	%M13.6	Bool				Error Status	
22	// Monitoring System								
23		"Bearing temperature i...	%M12.2	Bool				Bearing temperature Status	
24		"Surrounding humidity is ...	%M12.1	Bool				Surrounding humidity Status	
25		"Surroudning temperature...	%M12.0	Bool				Surrounding temperature Status	
26		"Measuring sensor Error"	%M11.0	Bool				Measuring error occur	
27			<Add new>						

Figure 44. Watch table of values to be sent to MindSphere

After the completion of the watch table, the project can begin operating. The watch table further appears in the cloud service and is ready to be grouped and analyzed by MindSphere tools.

6 Summary

The thesis work covered the selection process of monitoring equipment, communication protocol and integration process of the conveyor belt system to work with the cloud service. The predetermined goals, mentioned in the framework chapter were achieved, so the IIoT conveyor belt system could be built and operational in real life.

The system was created from the initial hardware by analysing and defining their operating principles as well as learning about the IIoT network topologies and the system's operating functions for the industries. Based on the information gained, thesis goals were specified and the whole work has been split into stages to create the system.

Network development of the IIoT based conveyor belt system was accomplished with respect to modern automation pyramid and IIoT network architecture. The developed topologies were described and shown for readers to have a better understanding of the proposed system on the first stage of the development. The system's network was then used for the following chapters to select the monitoring equipment and the data communication protocol.

The integration process for the conveyor belt system helped to present the instruction for each of the devices, that enable creation of the system to be properly linked with each other. The instruction covers the step-by-step procedure of connecting the devices according to the logical network and preparing the asset in the MindSphere cloud service to receive the feedback information from the monitoring system.

The completed thesis paper is a documentation of the proposed solution of digitalising actions for conveyor belt systems. The paper also has sufficient information that allows future studies and developments to be done on the proposed system.

References

D.V. Subba Rao. 2020. The belt conveyor: A concise Basic Course. Retrieved on 8 March 2021. Available at https://www.taylorfrancis.com/books/mono/10.1201/b22610/belt-conveyor-subba-rao

Wikipedia. 2021a. Conveyor belt. Retrieved on 8 March 2021. Available at https://en.wikipedia.org/wiki/Conveyor_belt

G. Lodewijks, Xiaoli Jiang, Wenfeng Li. 2016. An Application of the IoT in Conveyor Belt Systems. Retrieved on 12 March 2021. Available at https://www.researchgate.net/publication/308391010_An_Application_of_the_IoT_in_Belt_Conveyor_Systems

Oscar Carlsson. 2017. Engineering of IoT Automation Systems. Retrieved on 11 March 2021. Available at https://www.diva-portal.org/smash/get/diva2:1060783/FULLTEXT01.pdf

Davide Barbanti. 2017. Difference between drum motors and gear motors for food conveying. Retrieved on 12 March 2021. Available at https://www.motioncontroltips.com/difference-drum-motors-gear-motors-food-conveying/

Joseph J. Carr, John M. Brown. 1998. Introduction to Biomedical Equipment Technology, 3rd Edition. Retrieved on 13 March 2021. Available at https://www.ni.com/fi-fi/innovations/white-papers/13/sensor-terminology.html#section--537752733

Omega. 2019. RTD vs Thermocouple. Retrieved on 14 March 2021. Available at https://www.omega.com/en-us/resources/rtd-vs-thermocouple

Wikipedia. 2021b. Internet of Things. Retrieved on 16 March 2021. Available at https://en.wikipedia.org/wiki/Internet_of_things

Amrutha Varshini. 2021. What is the Difference between Analog and Digital Signal Processing. Retrieved on 17 March 2021. Available at https://www.circuitbread.com/ee-faq/analog-versus-digital-signal-processing

Wikipedia. 2021c. Programmable logic controller. Retrieved on 18 March 2021. Available at https://en.wikipedia.org/wiki/Programmable_logic_controller

Wikipedia. 2021d. Fourth Industrial Revolution. Retrieved on 19 March 2021. Available at https://en.wikipedia.org/wiki/Fourth_Industrial_Revolution

Carl Henning. 2021. The Difference Between PROFIBUS and PROFINET. Retrieved on 21 March 2021. Available at https://us.profinet.com/the-difference-between-profibus-and-profinet/

Siemens. 2016. ABB Drives function blocks for Siemens. Retrieved on 22 March 2021. Available at https://library.e.abb.com/public/8e4a4a75677e4b04a828059e684a672a/ABB_Drives_function_blocks_for_Siemens_PLCs_Quick_start-up_guide_A_A4.pdf

ABB. 2018. FENA-01/-11/-21 Ethernet Adapter Module. Retrieved on 22 March 2021. Available at https://library.e.abb.com/public/d6a31af1808043ba8ef53980031041ea/EN_FENA01_11_21_UM_E_A4.pdf

ABB. 2016. Hardware manual ACS880-M04. Retrieved on 22 March 2021. Available at https://www.galco.com/techdoc/abbi/acs880-m04-014a-5_hm.pdf

Jakes Mantle. 2019. The 5 Layer of Automation Pyramid and Manufacturing Operations Management. Retrieved on 5 April 2021. Available at https://www.syspro.com/blog/erp-for-manufacturing/the-5-layers-of-the-automation-pyramid-and-manufacturing-operations-management/

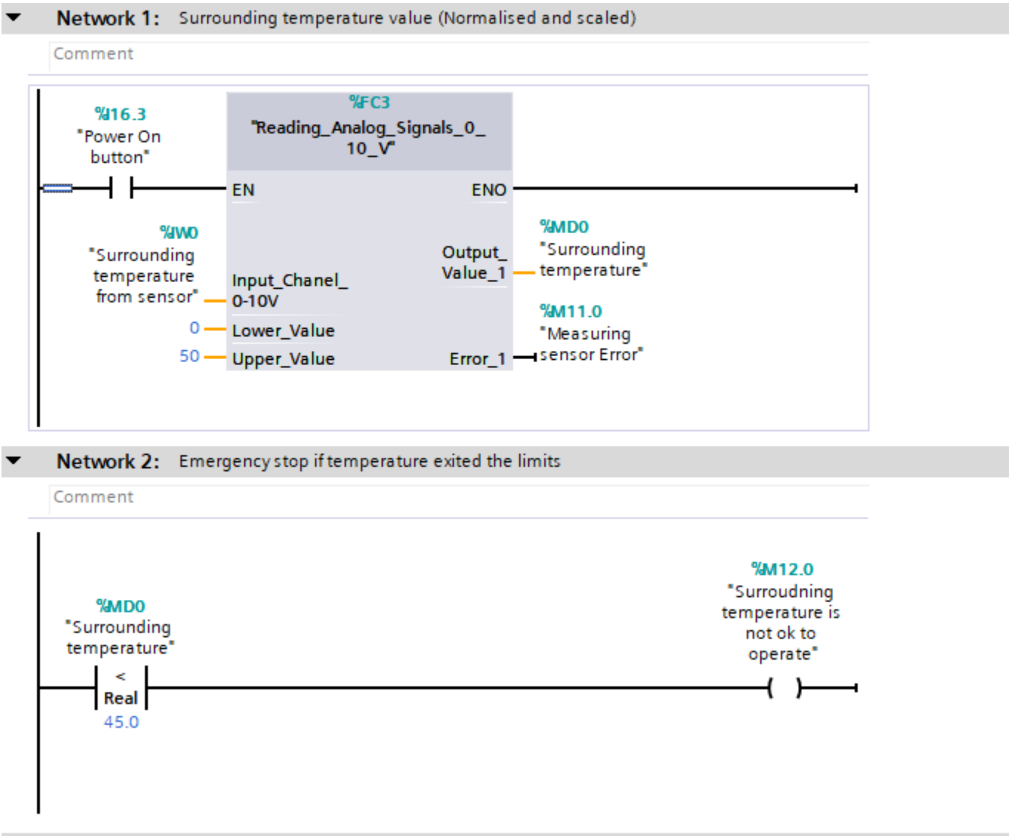
ELPROCUS. Basics of Phase AC Induction Motor. Retrieved on 7 April 2021. Available at https://www.elprocus.com/three-phase-ac-induction-motor-control-using-svpwm/

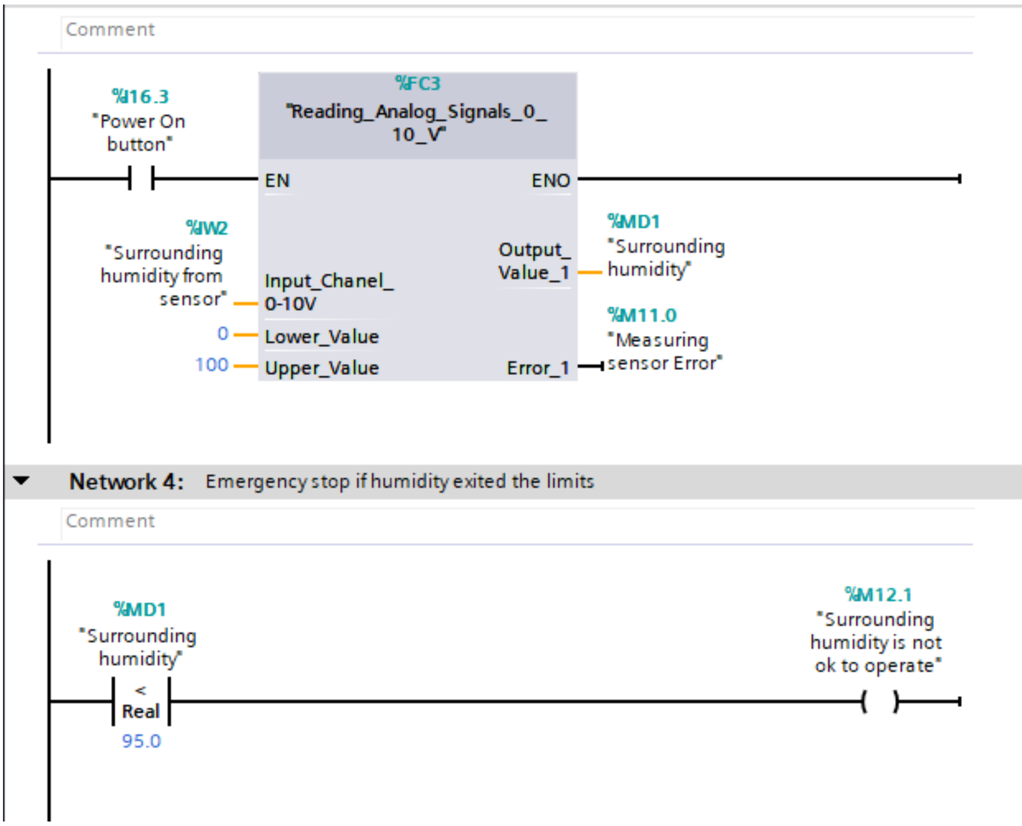
CurcuitGlobe. Difference Between Star and Delta Connection. Retrieved on 7 April 2021. Available at https://circuitglobe.com/difference-between-star-and-delta-connection.html

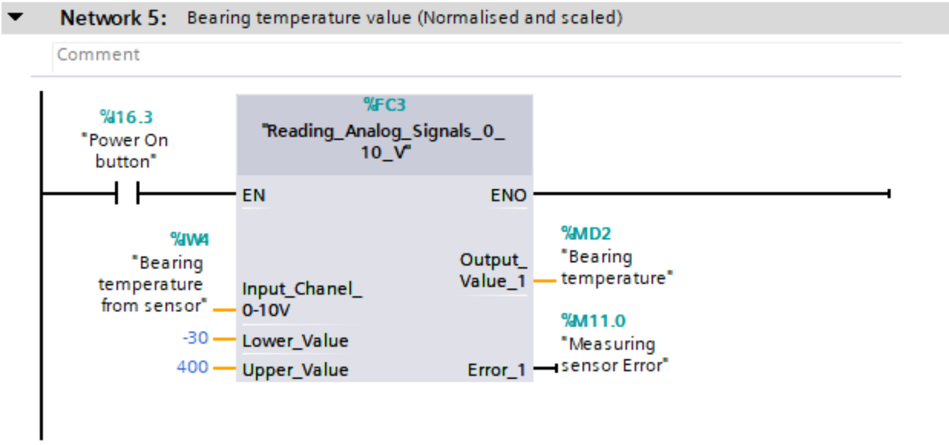
Altexsoft. 2020. IoT Architecture: the Pathway from Physical Signals to Business Decisions. Retrieved on 8 April 2021. Available at https://www.altexsoft.com/blog/iot-architecture-layers-components/

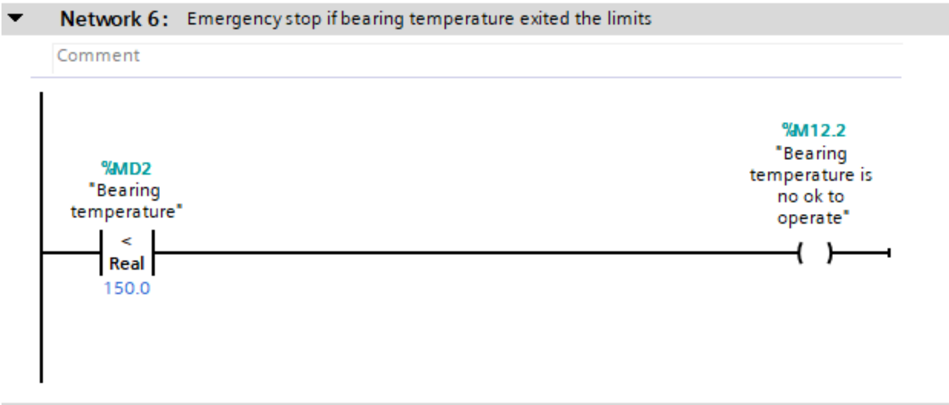
Appendices

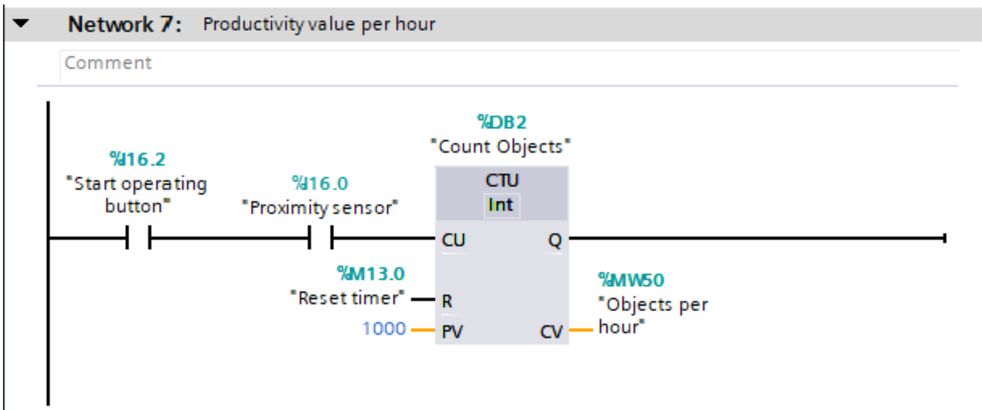
Appendix 1. Overview of the PLC program

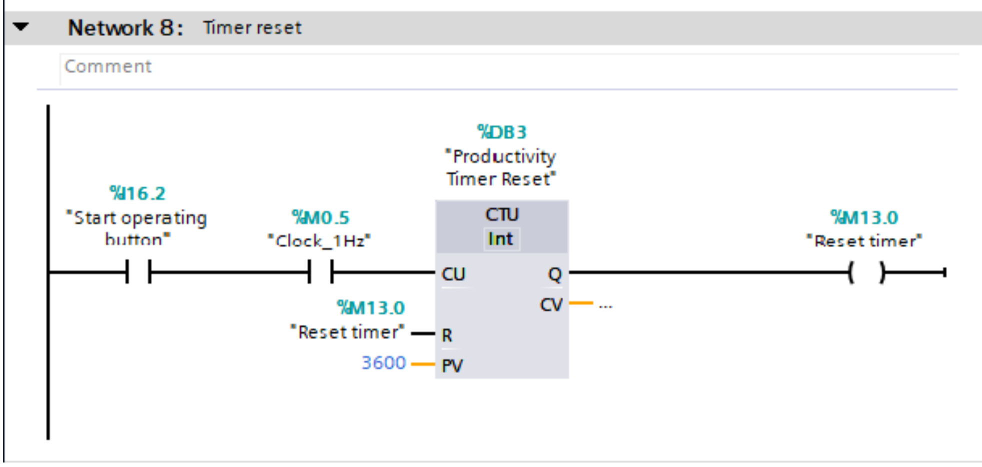


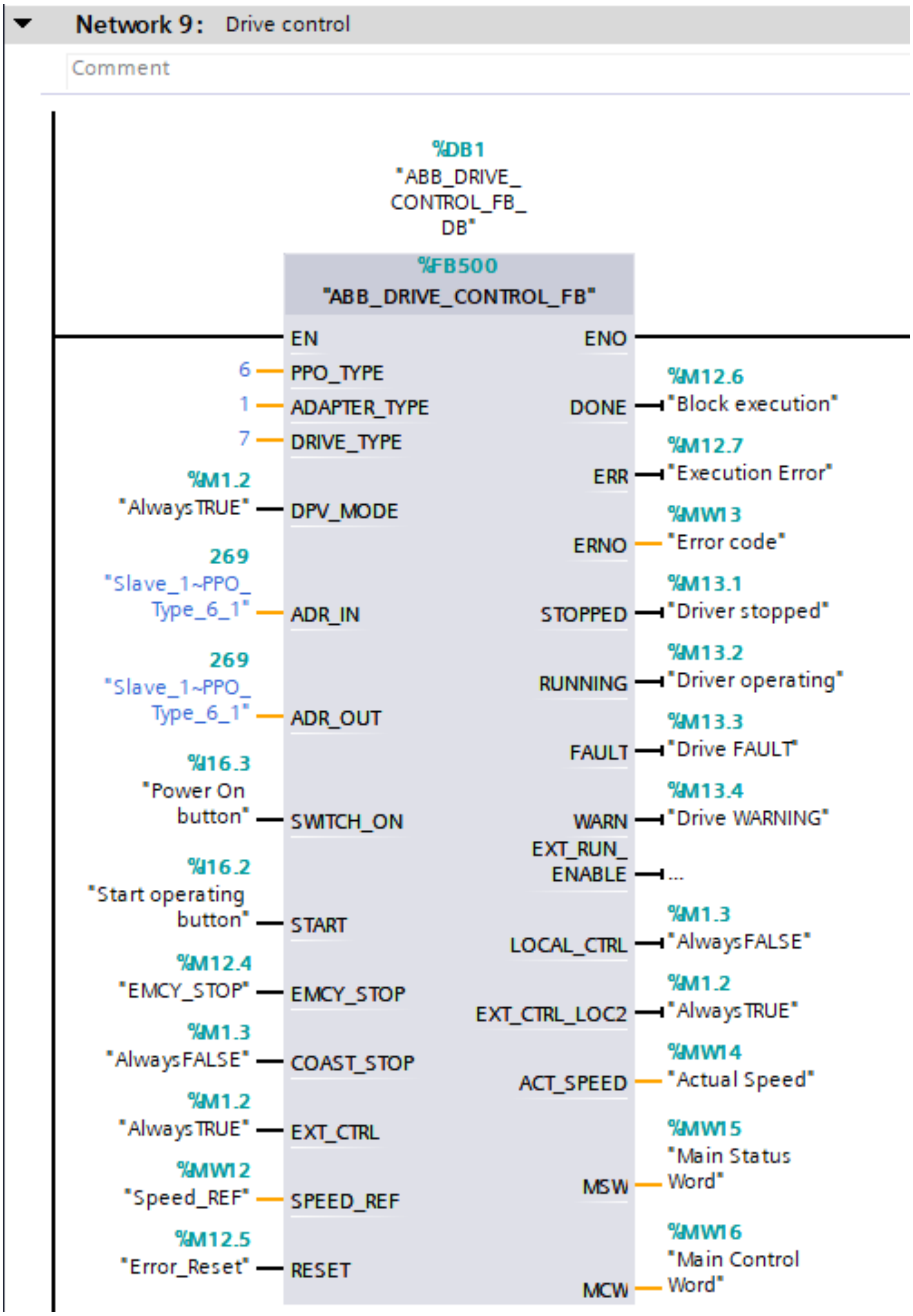


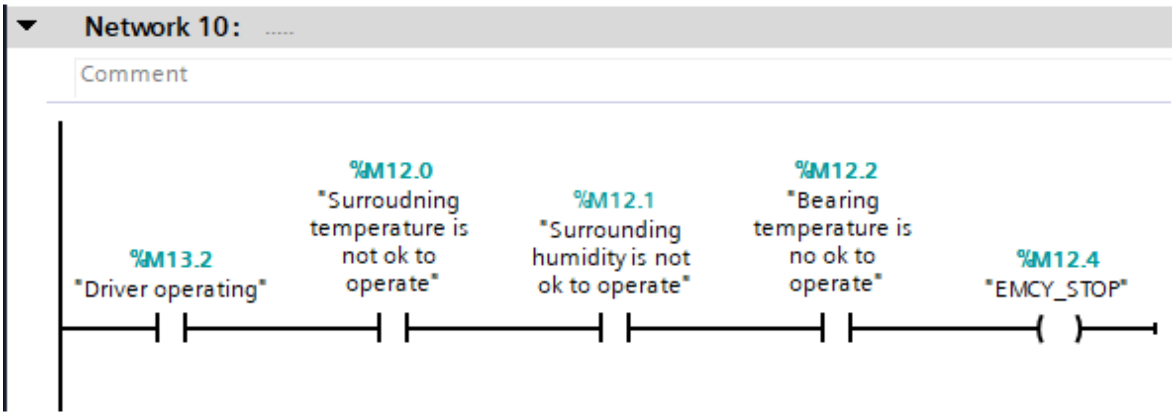

Network 5: Bearing temperature value (Normalised and scaled)
Comment
%I16.3
"Power On button"
%FC3
"Reading_Analog_Signals_0_10_V"
EN
ENO
%IW4
"Bearing temperature from sensor"
Input_Chanel_0-10V
-30
Lower_Value
400
Upper_Value
Output_Value_1
%MD2
"Bearing temperature"
Error_1
%M11.0
"Measuring sensor Error"

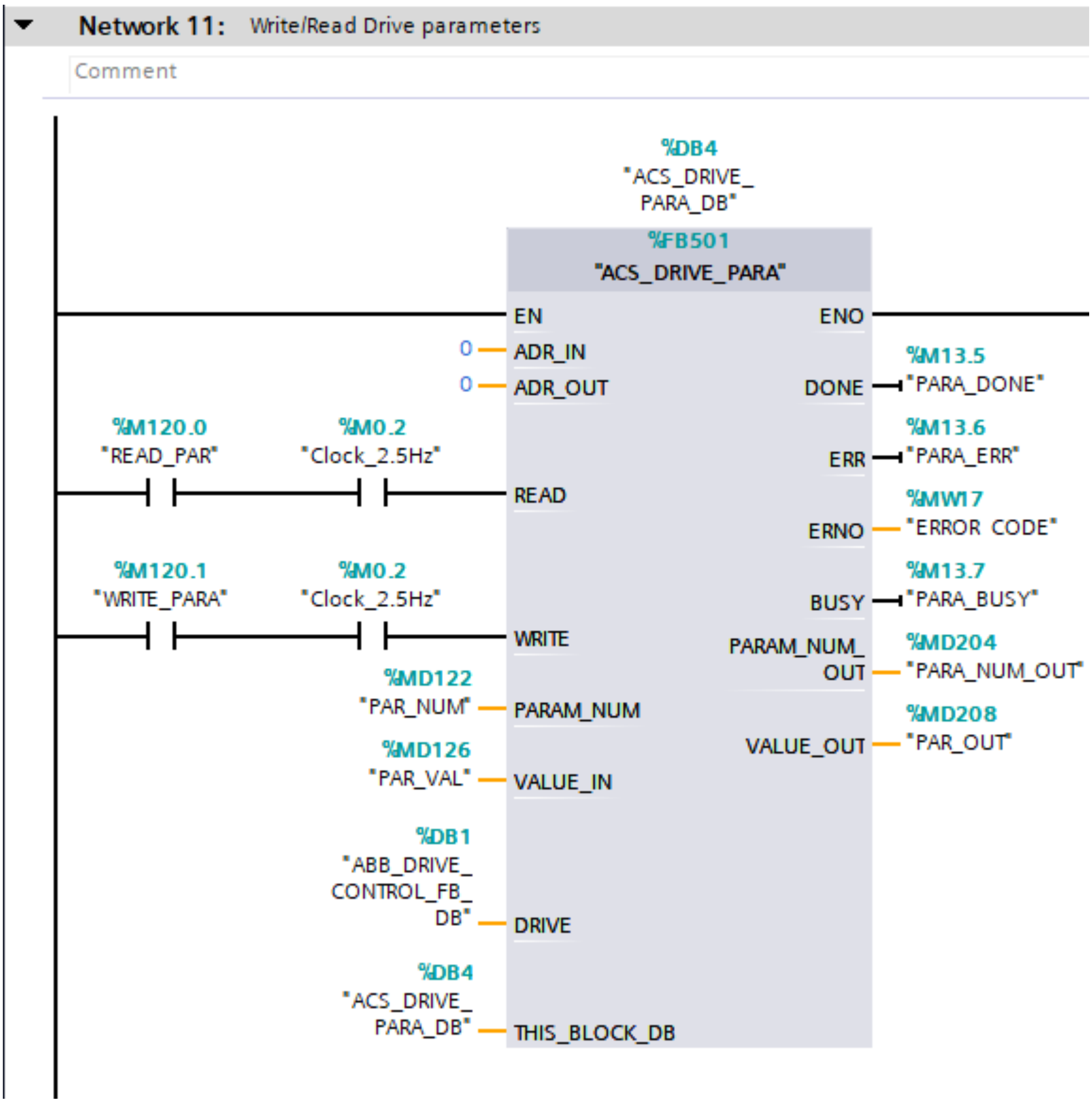

Network 6: Emergency stop if bearing temperature exited the limits
Comment
%MD2
"Bearing temperature"
<
Real
150.0
%M12.2
"Bearing temperature is no ok to operate"


Network 7: Productivity value per hour
Comment
%I16.2
"Start operating button"
%I16.0
"Proximity sensor"
%DB2
"Count Objects"
CTU
Int
CU
Q
%M13.0
"Reset timer"
R
1000
PV
CV
%MW50
"Objects per hour"


Network 8: Timer reset
Comment
%I16.2
"Start operating button"
%M0.5
"Clock_1Hz"
%DB3
"Productivity Timer Reset"
CTU
Int
CU
Q
CV
...
%M13.0
"Reset timer"
R
3600
PV
%M13.0
"Reset timer"


Network 9: Drive control
Comment
%DB1
"ABB_DRIVE_CONTROL_FB_DB"
%FB500
"ABB_DRIVE_CONTROL_FB"
EN
ENO
6
PPO_TYPE
1
ADAPTER_TYPE
7
DRIVE_TYPE
%M1.2
"AlwaysTRUE"
DPV_MODE
269
"Slave_1~PPO_Type_6_1"
ADR_IN
269
"Slave_1~PPO_Type_6_1"
ADR_OUT
%I6.3
"Power On button"
SWITCH_ON
%I6.2
"Start operating button"
START
%M12.4
"EMCY_STOP"
EMCY_STOP
%M1.3
"AlwaysFALSE"
COAST_STOP
%M1.2
"AlwaysTRUE"
EXT_CTRL
%MW12
"Speed_REF"
SPEED_REF
%M12.5
"Error_Reset"
RESET
%M12.6
DONE
"Block execution"
%M12.7
ERR
"Execution Error"
%MW13
ERNO
"Error code"
%M13.1
STOPPED
"Driver stopped"
%M13.2
RUNNING
"Driver operating"
%M13.3
FAULT
"Drive FAULT"
%M13.4
WARN
"Drive WARNING"
EXT_RUN_ENABLE
...
%M1.3
LOCAL_CTRL
"AlwaysFALSE"
%M1.2
EXT_CTRL_LOC2
"AlwaysTRUE"
%MW14
ACT_SPEED
"Actual Speed"
%MW15
MSW
"Main Status Word"
%MW16
MCW
"Main Control Word"


Network 10: .....
Comment
%M13.2
"Driver operating"
%M12.0
"Surroudning temperature is not ok to operate"
%M12.1
"Surrounding humidity is not ok to operate"
%M12.2
"Bearing temperature is no ok to operate"
%M12.4
"EMCY_STOP"


Network 11: Write/Read Drive parameters
Comment
%DB4
"ACS_DRIVE_PARA_DB"
%FB501
"ACS_DRIVE_PARA"
EN
ENO
0
ADR_IN
0
ADR_OUT
DONE
%M13.5
"PARA_DONE"
%M120.0
"READ_PAR"
%M0.2
"Clock_2.5Hz"
READ
ERR
%M13.6
"PARA_ERR"
ERNO
%MW17
"ERROR CODE"
%M120.1
"WRITE_PARA"
%M0.2
"Clock_2.5Hz"
WRITE
BUSY
%M13.7
"PARA_BUSY"
PARAM_NUM_OUT
%MD204
"PARA_NUM_OUT"
%MD122
"PAR_NUM"
PARAM_NUM
VALUE_OUT
%MD208
"PAR_OUT"
%MD126
"PAR_VAL"
VALUE_IN
%DB1
"ABB_DRIVE_CONTROL_FB_DB"
DRIVE
%DB4
"ACS_DRIVE_PARA_DB"
THIS_BLOCK_DB