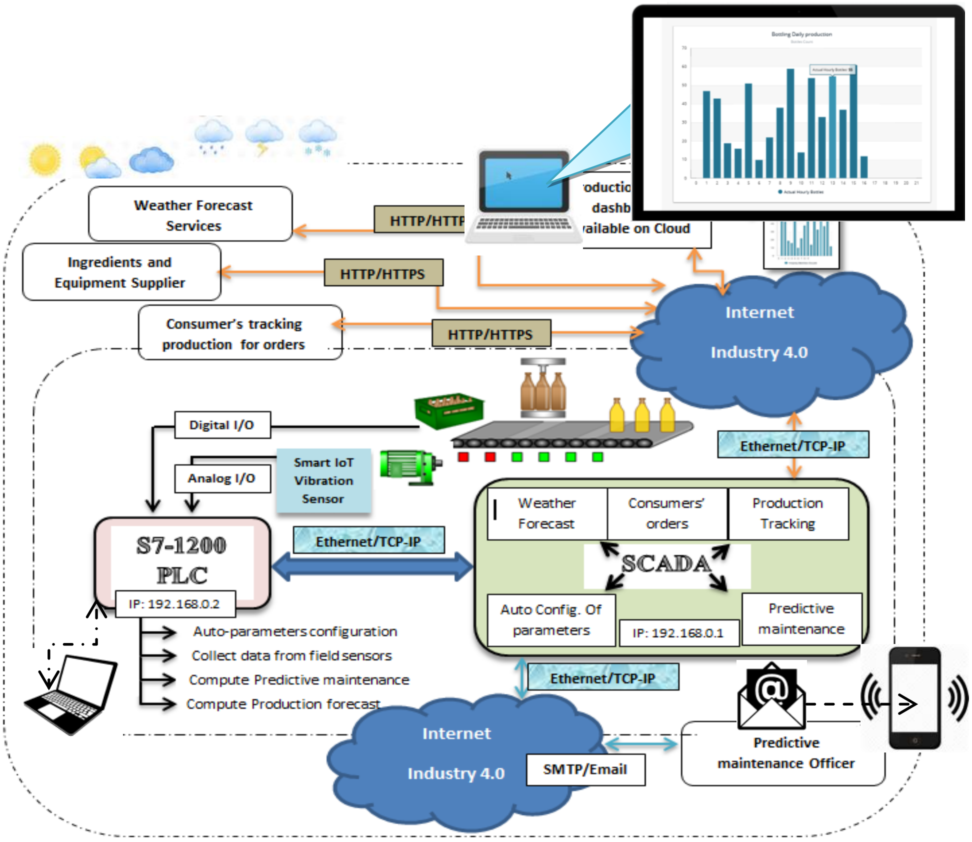


HIGH TECH AUTOMATED BOTTLING PROCESS FOR SMALL TO MEDIUM SCALE ENTERPRISES USING PLC, SCADA AND BASIC INDUSTRY 4.0 CONCEPTS

DISSERTATION

Sonia KIANGALA

HIGH TECH AUTOMATED BOTTLING PROCESS FOR SMALL TO MEDIUM SCALE ENTERPRISES USING PLC, SCADA AND BASIC INDUSTRY 4.0 CONCEPTS

by

KAHIOMBA SONIA KIANGALA

submitted in accordance with the requirements
for the degree of

MAGISTER TECHNOLOGIAE

in the subject of

ENGINEERING: ELECTRICAL

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: Prof Z Wang

August 2018

DECLARATION AND COPYRIGHT

I declare that *High tech automated bottling process for small to medium scale enterprises using PLC, SCADA and basic Industry 4.0 concepts* is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

I further declare that I have not previously submitted this work, of part of it, for examination at UNISA for another qualification or at any other higher education institution.

KS KIANGALA DATE

_________________ 14.08.2018

ACKNOWLEDGMENTS

I would like to thank my wonderful husband Minerve Mampaka for his support and advices throughout this study. Minerve, you have been a great motivation for me. I am equally grateful to my supervisor Professor Zenghui Wang for all his inputs and guidance. It has been a long journey and we made it until the end.

ABSTRACT

The automation of industrial processes has been one of the greatest innovations in the industrial sector. It allows faster and accurate operations of production processes while producing more outputs than old manual production techniques. In the beverage industry, this innovation was also well embraced, especially to improve its bottling processes. However it has been proven that a continuous optimization of automation techniques using advanced and current trend of automation is the only way industrial companies will survive in a very competitive market. This becomes more challenging for small to medium scale enterprises (SMEs) which are not always keen in adopting new technologies by fear of overspending their little revenues. By doing so, SMEs are exposing themselves to limited growth and vulnerable lifecycle in this fast growing automation world. The main contribution of this study was to develop practical and affordable applications that will optimize the bottling process of a SME beverage plant by combining its existing production resources to basic principles of the current trend of automation, Industry 4.0 (I40). This research enabled the small beverage industry to achieve higher production rate, better delivery time and easy access of plant information through production forecast using linear regression, predictive maintenance using speed vibration sensor and decentralization of production monitoring via cloud applications. The existing plant Siemens S7-1200 programmable logic controller (PLC) and ZENON supervisory control and data acquisition (SCADA) system were used to program the optimized process with very few additional resources. This study also opened doors for automation in SMEs, in general, to use I40 in their production processes with available means and limited cost.

Keywords: Bottling process; Industry 4.0 (I40); Small to medium enterprises (SMEs); Production forecast; Predictive maintenance; Production tracking; Decentralization; Interconnection; S7-1200 PLC; SCADA

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	**Definitions**
AC	Alternating Current
AP	Access Point
API	Application Programming Interface
BAS	Building Automation System
CC	Cloud Computing
CPS	Cyber Physical Systems
CPSS	Cyber Physical Production Systems
CPU	Central Processing Unit
CSV	Comma-Separated Value
DC	Direct Current
DCS	Distributed Control System
ERP	Enterprise Resource Planning
ES	Embedded Systems
FA	Factory Automation
FBD	Function Block Diagram
GUI	Graphical User Interface
HMI	Human Machine Interface
HTML	Hyper Text Mark-up Language
IEC	International Electro-technical Commission
I/O	Inputs and Outputs
I40	Industry 4.0
IASs	Industrial Automation Systems
IBWA	International Bottled Water Association
ICT	Information and Communication Technologies
IL	Instruction List
IoT	Internet Of Things
IP	Internet Protocol
ISO	International Standards Organization
IT	Information Technology
KPI	Key Performance Indicators

LAD	Ladder Logic
M2M	Machine-to-Machine
N.O	Normally Open
N.C	Normally Closed
OB	Organization Block
OEE	Overall Equipment Effectiveness
PA	Process Automation
PC	Personal Computer
PLC	Programmable Logic Controller
RFID	Radio Frequency Identification
RS232	Recommended Standard number 232 for serial communication
RTU	Remote Terminal Unit
SA	Substation Automation
SCADA	Supervisory Control And Data Acquisition
SFC	Sequential Function Chart
SMEs	Small to Medium Enterprises
SQL	Structured Query Language
ST	Structured Text
STL	Statement List
TCP/IP	Transmission Control Protocol/Internet Protocol
WLAN	Wireless Local Area Network

Chapter 1: INTRODUCTION

1.1 Background

The industrial sector is closely but surely moving toward the “fourth industrial revolution”, popularly known as Industry 4.0 (I40) or Internet of Things (IoT). This new paradigm introduced after the first three major industrial revolutions, is expected to bring more benefits for the whole society. Digitization of manufacturing enterprises, connectivity of devices and distributed intelligence being some of the key characteristics of I40, manufacturers throughout the world are already facing the enormous pressure to prepare for the integration of these new concepts within their production processes and standardize to its requirements in order to remain competitive in an upcoming era of smart factories. The challenge faced by manufacturers in small to medium enterprises (SMEs) becomes more critical because most of them do not always have the requested capital to adopt new technologies. In the bottling industry, many SMEs compete against each other to offer better services and products to end-users. They gladly invest on “cheaper” automation techniques that provide immediate answers to their needs usually neglecting the future proof aspect of the overall solution. These automation techniques are now seen as obsolete failing to standardize to I40 standards.

There is therefore an urge to develop an affordable and sustainable automation solution for SMEs that both solves their direct need, offering innovative methods to improve production, and provides an opening to the adoption of I40, and other future trend of automation.

1.2 Problem statement

Various studies have shown that most SMEs have a common perception that advanced automation techniques, specialized software like Enterprise Resource Planning (ERP), is best suited for large scale industries due to high cost of ownership, complexity of implementation and subsequent maintenance cost commonly known as “Big White Elephant” (Harpreet S.D, 2017). They therefore prefer holding onto old and sometimes archaic production methods in their everyday processes hoping to sustain production cost.

It was later proven that this lethargy in adopting current trends of automation is one of the main reasons why most SMEs are subject to low growth, high failure rates and loss of production in long term. Upkar S.A (Upkar S.A, 2014) says that while technology is essential for growth of a company; it does come at a price, which is sometimes not affordable for an SME.

While many options become unattractive because of the heavy investments they need, thus preventing small start-ups from entering, many start-ups find themselves lagging behind since they can’t have access to cutting edge technology, leaving them strategically exposed. The reality is that few companies have the necessary systems and capital in place to make leaps such as these in their operational processes, and find themselves presented with substantial barriers with respect to access.

Due to the vast scope of the technologies and methodologies, and substantial costs involved and lack of understanding and competence with advanced manufacturing techniques, at the employee level (Adolph, S., 2014). As per Upkar S.A et al. (Upkar S.A, et al., 2014) limited capital availability and lack of expertise, especially in country in development like in most African countries, are the main factors affecting modernisation and expansion plans for an SME. Since the initial years of business do not produce enough cash flows, the available cash is used up in operating activities, and there is shortage of funds for modernisation and expansion.

To remain competitive in a continuously growing market, the challenges of any industrial enterprise is to enhance product quality, reduce cost and improve on-time delivery. Advanced automation techniques, new strategies, upgrade trend of automation are developed to help companies fight this challenge and satisfy customers’ demand. While the automation sector is undergoing its fourth revolution, known as Industry 4.0 (I40) or “smart factory” which is about digital transformation and the use of Cyber Physical Production Systems (CPPS), many SMEs are still using first, second or third revolution in their production systems.

Compared to the rest of the world, the current adoption and impact of industry 4.0 on the African continent remains low. However, it is a topic that is increasingly being acknowledged and discussed by industry leaders and policy makers, mainly because of the impact smart technologies can make at a socio-economic level. The biggest challenges in Africa/South Africa remain connectivity and accessibility (Deloitte South-Africa, 2016).

With a worldwide population growth rate of 1.09% in 2018, as per the United Nation Department of Economic and Social Affairs (UN Department of Economic and Social Affairs, 2018), the beverage production industry is one of those that will have to optimize its processes in order to respond efficiently to an higher demand. The current average worldwide population increase is estimated to about 83 million a year.

The purpose of this study is to introduce small to medium bottling industries to the use of current trend of automation and improve their operations at low cost, using available means.

1.3 Research goals

The main objective of this research is to develop a low cost and efficient high tech bottling process using advanced automation techniques and current trend of automation, Industry 4.0. Advanced techniques and methods used in this research will offer tips to other bottling SMEs to start integrating the current trend of automation in their production processes using available means.

This research will therefore focus in:

- Improving production output by forecasting weekly production and ordering ahead, via the internet, production stakeholders to reduce stock shortage while demand is high.

- Minimizing production downtime by initiating predictive maintenance of conveyor motors and sending email notification directly, real-time and anytime, to maintenance experts as soon as a maintenance schedule is generated.

- Developing decentralized monitoring of the overall system performance by implementing a cloud-based intelligent dashboard accessible anywhere through the internet.

- Minimizing human intervention in system's operation and improve accuracy by developing auto-parameters configuration programs in the controller.

1.4 Assumptions

The following assumptions will be made:

- The existing central Programmable Logic Controller (PLC) Siemens S7-1200 is connected via Ethernet to a panel Personal Computer (PC) in which a Windows 7 or Windows 8 operating system is running.

- The existing system's Supervisory Control And Data Acquisition (SCADA) ZENON software is running and displayed on the same panel PC

- The panel PC contains all other additional open source software running scripts for specific functions' execution

- The panel PC is securely connected to the internet, acting as the digital gateway between the bottling plant and the outside world.

- The python script sending targeted e-mails to maintenance experts is continuously running.

- The Comma-Separated Variables (CSV) files generated by the PLC are saved by a script to a specific folder in the panel PC.

1.5 Delimitations

- Our research only focuses on Siemens PLCs controllers: S7-300, S7-400, S7-1200, S7-1500 series which allow web server interfacing through which we will make use of computing functions for the programming section.

- Our research incorporates only basic requirements of I40 to the bottling process: mainly interconnection and decentralization.

- Our research focuses on the intelligent use of programming functions in the PLC and the additional open source software as well as the interaction between them to achieve research goals

- Our research is mainly intended for SMEs or small to medium scale industries. We only make use of low cost and basic devices that SMEs can afford.

1.6 Limitations

- The panel PC operation could get very slow because of the continuously running python script, the SCADA software, cloud dashboard tool and internet connection.

- All interaction of the bottling plant with internet applications will be down if the panel PC is turned off or fails.

- The bottling plant operation will still require a human operator to physically start machine’s operations.

1.7 Research methodology

The research approach of this study consisted first of a clear understanding and definition of the current problem SMEs are facing toward adopting new or advanced automation technologies and current trend of automation. Then, theoretical researches were made for all the useful concepts to be used in the design and compiled in a form of a literature review. Existing researches, technologies, literature review and automation techniques for bottling processes were also studied and some of their gaps depicted. With a clear understanding of the problem, a hypothesis was stated to palliate to the issue. Practical experiments were also performed using available platforms to support this hypothesis and experimental results recorded.

1.8 Existing automation techniques for small to medium scale bottling processes

Several studies and researches about different ways of automating bottling line and process have been published in the past years. A method consisting of finding strategic positioning for photocell sensors on an automatic bottling production system in order to improve productivity was carried out by Tarek Al-Hawari et al. (Al-Hawari T. et al., 2010). This method was quite helpful in reducing production downtime but required a couple of hardware changes to meet new sensor positioning.

A proper and consistent feeding of bottles into the production line plays an important role in the overall system productivity. ZHANG Tianxia et al. (Tianxia Z. et al., 2012) proposed another strategy to make a bottling line more efficient by designing a special hardware to feed empty bottles automatically into a beer filling production line. This special architecture made use of several stepper motors, solenoid valves, conveyors and photoelectric sensors to blow empty bottles in the system without human intervention every time the system is ready to receive bottles. The strategy was very handy because it successfully optimized one of the most manual subsections of the bottling line though it didn’t take into account the continuity of bottles path passing the feeding stage.

“Energy efficient automized bottling plant using PLC and SCADA with speed variable conveyor assembly” was developed by Jain S.P. and Haridas S.L. (Jain S.P. & Haridas S.L., 2014). The major contribution of this approach to the automation of bottling plant process is the flexibility brought into the in feed section by the speed control of the conveyor system transporting the bottles from the manual loading point to the filling section. This new approach gives the operators of the bottling plant, the possibility to speed up or slow down the feeding of bottles depending on the market demand of the production. The main issue with this approach was that it still depended on human operator intervention for some crucial operations like speed changes.

One advanced research for the automation of the bottling line conducted by Sougata Das et al. (Das S. et al., 2014) consisted on the recognition and disposal of faulty bottles in a bottle filling industry using PLC and producing human machine interface by SCADA. It is based on automatic recognition system for faulty bottles entering the system toward the filling section of the plant. The detection of bottles is done using a scanner comparing the overall shape of a normal (ideal) bottle from a preloaded picture into the scanner to every bottle passing through the conveyor system. A faulty bottle will then activate a plunger that will push the bottle to another conveyor away from the normal system cycle. This process seems perfect but the main problem with this system is that it doesn't keep track of bottles after the fault detection part. It focuses more on the detection of faulty bottles before the filling section. However problems can occur in any of the stages afterward reducing the number of bottles and affecting the output of the process.

In 2015, Chakraborty K. et al. (Chakraborty K. et al., 2015) proposed a design to feed bottles into the system as a group rather than one by one. It also describes the filling and capping stages which have been designed to receive more than one bottle at a time. The operations are controlled by a PLC and a SCADA system is in place to display the different stages of the system and to control the process from a remote location. This method saved time to the whole process compared to previous ones.

As much as the industrial sector has been evolving for the past years, advanced research were similarly conducted for the automation of the bottling process. Jaspreetkaur pannu et al. (Jaspreetkaur pannu et al., 2016) published a method for automating the filling of multiple liquid on a bottling line based on the colour of the bottle. This method makes the bottling process a bit more flexible allowing different types of beverages to be packaged on a single line.

In that same idea, Abashar A. I. et al. (Abashar A.I. et al., 2017) developed a strategy for automating and monitoring multiple liquid filling systems as done by their predecessors (Jaspreetkaur pannu et al., 2016) but this time through a wireless network. The system uses Access Points (APs) to create a Wireless Local Area Network (WLAN) connected to the internet which allows connected users to monitor the system diagnostic through the PLC browser knowing its Internet Protocol (IP) address. This strategy introduces an interesting concept: decentralized monitoring that allows users to check the system status through the PLC not only by being physically present close to the bottling system but wherever there is internet access.

The current trend of automation: I40 or “smart factory”, another remarkable advanced concept in the industrial sector, is making its way in bottling plants. Sagar T. (Sagar T. et al, 2016) introduced the I40 concept to the automation of a bottle filling plant to improve management level and efficiency of the system. This new concept introduces a market based production method where the bottling plant only produces based on the market demand, saving therefore energy when it is unnecessary to produce. Through cyber-physical (one of the I40 design requirements), the system was also able to receive daily target from vendors and when unable to meet target, would send signal to other bottling lines. Though only basic aspects of the I40 were used in this method, it opened up the door for many more improvement in bottling lines using the current trend of automation.

The previous work done on the automation of bottling processes in several industries shows that most of them used first, second or third trend of automation: mechanization, mass production and use of Information Technology (IT), respectively, to improve this process. While moving toward the fourth trend of automation, I40 or smart factory, it becomes imperative for the industrial automation sector to open up doors for this new concept in order to stay competitive in the market. The I40 is still emerging and many researches are being done to deploy much more of its capacities to the benefit of the industrial automation sector.

As advanced as I40 seems, it is not a concept reserved to high tech manufacturing industries. Its results are slowly but surely affecting multiple industrial areas. It is therefore important to apply I40 to all level of the automation sector from small to high tech through medium scale industries. However, the I40 concept is only a theory until it is applied to industrial processes to improve their quality of services compared to what they used to be.

In this research, we do not only limit to the use of I40 on its own, as a theory, but we combined some of its basic concepts to principles like linear regression, predictive maintenance and auto-parameters configuration to enhance the efficiency of a SME bottling process.

1.9 Organization of dissertation

Chap 1. Introduction: The first chapter contains mainly research background, problem statement, objectives of the project, limitations, assumptions as well as previous work done on the automation of bottling processes.

Chap 2. Literature review: This chapter provides a detailed review on all the important principle and components to be used in the research: Industrial automation, PLC and SCADA, bottling process and the current trend of automation, Industry 4.0.

Chap 3. High-Tech Automated Bottling Process: Theoretical Modelling. This chapter explains all mathematical models used in this research to achieve the objectives. Principles like linear regression, predictive maintenance and production output tracking in the bottling plant are modelled.

Chap 4. High-Tech Automated Bottling Process: Implementation. This chapter describes all steps, platforms, methods used for the implementation of the high-tech bottling process based on theoretical models. The overall system's architecture and overall process overview are described. PLC code, programming functions, python scripts, SCADA system, and cloud dashboard, designed after implementation of I40 concepts are also detailed.

Chap 5. Experimental Results and Analysis: This chapter analysis and explains the experimental results of the overall designed high tech automated bottling process; Production forecast, predictive maintenance and hourly decentralized production tracking.

Chap 6. Conclusion and future work: The sixth chapter concludes the research, highlights the benefits of this high-tech automated bottling process gives paths for future work that could be done to improve the process.

Chapter 2 : LITTERATURE REVIEW

Good background knowledge of the main stakeholders of this project is an advantage for the smooth development of the system. The aim of this chapter is to provide a detailed review on all important components and principles to be used in the research. We start by an understanding of the controller system components; its brain, the PLC and all other components in direct communication with the brain, for example, the SCADA system etc.; the principle behind operations of bottling plants, the hardware used including the field devices interfacing with the PLC, such as sensors, actuators, motors etc. Last but not least, fundamental principles of the fairly new “Industry 4.0” paradigm are reviewed for a better understand of the concept.

2.1 Industrial Automation

Industrial automation is the use of robotic devices to complete manufacturing tasks. In this day and age of computers, it is becoming increasingly important in the manufacturing process because computerized or robotic machines are capable of handling repetitive tasks quickly and efficiently (Kumar P., 2015). In other words, industrial automation uses various control systems in order to operate diverse machineries with reduced human labour aiming to achieve:

- Higher productivity
- Superior quality of end product
- Efficient usage of energy and raw materials
- Improved safety in working conditions etc.

Modern Industrial Automation Systems (IASs) are composed of physical plants that perform the physical processes and networks of embedded computers that perform the computational processes necessary to monitor and control the physical processes (Cheminod M. et al., 2013). It doesn’t matter how big or small a plant is, the use of diverse control systems and machineries within its operation to reduce human intervention and improve its output makes it part of an IAS. Industrial automation can be divided into several sub-areas, Building Automation System (BAS), Process Automation (PA), Factory Automation (FA), and Substation Automation (SA) (Gidlund M. et al., 2017).

With the industrial automation, the process control, which is in simple terms, all the methods and ways made in place to perform operations and to get the final product, is being affected. The outputs of the process are changed by varying the inputs. These outputs may be affected by other factors in the process or several external conditions as “disturbances”. The challenge for the process control designer is to maintain the controlled process variable at the target value or change it to meet production needs whilst compensating for the disturbances that may arise from other inputs (Medida S., 2007). One way of overcoming this challenge is to give instructions to controllers and/or programmable logic controllers through software programs and recover the right balance of the system.

Another important aspect to also consider is that system tasks or instructions of IASs have real-time characteristics. They are relatively simple and fixed, and these tasks are generally scheduled by table-driven scheduling algorithms to achieve predictable behaviours (Stankovic J.A. and Rajkumar R., 2004), (Chen H. et al., 2009); all programmed through software in selected controllers. The most important requirement for an industrial automation system is availability. I.e., the plant must be able to produce or construct as intended do with minimal downtime. Downtime means no revenue and only costs for the plant owner(s). In order to maximize the availability, the reliability of various system components (e.g., networks) becomes a critical requirement (Gidlund M. et al., 2017). Figure 2-1 is an overview of an Industrial automated system.

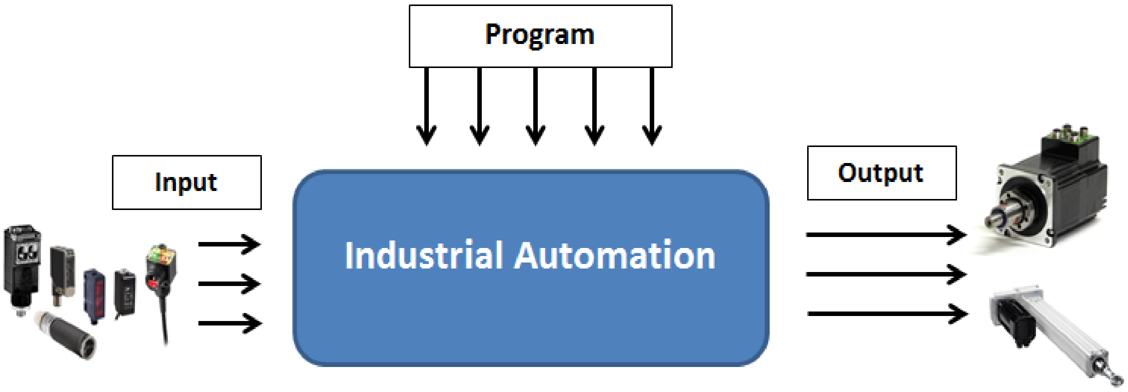


Figure 2-1: Industrial Automation System Overview

Referring on Figure 2-1, the input of the system could be any types of sensors reporting data to programmable controller they have been connected to. Based on these data, the programmable controller will manipulate the outputs of the system by either starting or stopping a motor, opening or closing an actuator etc.

As mentioned earlier on, controllers are being used in industrial automation to adjust process operation, maintain certain equilibrium within the system despite inputs changes, disturbances and achieve the desired output. One of the popular outputs of an IAS is the manufacturing of goods which usually consists on reproducing a specific product, packing, and making it available for consumers. All these steps are managed and controlled by a specific programmed controller whose features influence the whole process and operation. The choice of controllers to use for the operation of an industrial automation system is therefore as important as the process itself.

2.2 Programmable Logic Controllers

In every control environment, there is a device known as “the brain” of the system. The brain has an overview and control over every single ramification of the system. Damage on the brain affects the overall performance. Nowadays, technology is moving away from the single brain controller type concept to a more redundant and distributed architecture where a fault on the “brain device” would not stop the system from functioning.

Programmable logic controllers (PLCs) have been for a very long time the brain device in many industrial and manufacturing plants. Introduced as a replacement to the old mechanical relays, timers and counters, PLCs had a successful implementation in the industrial sector due to their compactness and robustness in this harsh environment. A PLC is known as a microprocessor developed for automation processes and control system in industrial environments. It can perform multiple functions from simple to more advance in arithmetic, counting, logic, sequencing, and timing. PLCs use internal or external memories and can be programmed to control industrial equipment and machines by integrating a number of Inputs and Outputs (I/O), which interface external electrical signals such as sensors, valves, actuators etc. (Jain S.P. and Haridas S.L., 2014).

A PLC like many other controllers is made of an I/O unit, Central Processing Unit (CPU) and memory. The I/O unit is the interface between PLC and external field devices. All control operations, logic, data transfer and manipulation work is done by the CPU. As previously mentioned, the PLC is designed to operate in the industrial environment with wide ranges of ambient temperature, vibration, and humidity and is not usually affected by the electrical noise that is inherent in most industrial locations. It also provides the cost effective solution for controlling complex systems (Ahuja H. et al., 2014).

A PLC is a user friendly, microprocessor specialized computer that carries out control functions of many types and levels of complexity. Its purpose is to monitor crucial process parameters and adjust process operations accordingly. It can be programmed, controlled and operated by a person unskilled in operating computers. Essentially, a PLC's operator draws the lines and devices of ladder diagrams with a keyboard onto a display screen. The resulting drawing is converted into computer machine language and run as a user program (Kumar P., 2015). Figure 2-2 displays the diagram of a typical PLC.

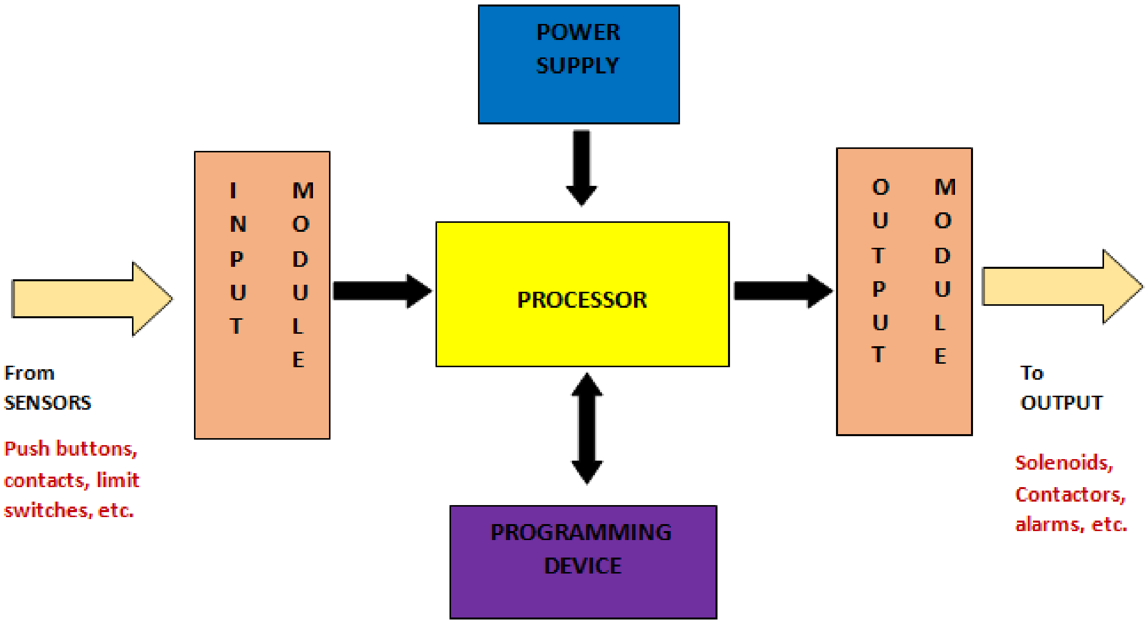


Figure 2-2: Diagram of programmable logic controller (Technical Editor, 2017)

As we would have noticed, there is a similarity between Figure 2-1 displaying the Industrial Automation Overview and Figure 2-2 displaying the Diagram of programmable logic controller. This simply shows that the programmable logic controller is the brain and heart of the industrial automation system. The typical execution schema of a PLC is cyclic: the user-defined program is executed in an infinite loop (so-called *PLC* or *scan cycle*). At the beginning of the loop the physical inputs are read, and then the program is executed using these stable input values. At the end of each loop the physical outputs are assigned which are then stable until the end of the next cycle. The duration of the scan cycle may vary in most of the PLCs (Darvas D. et al., 2016). PLC programs are writable in different types of languages depending on different PLC brands.

The International Electro-technical Commission (IEC) states five languages for PLC programming:

1) Ladder Logic (LAD) – graphical language.

2) Function Block Diagram (FBD) – graphical language.

3) Sequential Function Chart (SFC) – graphical language.

4) Instruction List (IL) – textual language.

5) Structured Text (ST) – text language (Jones C.T., 1996).

It is very important to note that as much as the advent of PLCs brought a radical change in the use of archaic mechanical relays, timers and counters we slowly but surely see in the industrial sector an introduction to the use of the so called “distributed control systems” (DCS) that are known are as advanced computer control systems using multiple autonomous controllers distributed in many section of the system with various control loops but reporting to a central operator control.

2.2.1 S7-1200 Siemens PLC

The S7-1200 Siemens PLC is the available PLC at the plant where our research was conducted. We had the task to find most important functions and twists of this PLC that could be useful in achieving our project goals. These functions are presented in the next chapters. A background research had to be performed on this Siemens model.

Over the years, Siemens manufactured different PLC models: Siemens PLC logo, S7-200, S7-300, S7-400, S7-1200, S7-1500 etc.; each model bringing in different variations and features. The S7-1200 PLC is one of the last developed models before the S7-1500 series. The SIMATIC S7-1200 controller is known as a modular and compact controller. It is the controller for open-loop and closed-loop control tasks in mechanical equipment manufacture and plant construction. It combines maximum automation and minimum cost. Due to the compact modular design with a high performance at the same time, the SIMATIC S7-1200 is suitable for a wide variety of automation applications: placement systems, conveyor systems, elevators and escalators, material transportation equipment, metalworking machinery, packaging machines, printing machines, textile machines, mixing machines, fresh water treatment plants, etc. Its range of use extends from the replacement of relays and contactors up to complex automation tasks in networks and within distributed structures. Some of the main advantages of the S7-1200 PLC are the following:

- **Scalability and flexibility in the design:** The SIMATIC S7-1200 series was designed with maximum flexibility to fit individual machine requirements allowing programmers to custom their design to meet their needs. The flexibility of the S7-

1200 is not only remarkable in its programming interface but also in the quick and ease of system expansion for possible future work.

- **Industrial Communication:** Because communication is key in every industrial system, the S7-1200 is designed with integrated PROFINET interfaces to facilitate communication with engineering system for programming, with Human Machine Interface (HMI) panels for visualization, with additional controllers for PLC-to-PLC communication and with third-party devices for advanced integration options. A Siemens S7-1200 PLC is shown on Figure 2-3.

Figure 2-3: SIMANTIC S7-1200 series PLC (Combest S, 2010)

The S7-1200 Siemens PLC can be configured and programmed through the TIA portal software. TIA portal is engineering software released by Siemens Corporation in 2011. It’s an environment that configures and programs an entire project in a single framework. The TIA portal can program PLCs, configure distributed I/O units, visualize screens, parameterize drivers and do other tasks. It consists of a number of software installed in a single framework such as step 7 professional, WinCC and others (Berger H., 2013).

2.3 Supervisory Control and Data Acquisition

Telemetry is automatic transmission and measurement of data from remote sources by wire or radio or other means. It is also used to send commands, programs and receives monitoring information from these terminal locations. SCADA is the combination of telemetry and data acquisition. SCADA system is compose of collecting of the information, transferring it to the central site, carrying out any necessary analysis and control and then displaying that information on the operator screens. The required control actions are then passed back to the process (Dhiman J. and Kumar E.R.D, 2014). SCADA stands for Supervisory Control and Data Acquisition. SCADA system is used to gathering, analysing and monitoring the real-time data of process in any type of industries. This system has the ability to control the data of remote locations and also provides the proper monitoring results (Kaur A. and Bansal D., 2016).

It is almost impossible to talk about SCADA without associating the controller from which it reads data. The controller, PLC or Remote Terminal Unit (RTU), is physically connected to field devices (sensors, motors, solenoid etc.) and the SCADA receives from the controller through a specific communication protocol: Ethernet Transmission Control Protocol/Internet Protocol (TCP/IP), Serial (RS232) etc. necessary information on the devices and can display them to operators. On one hand field devices (transducers etc.) reading pressure, temperature and flow (from field) and physically connected to the PLC and on the other hand a PLC connected to the SCADA.

The main components of a SCADA system will then be:

- RTUs or PLCs that are connected to field devices.
- Central SCADA master receiving data from RTUs or PLCs
- HMI which is the interaction between operators and the field devices.

For many years the centralised visualisation of process plant, including the storage of critical infrastructure data variables, has centred on SCADA systems, where controllers can monitor and manipulate the interaction of a system and its process assets (Grilo A. M., 2014).

In the present day, SCADA systems tend to be more mobile and dynamic comparing to the former version of SCADA. Users and engineers are no longer tied to a project site or a computer. The Application Programming Interface (API) gives customers and operators the access to monitor, control and even program various SCADA operations from any remote location (Kim T.H, 2010).

A very good example of the SCADA mobility is its access through the internet or the cloud.

2.3.1 ZENON SCADA System

The SME bottling plant where our research was conducted uses the ZENON SCADA software to monitor some of its operations. To reduce cost of the study, the same ZENON SCADA was used in experiments.

ZENON is a SCADA software developed by a company called COPA-DATA. As any other available SCADA systems, ZENON offers the graphical user interface through which operators can interact, monitor and control industrial processes in a more understandable way. Programming industrial processes requires a certain level if expertise. SCADA system presents the so called “complicated” industrial processes in a more user-friendly and graphical way usable with very little expertise.

ZENON SCADA was created to operate in various industrial sectors such as:

- **Energy and Infrastructure**: where it offers intelligent and secure interlocking systems for this domain as well as simple configuration wizards.
- **Pharmaceutical Industry**: where it provides specific graphical objects for control and monitoring fitting this sector.
- **Food and Beverage**: where it offers a large number of predefined and easy customizable figures displaying Key Performance Indicators (KPIs) in real time, Overall Equipment Effectiveness (OEE) and Multi-Touch screens for HMIs.

- **Automotive**: where it provides condition monitoring for different stages of the process, connection systems like ERP and quality management.
- **Process control system**: where it offers optimal visualization, control and supervision for different processes included complex networks (Copa-Data, 2017)**.**

The COPA-DATA website presents examples of the ZENON SCADA for the different listed industrial sectors.

ZENON is also designed to communicate with various hardware such as PLCs and many other controllers. This communication is enabled by preloaded drivers from different manufacturers, e.g. Siemens, ABB, Mitsubishi, Allen Bradley, etc. In this research, the S7-1200 Siemens PLC driver will be used to enable communication with ZENON SCADA.

2.4 Beverage industry

The role and importance of the beverage industry is becoming very remarkable in people's life. Products of this industry are being used every day by different social classes for different purposes: from normal daily routine and necessity to special event's needs.

The beverage industry is composed of two major categories and eight sub-groups. The first non-alcoholic category takes care of the production of soft drink syrup, water bottling and canning; fruit juices bottling, canning and boxing; the coffee industry and the tea industry. The second category, made of Alcoholic beverages, include distilled spirits, wine and brewing. Everyone would agree that all these listed products are in high-demand in today's economy and producers are challenged to meet all consumers' requirements to remain competitive in the market.

The beverage products industry is vast. This is obviously due to the number of manufacturers, methods of packaging, production processes and final products offered to the market (Franson D., 2016)**.** The beverage industry needs raw materials as ingredients to manufacture its products. Depending on the product to manufacture, these raw materials could be oranges, lemons, coffee beans, barley, hops or grapes etc.

The processing of beverage products involves more and more automated and mechanized operations in order to produce the expected amount of goods for the market in a limited time. This is one of the reasons why enterprises are trying to adopt the best automation mechanisms that would allow them to gain competitive advantages in the market. As technology and automation evolve, the workforce diminishes in number and technical training becomes more important (Franson D., 2016)**.** Figure 2-4 displays beverage industry categories groups and sub-groups.

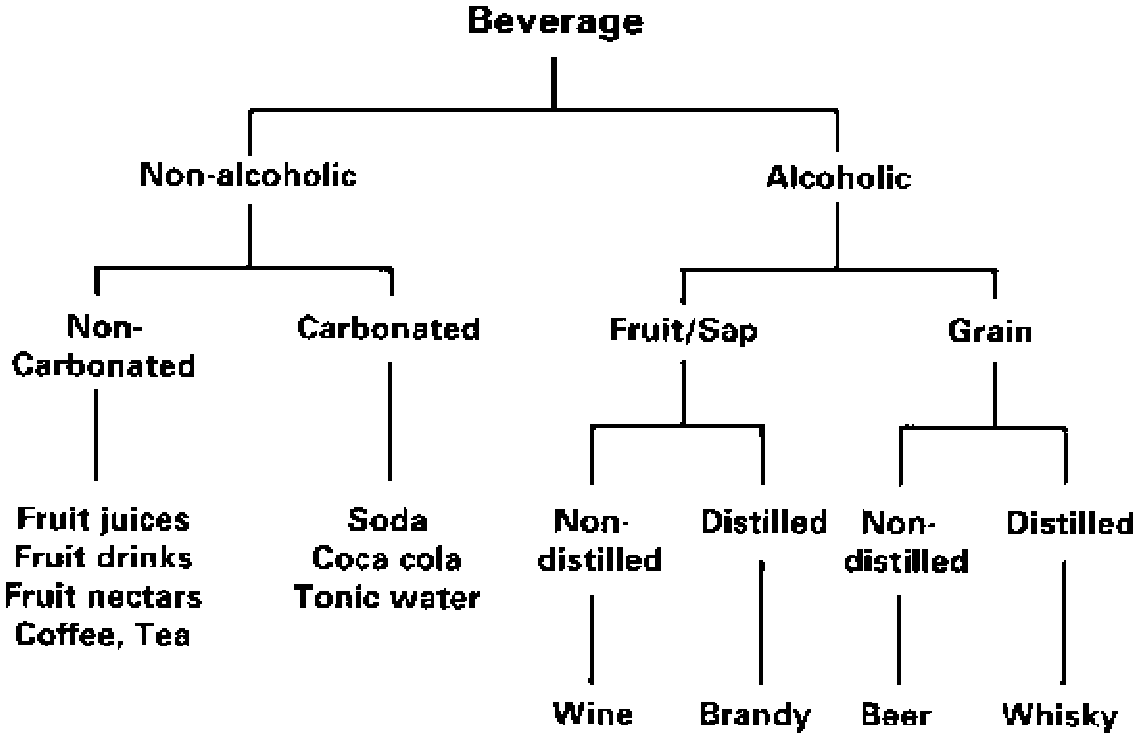


Figure 2-4: Beverage Industry categories and sub-groups (Franson D., 2016).

2.5 Bottling Process

The bottling process is one of the most frequently used processes in the beverage industry. It was reported that within breweries about 30% of the total production costs comes from the bottling processes alone. Researchers came to a conclusion that a 10% increase of the bottling line efficiency would lead to a 3% decrease of total production costs (Haider, 2008). Investing on the creation of a reliable and efficient automated production line should therefore be the way to go for bottling enterprises.

The bottling process usually consists of filling and capping bottles intended to store beverages. It can be divided into four main stages: the feeding stage, the filling stage, the capping stage and the packaging stage. Depending on the size of the industry, there could be more of fewer stages than the ones listed.

All bottles entering the bottling process are driven on a conveyor system controlled by a single or multiple motors. Many sensors are placed along the conveyor architecture to detect position of bottles at different stages of the process. The signal from sensors (as input to the system) is sent to the controller to start or stop actions on the process. There are three main electronic components used in the operation of a bottling process:

- **Alternating Current (AC) motor or Direct Current (DC) motor:** to control the conveyor belt. The two types of motors, AC and DC, are powered, constructed and controlled differently (Niku S., 2011).

  As the its mentions, AC motors use alternative current to get powered while DC motors depend on direct current sources such as batteries, DC power supplies or an AC-to-DC power converter. DC motors have shorter life span because of they are constructed with brushes and a commutator, while AC induction motors do not use brushes; they are very rugged and have long life expectancies. To control the speed of a DC motor a variation on the armature winding should be applied. An AC motor usually depends variable frequency drive control to get its speed adjusted (Arrow R.S, 2001).

  In industrial plants, AC motors have preference over DC motors because of their robustness and the long life span. Figure 2-5 shows main parts of an AC motor.

Figure 2-5: Parts of an AC motor (Electrical Engineering 123, 2016)

Important concepts for AC motors are:

Synchronous Speed

$$n_s = \frac{120 * f}{P}$$

Frequency

$$F = \frac{P * n_s}{120}$$

Number of Poles

$$P = \frac{120 * f}{n_s}$$

Horsepower

$$HP = \frac{T * n}{5250}$$

Motor Slip

$$\%Slip = \frac{n_s - n}{n_s}(100)$$

(Anaheim Automation, 2017)

Figure 2-6 shows main parts of a DC motor.

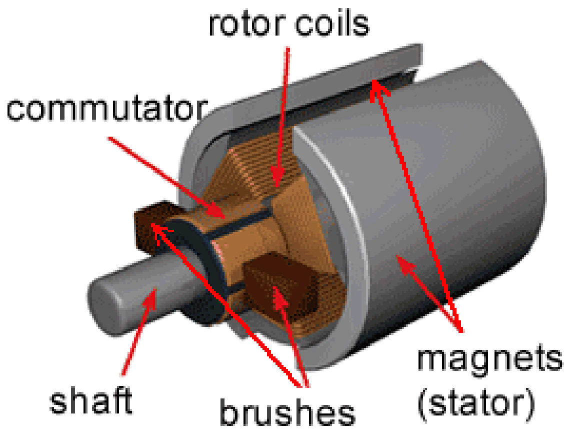


Figure 2-6: Parts of a DC motor (Electric4U, 2011)

Important concepts for DC motors are:

Torque

$$T = K_T I_A \phi$$

K_T = a constant based on the motor construction
I_A = armature current
ϕ = magnetic flux

Electromotive force

$$EMF = K_E S \phi$$

K_E = a constant based on the motor construction
S = motor speed (rpm)

(Haq M. Z., 2014**)**

- **The input sensors:** There are two main types of inputs sensors: digital sensors and analog sensors. Digital, also called ON/OFF sensors, are usually used to detect the position of bottles within the process. Whether the sensor is a Normally Open (N.O) or Normally Close (N.C), its feedback to the controller will be logic ‘1’ or ‘0’ respectively when detecting a bottle.

For all the “n” bottles present in the input side, the sensor gives the corresponding output to the controller (PLC) which in turn switches ON the corresponding solenoid valve for filling operation to start. If a particular bottle is not present, the corresponding valves remain OFF (Kothari K. et al., 2015). Figures 2-7, 2-8, 2-9 and 2-10 are examples of input sensors.

Figure 2-7: Retro-Reflective sensors - Digital (Daughtry R., 2014)

Figure 2-8: Inductive sensor – Digital (AB Elektronik, 2018)

Figure 2-10: Capacitive sensor – Digital (Fargo Controls, Inc., 2016)

Figure 2-9: IFM vibration sensors (Ifm Electronics, 2013)

- **The solenoid valve** to activate and/or deactivate the pumping of liquid into the bottles. The solenoid valve is connected as an output to the controller. As explained in the input sensor section, based on the bottle position in the process, it will receive commands from the controller to start or stop filling bottles. Figure 2-11 displays main parts of a standard solenoid valve.

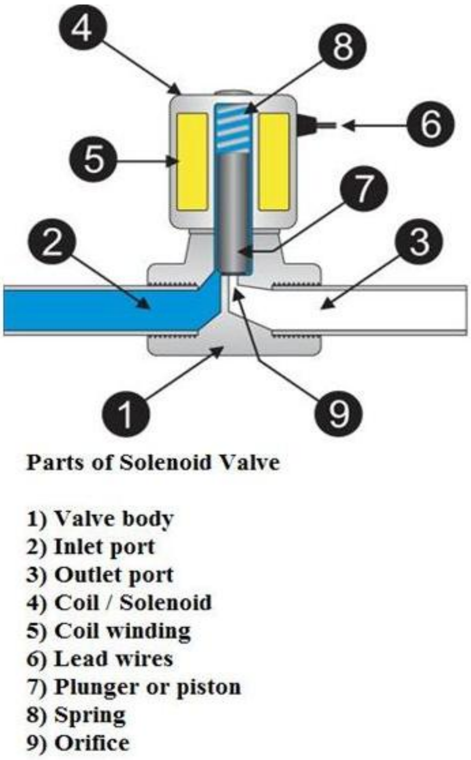


Figure 2-11: Parts of a solenoid valve (Khemani H., 2009)

Figure 2-12 is a simple representation of a bottling process with a conveyor belt represented as a green dotted line.

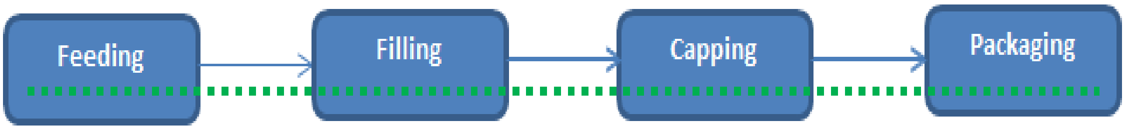


Figure 2-12: Simple bottling process stages

- **The Feeding stage:** receives and aligns on the conveyor belt all the cleaned bottles ready to be filled. The feeding stage design varies from one bottling plant to another. In very big scale industries, this stage is made automatic and controlled by some specialized motors, conveyors that will blow bottles in the system and arrange them on the feeding conveyor without human intervention. Figure 2-13 shows a more graphical representation of a feeding stage.

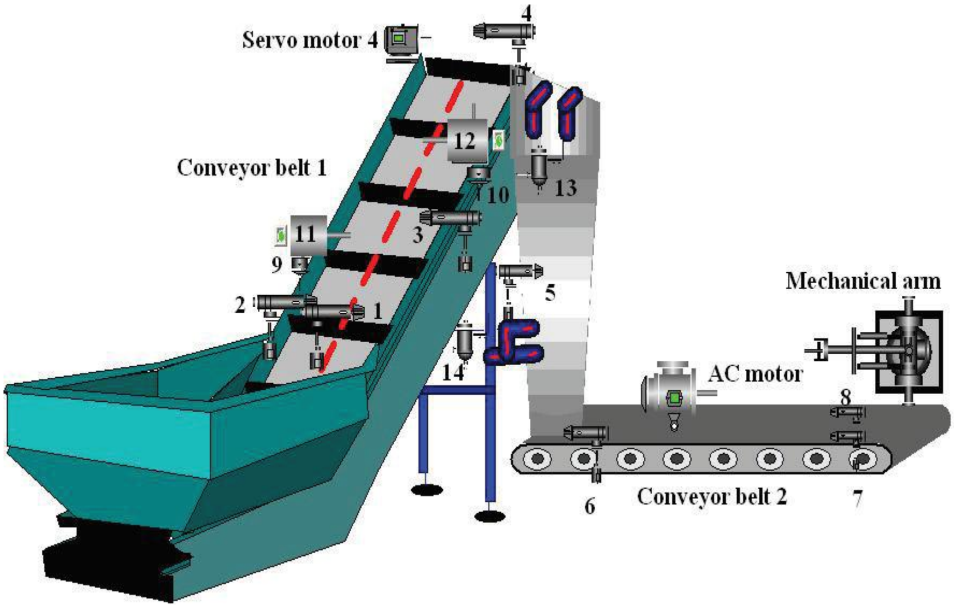


Figure 2-13: Feeding stage of a bottling process (Tianxia Z. et al., 2012)

While big industries are automating the feeding stage, small scale industries are still using human labour to align bottles into the feeding stage.

- **The Filling stage:** dispenses the intended liquid into the bottles. The filling stage of the bottling process can be customized to meet operation of one plant to another. One typical example of filling stage is described: Once the bottles are detected in the input side the conveyor motor switches ON and it starts moving in the forward direction. The bottles then reach the desired position for filling and the conveyor stops. The corresponding pumps in process tank switch ON and filling operation takes place (Kothari K. et al., 2015). Figure 2-14 represents a filling stage.

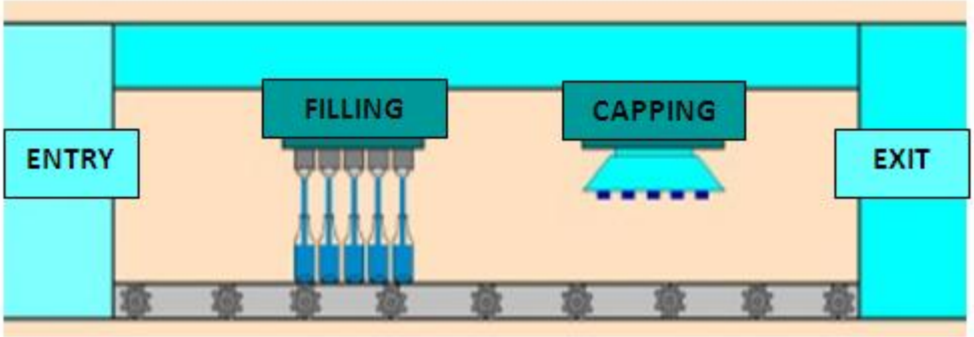


Figure 2-14: Filling stage (Chakraborty K. et al., 2015)

- **The Capping stage:** places and tightens caps on each filled bottles. The capping stage usually takes place after the filling stage. The same way bottles ran and stopped to get filled in the filling section, they will enter the capping section and wait for the caps to be tightened on them before moving on to the next stage. The capping of bottles can be controlled by some kind of motors that will apply rotary pressure on caps on top of each bottle. Figure 2-15 represents a capping stage.

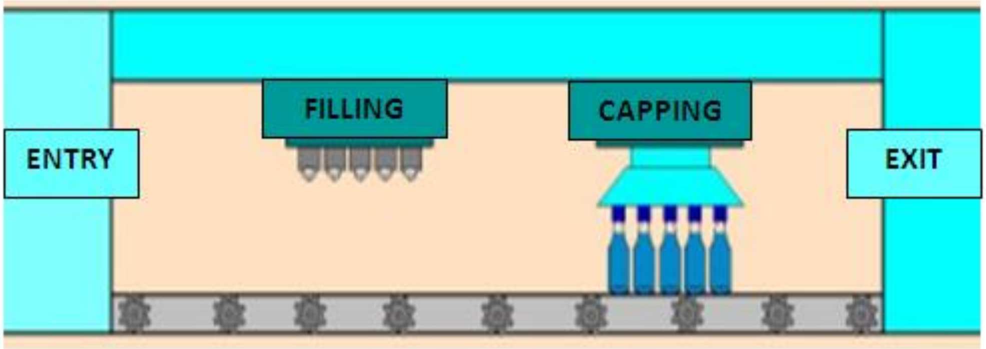


Figure 2-15: Capping stage (Chakraborty K. et al., 2015)

- **Packaging stage:** Wraps the capped bottles into groups of **ß**, **2ß etc.** bottles ready to be consumed; **ß** being the number of bottles successfully filled and capped.

2.5.1 Water Bottling Plant

In the past years, only a certain category of people, foreign tourist, highly health conscious people, high class, was privileged enough to drink bottled water. It was later established that in the present decade another flow of consumers among average consumers, education, offices etc. have increased the sales of bottled water (MSME, 2011).

As per the International Bottled Water Association, bottled water is becoming the fastest growing beverage industry in the world. Sales of bottled water have increased by 500 percent over the last decade.

The bottling procedure is often associated with the production of the bottles. In most cases, a single plant produces bottles and then fills them with the liquid product (Mainardi E. et al., 2007). The bottling process stages in a water bottling plant are similar to the ones listed in the previous section except a saturating stage included if there is a need to produce carbonated water (Comac kegging & bottling plant, 2015**).** A water bottling line is displayed on Figure 2-16.

Figure 2-16: Water bottling plant line (Fran R., 2017)

2.5.2 Soft drink bottling plant

The main ingredient in soft drink manufacturing is water, which is treated and cleansed to meet exacting quality-control standards, most of the time exceeding the quality of local water supply. This is a very critical process in order to achieve high product quality and consistent taste profiles. While ingredients are prepared for beverages, the treated water is piped into large, stainless-steel tanks. At this stage various ingredients are added and mixed.

It is common for bottling companies to purchase concentrate from other firms. Soft drinks get cooled in large, ammonia-based refrigeration systems for carbonation (absorption of carbon dioxide (CO_2)) to occur. Fruit-flavoured soft drinks usually have less carbonation than colas or sparkling water. Once carbonation is done, the product is ready to be filled into bottles and cans.

Unlike water bottling plant, the filling room usually is separated from the rest of the plant, to protect open product from being contaminated. The filling operation is mainly automated with very minimal personnel. Filling room operators are intended to monitor the equipment for efficiency adding bulk lids or caps to the capping operation as necessary (Franson D., 2016)**.**

The two bottling plants we just had a look at, present similarities in their processes. Though internal variants might differ, they both use bottling process structure to generate their final products. Figure 2-17 shows a soft drink bottling line.

Figure 2-17: Soft drink bottling plant line (Qureshi W., 2017)

2.6 Industry 4.0

The term “Industry 4.0” is used for the next industrial revolution - which is about to take place right now. This industrial revolution has been preceded by three other industrial revolutions in the history of mankind.

The first industrial revolution was the introduction of mechanical production facilities starting in the second half of the 18th century and being intensified throughout the entire 19th century. The first industrial revolution brought the mechanization of production. From the 1870s on, electrification and the division of labour (i.e. Taylorism) led to the second industrial revolution. The second industrial revolution was about mass production. The third industrial revolution, also called “the digital revolution”, set in around the 1970s, when advanced electronics and information technology developed further the automation of production processes. The third industrial revolution means the digitization (electronic component computer and IT. I40 enables suppliers and manufacturers to leverage new technological concepts like Cyber Physical Systems (CPS), Internet of Things (IoT) and Cloud Computing (CC): New or enhanced products and services can be created, cost can be reduced and productivity can be increased (Petrasch R. and Hentschke R., 2016).

Industry 4.0 comprises the following terms or technological concepts:

- Embedded Systems (ES) / CPS: Networks of IoT devices that interact physically with its environment, e.g. industrial robots with sensors and actors need physical input and provide physical output. ES and CPS are can also be equipped with digital interfaces. An ATM is an example for an embedded system.

- Internet of Things / Cloud of Things (IoT / CoT): Physical objects or components like ES or CPS that contain software and are connected to a network (Internet connectivity) and typically to a Cloud application creating opportunities for new services through integration of the physical and the digital world, e.g. automatic monitoring. The “things” can be any items, e.g. a conveyor belt in a factory. In the context of Industry 4.0 the term *Industrial Internet of Things* (IIoT) is used.

- Service-Oriented Architecture (SOA) / Internet of Services (IoS) / Cloud Computing: Service-oriented and cloud-based infrastructures and applications have advantages concerning scalability, elasticity, reliability, performance, device and location independence and more. Service models are for example Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Cloud applications can also be categorized in private, public or hybrid Cloud. Products for CC are available as open or closed source software systems (Petrasch R. and Hentschke R., 2016).

The German Federal Ministry of Education and Research defines Industry 4.0 as “the flexibility that exists in value-creating networks is increased by the application of CPPS. This enables machines and plants to adapt their behaviour to changing orders and operating conditions through self-optimization and reconfiguration (Shrouf F. et al, 2014).

T. Niesen et al. (Niesen T. et al., 2016) mention that I40 is characterized by a progressing integration of ICT (Information and Communication Technologies) into manufacturing systems. Based on that, so-called CPS emerge at the intersection of IT components for information processing as well as data exchange and mechanical or electrical machine components.

Smart factory techniques and smart manufacturing areas originated from the research in "artificial intelligence" when the sector of manufacturing has introduced artificial intelligence in 1980's. Since then, smart machinery, processes and systems have emerged, and the techniques of smart manufacture become diversified. In addition, thanks to the advance of technologies like sensing systems, system platform, software technology and cyber communication in recent years, smart manufacturing systems are becoming more reliable, and there are even remote controls that transcend regional restrictions, which are all technical embodiments of smart factory (Chang H.F et al., 2016). A summary of the I40 concept is displayed on Figure 2-18.

Figure 2-18: Industry 4.0 concept (Lean Manufacturing Times, 2018)

2.6.1 Industry 4.0 Design principles

As per S.T. Payghan et al. (Payghan S.T. et al., 2016) there are six design principles in Industry 4.0. These principles support companies in identifying and implementing:

- **Interoperability**: the ability of cyber-physical systems (i.e. workpiece carriers, assembly stations and products), humans and Smart Factories to connect and communicate with each other via the Internet of Things and the Internet of Services.

- **Virtualization**: a virtual copy of the Smart Factory which is created by linking sensor data (from monitoring physical processes) with virtual plant models and simulation models.

- **Decentralization**: the ability of CPS within Smart Factories to make decisions on their own.

- **Real-Time Capability**: the capability to collect and analyse data and provide the derived insights immediately.

- **Service Orientation**: offering of services (of CPS, humans or Smart Factories) via the Internet of Services.

- **Modularity**: flexible adaptation of Smart Factories to changing requirements by replacing or expanding individual modules

The basic principle of I40 is that by connecting machines, work pieces and systems, businesses are creating intelligent networks along the entire value chain that can control each other autonomously. However, the I40 concept is only a theory until it is applied to industrial processes to improve their quality of services compared to what they used to be.

The term Industry 4.0 is recent and the concept is starting to take shape with the increase of research and academic studies on the theme. The idea behind Industry 4.0 converges on the execution of industries referred to as intelligent. This concept is related to a modular structure, which uses physical and virtual systems that monitor industrial processes, aimed at continually improving the variety of different performance indexes within the process. This concept is made possible by IoT, which adds a flexible network of sensors on a large scale, these communicate over the Internet as well as through communication between machines Machine-to-Machine (M2M). In this manner, within the IoT, physical and virtual systems communicate and cooperate amongst themselves and with humans in real time (Zanella A. et al., 2014).

According to ZVEI (ZVEI, 2015), the model for the intelligent industry is composed of four aspects. The first is Vertical Integration through network elements. For example, equipment and machines communicate with the products and these in turn with the processes. The second is Horizontal Integration through networks; this aspect provides the interconnection between distinct areas by use of the Internet. However, the third aspect concerns the management of the production and engineering life cycle from end-to-end, where the whole production and logistics process is monitored and managed in real time. Finally, the fourth aspect, the workings of the company, this has a fundamental role in the management of the entire mechanism through new technologies and tools, which help in the search for quality improvements (Da Cunha M.J et al., 2017).

Industry 4.0 brings in a new generation of manufacturing systems that will generate additional streams of in-process data with the potential of transforming the way products are designed, manufactured and serviced. The availability of large data sets provides new opportunities to enhance quality improvement activities through better monitoring and control of processes and real-time adjustments based on the analysis of continuous data streams. ISO 9001:2015 standard defines continual improvement as a “recurring activity to enhance performance”, and the one that generally leads to a corrective or preventive action (Giannetti C., 2017).

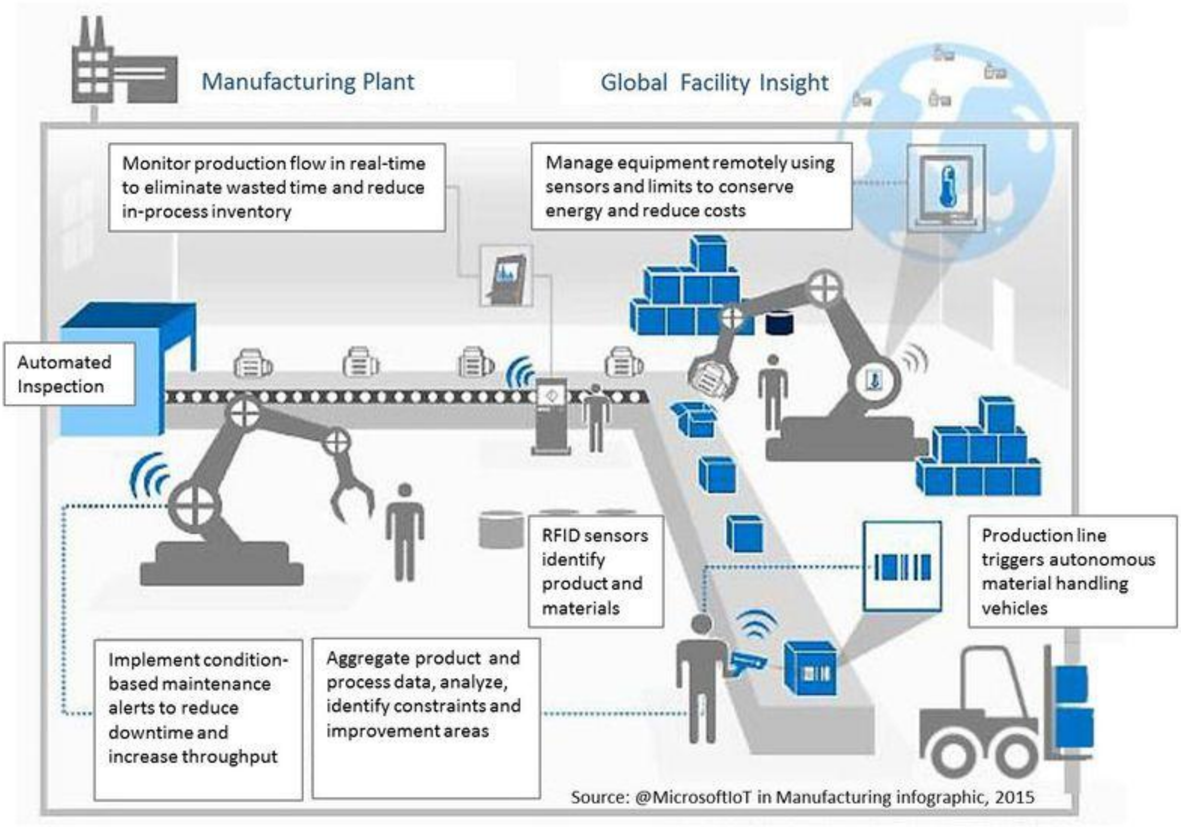
In the factory of the future, dynamically interconnected systems produce a large amount of real time data, also relying on widespread IoT technologies. By collecting and exploring data, modern enterprises might be able to implement I40 capabilities, such as self-awareness, self-configuration and self-repairing, and consequent decision making, through the design and provision of new generation services that, if properly integrated with traditional products, can determine the success of the organisation (Monostori L., 2014). According to this vision, data becomes a valuable industrial asset, useful to create new business opportunities shifting from product offering to an integrated “product plus service” offering, according to the emerging data-driven innovation paradigm (Hou Z. and Wang Z., 2013). Due to the recent addition of the theme IoT in technological terms, the concepts have not yet reached a degree of maturity or are still divergent and not well defined. Proof of this can be found in the appearance of a variety of technologies and standards with diverse proposals that depend on their scenario of application. An example of such according to D. Z. D. Kolberg (Kolberg D. Z. D., 2015), where the industrial architecture 4.0 can be seen through the integration of four main elements within industry, which are:

- Intelligent Products collect data from the process for analysis during and after its production. The product knows its production processes and integrates with intelligent machines.

- Intelligent Machines possess computational and sensing power capable of aiding the system and the operators in the detection of possible failures during the process, in real time.

- Intelligent Plans optimize the processes by negotiating execution cycles and finding the optimal operation point of the system.

- Intelligent Operators supervise and control the whole process in real time, aided by the entire virtual system, which can flag possible deviations and plan online maintenance for different IoT equipment, such as glasses, watches and smartphones (Da Cunha M.J. et al., 2017). Figure 2-19 displays main components of a smart manufacturing system in I40.



Source: @MicrosoftIoT in Manufacturing infographic, 2015

Figure 2-19: Smart manufacturing in Industry 4.0 (Jackson B., 2016)

2.6.2 Smart manufacturing

Tim Niesen et al. (Niesen T. et al., 2016) mention that I40 is characterized by a progressing integration of ICT into manufacturing systems. Park C.Y et al. (Park C.Y et al., 2017) defines manufacturing as the process of transforming (raw) materials and energy, by means of workers and machinery, into products that address manufacturing requirements from stakeholders.

On the other hand, smart or intelligent manufacturing refers to any manufacturing processes which involve a degree of computational intelligence. This can be via the use of embedded sensors as in the case of IoT technologies (Atzori, L et al., 2010), and cover the use of analytical techniques on historical process data to provide knowledge discovery and support decision making within manufacturing systems (Lee, J et al., 2016), (Mi, M. & Zolotov, I., 2016) or, ultimately, the development and implementation of full Cyber-Physical-Systems (Monostori L., 2014), a synthesis of physical and digital technologies across the entire manufacturing system; and necessary associated technologies and frameworks (Ollif H. & Liu Y., 2017).

Lu, Y., et al. (Lu, Y., et al., 2016) further summarizes smart manufacturing as any manufacturing system having one or a combination of the following characteristics:(1) Digitization of manufacturing enterprises, (2) Connected devices and distributed intelligence, (3) Collaborative supply chain, (4) Integrated and optimal decision making, and (5) Sensors and big data analytics. While the industrial automation sector currently offers this advanced manufacturing scheme for improvement of industrial processes, many SMEs are still under the first, second or third industrial revolution regarding this new concept as reserved to big industrial players.

2.6.3 Advantages of Industry 4.0 in industrial processes

Decentralization of the intelligence helps to create independent management of the processes and smart objects interconnected throughout networks, and the interaction between the virtual and real world is a crucial new aspect on manufacturing and production processes. Cyber-physical systems represent the next revolutionary step based on existing embedded systems.

Along with the Internet and available online services and data, embedded systems join to form cyber-physical systems.

CPS provides the basis for the Internet of Things creation, which in combination with the Internet of Services will enable Industry 4.0. These systems are relied on innovative technologies that transform multiple applications, and the boundaries between the virtual and real world will disappear. As a result, these systems promise to revolutionize our interaction with the physical world just as the Internet has transformed communication and personal interaction (Ungurean I. et al., 2014).

The main focus is on the ability of the systems to perceive information, to derive findings from it and to change their behaviour accordingly, and to store knowledge gained from experience. Intelligent production systems and processes as well as suitable engineering methods and tools will be a key factor to successfully implement distributed and interconnected production facilities in future Smart Factories”. Since exchange data and information between different devices and parties in real time is the key element of smart factories; such data could represent production status, energy consumption behaviour, material movements, customer orders and feedback, suppliers’ data, etc. The next generation of smart factories therefore will have to be able to adapt, almost in real time, to the continuously changing market demands, technology options and regulations. The smart factories provide the customers with smart products and services which will be connected to the internet. Then, the smart factories will collect and analyse data coming from the smart products and related smart applications. This analysis enables the factories to better define customers’ behaviours and needs and so provide them with new and more sustainable products and services. In addition to that, IoT technology enables the customers to be involved in production design process (Shrouf F. et al., 2014).

In the vision of the IoT network connectivity is extended to electronic devices from very simple to the point where anything can connect to the Internet. Field buses, embedded systems and distributed systems come to provide enhanced support to implement such a vision. In a report from 2005, the International Telecommunication Union (ITU) suggested the following: "*Internet of Things will connect objects from the world, both in a sensory and intelligent way*". By unifying various techno-logical development methods, ITU identified four dimensions in IoT: identifying elements ("labelling things"), sensor networks and wireless sensor networks ("things that feel"), embedded systems ("things that think") and nanotechnology ("contraction things"). The terms of "things" in the IoT vision is very broad and includes a variety of physical elements. The terms of things include portable personal items such as smart phones, tablets and digital cameras. Furthermore, IoT includes elements of our environments (be it home, car or office), and things equipped with Radio Frequency Identification (RFID) tags (or other) connected to a gateway device (e.g. smartphones). From those mentioned so far, a huge number of devices and things will be connected to the Internet, each providing data and information and some even services (Ungurean I. et al., 2014).

Industry 4.0 focuses on developing concepts and methods to make production processes more flexible and transforming currently fixed production lines into automated, autonomously organized assembly lines. The collection and analysis of extensive amounts of real-time data from various production resources provides the basis for achieving this goal. Processing this data by means of adapted methods from the fields of *business intelligence* and *business analytics* is key to establishing an integrated management information system, which supports appropriate strategic and operational decision-making. In particular, an integrated analytics database opens up new perspectives with respect to company-wide risk management (Niesen T. et al., 2016).

Industry 4.0 will set up a new manufacturing mode, in which its products and services are high flexible, personalized and economical. In this mode of social manufacturing, traditional industry boundaries will disappear; some new activity and cooperation forms will come into being. The progress of creating new value will be changed and industrial chain will be recombined. Meanwhile, in this mode, the most core of social manufacturing is social computing and we must transform traditional factories to these industries that can actively perceive and response to user' large-scale personalized needs.

At that time, public will energetically participate in the industrial chain and send their demands in the form of crowd sourcing. And, the factories will be just to find the demands and fulfil their needs. What's more, in industry 4.0, it will be possible to simulate all the steps in the manufacturing process and depict their influence on production. This will include simulation of inventory levels, transport and logistics, the ability to track the usage history of components that have already been used in production and provision of information relating to how long components can be kept before they expire. This will enable product-specific set-up costs to be calculated and reconfiguration of production resources to be kept to a minimum. It will also be possible to assess the relevant risks and simulate the different costs and margins of alternative suppliers, including simulation of the environmental impact associated with using one supplier over another. Extensive networking of manufacturing systems will make it possible to analyse alternative suppliers and their capacity in real time. It will be able to contact and engage suppliers directly via the appropriate secure channels in the supplier cloud (Wan J. et al., 2015).

Industrial controls and, in particular, PLC controllers currently form an important technological basis for the automation of industrial processes. Even in the age of I40 and Industrial Internet, it can be assumed that these controllers will continue to be required to a considerable extent for the production of tomorrow. However, the controllers must fulfil a range of additional requirements resulting from the new production conditions. When applying Industry 4.0 principles, high-quality networked production systems result based on CPS, also referred to as CPPS. A series of I40 requirements are placed on the future controllers used in these systems.

These include:

- Autonomy, Reconfigurability and agility (Plug & Work);
- Overcoming the strict information encapsulation of controllers;
- Introduction of the service paradigm in the production automation (production services);
- Networking in local and global networks;
- Interoperability between heterogeneous control systems;
- Dependencies are to be changeable dynamically to the runtime;
- Use of models for the development of "higher-quality" control approaches;
- Orchestration of heterogeneous controllers.

Current PLC controllers cannot yet fulfil the majority of these requirements or can only do so on a rudimentary basis or with extremely high expense. One of the basic requirements of future and I40- able PLC controllers involves efficient networking in an, at least partially, global network. Here the IP network functions as a global network in the version as Intranet or Internet with all associated standardized ICT. Only in this way can the required integration become part of a future I40 production landscape (Langmann R. and Rojas-Pena L.F., 2016).

According to a survey by American Society for Quality (ASQ) in 2014, 82 % of organizations that claim to have implemented smart manufacturing say that they have experienced increased efficiency. 49 % experienced fewer product defects and 45 percent experienced increased customer satisfaction”. Also, the Economist Intelligence Unit estimated the current and future use of the IoT through running a survey in June, 2013 on the global business community: according to their results, 38% of respondents believe that the IoT will have a major impact in most markets and industries. Three years from survey time, 96% of the respondents expect their business to be using the IoT in some respect, 63% believe that “companies slow to integrate the IoT will fall behind the competition” and 45% believe “adopting the IoT will make their company more environmentally friendly" (Shrouf F. et al., 2014).

I40 is the latest industrial revolution, but instead of steam trains and textiles, this generation is using artificial intelligence, forever changing the way machines collect and interpret data. Manufacturing in this new era of machine learning requires little human intervention, transitioning from an input and output approach to a fluid conversation between humans and robots. Machines are now equipped to make decisions and provide technical assistance, which has led to more transparent communication (Roubaud J., 2017).

I40 has improved quality of service of many industrial strategies: predictive maintenance is one of them. Below are some of the values I40 brings into the predictive maintenance concept:

- **Real-time condition monitoring:** I40 software can help to create the information availability and processing required. Machine and sensor data is recorded and displayed in real time, providing the basis for real-time condition monitoring. Data visualization is not restricted to the control station. The same solution can be available everywhere – from big screens to tablet computers and smartphones, on premises and in the cloud. And it can be accessed by everyone – from those in charge of machine settings to a variety of experts (Kohler M, 2016). In this research, real-time condition monitoring is performed by the PLC and its SCADA.

- **Flexible evaluation and analysis options**: I40 software is designed in a flexible manner using highly customized rules and analyses that allow production planners, process experts, or even the maintenance technicians to configure on their own without the need for complicated IT know-how. The software adapts to the expert's needs.

  As such, I40 is ushering in a paradigm shift. The software is designed for human needs and not, as before, programmed with just the machine in mind. This means the maintenance technician can create rules such that defined machine parameters trigger notice of upcoming maintenance. And this maintenance is performed only when it is actually needed. Conversely, of course, limit values and rules can also be set such that unscheduled machine stoppages are immediately displayed and notifications sent to the relevant people (Kohler M, 2016). Technicians, productions planners and experts will sit together and decide on different rules that will define the predictive maintenance. This rule is modeled in a computer language, in this research in a PLC ladder language, and loaded into the system.

- **Targeted notification of experts**: As soon as the software has identified an upcoming maintenance task based on the pre-set parameters, the information must be forwarded quickly and specifically to the right team member via a digital ticket. This simply means that, for example, an available worker with the right qualifications for maintaining the laser machine in hall 3 receives a ticket in their account and on their phone telling themto carry out the maintenance (Kohler M, 2016).

  It is one thing to detect a threat in a system and it is another one to take action against that threat. Once predictive maintenance has been schedule for the system, it has to be addressed to the intended person as soon as possible for action to be taken before actual faults occur. In this project, we will also make use of this aspect to send e-mail notifications through the internet directly to the intended maintenance officer.

At the end of this chapter, we have successfully gained good and common understanding of main components and principles of the research. All this background is useful for us to start the theoretical modelling of the high-tech bottling process using Industry 4.0 concepts and to later transform it into practical experiments.

Chapter 3 : HIGH-TECH AUTOMATED BOTTLING PROCESS - THERORETICAL MODELLING

The design of any sustainable engineering system is founded on a theoretical mathematical model. One of the advantages of the mathematical modelling is that it allows the engineers to test the system, detect possible abnormalities, and adjust results prior to actual installation. In quantifying the design of the system through mathematical modelling, it becomes easier to assess results obtained and open a path for optimization in future.

Our High-Tech automated bottling process is designed using Industry 4.0 concepts which emphasizes on increasing intelligence in machines. Systems that are able to communicate and interconnect with the outside world; that can detect possible threats on their own and foresee actions to be taken before extensive failure occur; that operate with high accuracy using least human intervention; that avoid wastage of resources by tracking closely the exact amount of items being produced.

The mathematical modelling of the high-tech automated bottling process comprises the following:

- **Optimization of bottling production through simple linear regression:** Linear regression is used to predict a certain value based on a correlation between some existing parameters. Industry 4.0 by the internet allows the bottling process to collect useful data that are used to forecast the production of the plant by being substituted in the regression formula. By forecasting the production of the plant, it becomes possible to get on time the right amount of stakeholders needed and to reduce downtime due to material shortage.

- **Predictive maintenance and early fault detection:** Industry 4.0 uses smart sensors to collect important data that can be analyzed and interpreted for different use. In this project, predictive maintenance, which is the detection of possible faults on a system before they actually occur, is achieved using an intelligent vibration senor mounted on the conveyor motor of the plant. Once these threats are detected, information is transmitted real-time, thanks to Industry 4.0, to the intended personal for proper action to be taken.

- **Auto configuration of parameters:** The least operators intervene in the bottling process operation the least downtime the system is exposed to. Our project develops a method for the bottling process to configure its operational parameters on its own without human intervention after the first system start. This was performed by modeling the production as a time function in order to track closely its actual value throughout the process. By tracking closely the daily production of the plant, it becomes easier to monitor whether the daily target of the plant will be met or if any fault caused during the day affected it; this method then foresees changes of parameters to meet production target or at least get very close to it compare to what it should have been.

- **Decentralized monitoring through an intelligent dashboard in cloud:** A bottling plant has many information and data to monitor: status of sensors, motors, fans, bottling counts during working hours, after hours, total count, alarms etc. Some of this information is meaningful for operators, some for supervisors and some only for management. Usually management relies on reports done by supervisors who need to collect information from operators or directly from the system. This approach is not always accurate because relying on human beings. It’s time dependent because access to information can only be done during the day after supervisors collected data and compiled a report. It’s also time consuming because supervisors or operators have to physically go in front of the machine to get data.

  To palliate to this issue, our project also designs an intelligent dashboard to manage and track all important information for management in one place. This dashboard is also available in a cloud which means that it can be accessed anywhere and anytime, where there is internet connection and refreshes automatically data from the bottling plant through the internet.

3.1 Simple linear regression to forecast production of a small beverage supplier: theoretical modeling

As mentioned by Jiafu Wan (Wan J. et al., 2016), the development of intelligent factories is a new direction of I40. One of its manifest advantages is: system with independent capability meaning that the system can collect and analyse inside and outside information to programming their behaviour. The main focus is on the ability of the systems to perceive information, to derive findings from it and to change their behaviour accordingly, and to store knowledge gained from experience. Intelligent production systems and processes as well as suitable engineering methods and tools will be a key factor to successfully implement distributed and interconnected production facilities in future Smart Factories (Shrouf F., et al., 2015).

The design of our method started with data collection of important values in the production of a small beverage supplier. This was done in order to learn its behaviour, determine a possible correlation between these data and find a way to improve production based on that relationship. Data collected were: the daily amount of beverage sold to the public on a period of 21 days and the corresponding average temperature during these days. Daily temperatures were collected by a small analogue temperature sensor. As a Siemens PLC S7-1200 was used to operate the beverage plant, all collected data were directly saved into the PLC database and exported as a csv file on the PLC webserver application. The temperature sensor was also wired as an analogue input of the PLC, making it easier for temperature to be recorded. Because of the possible inaccuracy arising in analogue measurement, three sets of temperature, $\boldsymbol{\delta}_{\mathbf{(t1)}}$, $\boldsymbol{\delta}_{\mathbf{(t2)}}$ and $\boldsymbol{\delta}_{\mathbf{(t3)}}$, were daily recorded at three hours of the day, **t1=09:00, t2=13:00, t3=17:00**, and only their average was calculated in the PLC, through Equation (3.0) and saved into the database as the temperature of the day:

$$\delta_T = \frac{\delta_{(t1)} + \delta_{(t2)} + \delta_{(t3)}}{3}; \qquad (3.0)$$

Equation (3.0) represents the average daily temperature saved in the PLC database as the “Temperature in ^{o}C” for each day.

The following data on Table 3-1 were collected in the PLC after the selected 21 days:

Table 3-1: Beverage quantity sold versus Temperature of Day data

Day	Temperature in ◦C	Beverage Sold
1	19	70
2	22	72
3	20	71
4	25	74
5	30	79
6	23	73
7	26	75
8	32	79
9	34	79
10	45	86
11	37	81
12	31	79
13	29	77
14	23	73
15	21	71
16	22	71
17	22	69
18	24	67
19	17	68
20	10	68
21	15	70

Note: Please note data were not all collected on consecutive days. This is for the logical reason that when collecting samples for testing, there is much more reliability to rely on random sample. The sale of beverages could have been influenced by external factors like certain event happening during the time our testing took place increasing the sale, or the plant top personnel being off or on leave reducing the sale, or the plant machine equipment being under repair or maintenance etc. To reduce the risk of being influenced by some of these unusual factors, we decided to opt for non consecutive day's data collection. This would offer more reliability in our results. For better experimental results, data collection varied between colder and hotter days.

Also, the temperature reading came from an analogue sensor temperature wired to the PLC. The readings recorded are not necessarily the exact daily temperature. The most important aspect is to notice the relationship between temperature variation and amount of beverages sold.

The above data collected reveals an important behaviour or characteristic of the system: the number of beverages sold in a day is proportional to the daily weather temperature; the higher the temperature, the higher the sale of the day. We can therefore say that the beverage sale is a dependent variable of the daily temperature (independent variable):

Daily Beverage sale: $\mathbf{B_X}$ and **Daily Weather Temperature**: $\mathbf{T_X}$.

With these two variables related to one another: one depending on variation of the other, it is possible to generate a mathematical expression to forecast or predict the approximate value of the dependent variable knowing the independent one; this is called simple linear regression analyses. We will first plot on Figure 3-1 a graphical representation of the relationship between these two variables that will be later on, be converted into mathematical expression.

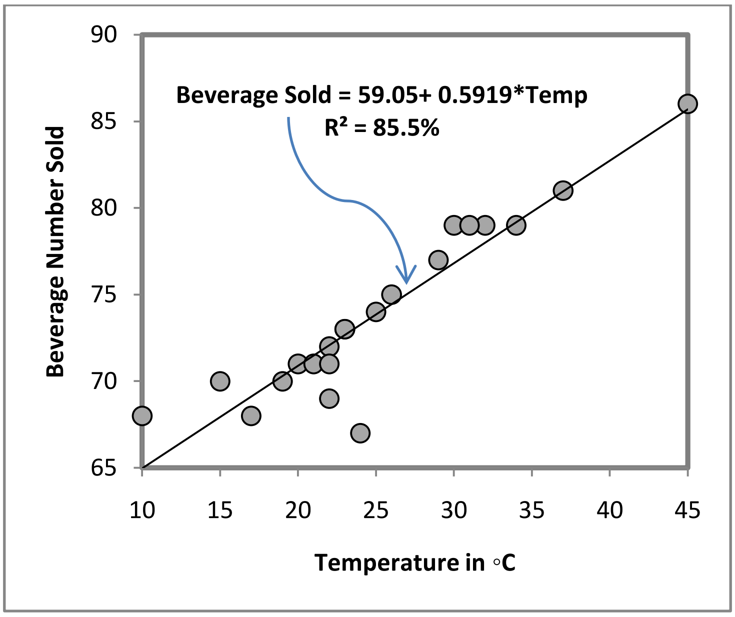


Figure 3-1: Graphical Representation of Beverages Sold versus Temperature

From Figure 3-1, the mathematical expression representing the relationship between the variables can be written by equation (3.1) as follows:

$$B_X = 59.05 + 0.5919 * T_X \tag{3.1}$$

To verify, the consistency of equation (3.1), any value of the temperature **T_X** from Table 3-1 cam be substituted into it to get its equivalent beverage number sold **B_X** for the day. The results from the equation will obviously not all be 100% exactly equal to their equivalent value on the table; the system being operated by human beings there will be a certain percentage of error caused by factors like possible incorrect readings on the temperature sensor, inconsistency of beverage number produced due to machine faults, non-performing operators but mainly shortage of equipment stock to bottle beverages: bottles, ingredients etc. On Figure 3-1, errors are represented by all the dots far away from the equation axis (3.1) and circled in red on the graph in Figure 3-2:

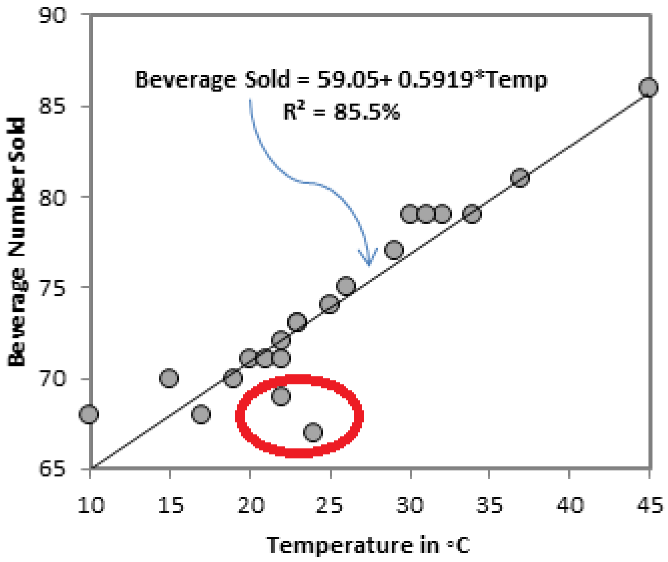


Figure 3-2: Production Error in beverage plant

In the design of our new method, equation (3.1) will be used to forecast the weekly production of the beverage plant and acquire in advance all possible equipment useful for the production. The two variables on equation (3.1) will then be transformed to vectors of respectively daily beverage sale and daily temperature.

Independent variable: Temperature of day T_X will be converted into temperature of the week T_W that is in reality a vector of day temperature from Monday to Saturday (production week days). The vector is defined in equation (3.2).

$$T_W = [T_M, T_T, T_W \ldots T_{SAT}] \tag{3.2}$$

- $T_{M:}$ Temperature on Monday
- T_T: Temperature on Tuesday
- T_W: Temperature on Wednesday etc.

Dependent variable: Beverage Sale of day B_X will also be converted into beverage sale of the week B_W that is in reality a vector of the daily beverage sale from Monday to Saturday (production week days). The vector is defined in equation (3.3).

$$B_W = [B_M, B_T, B_W \ldots B_{SAT}] \tag{3.3}$$

- B_M: Beverage sale on Monday
- B_T: Beverage sale on Tuesday
- B_W: Beverage sale on Wednesday etc.

It is very important to remember that equation (3.3), the dependent vector, will only be a result of equation (3.1). Figure 3-3 is an overview of the overall operation to obtain the weekly production forecast:

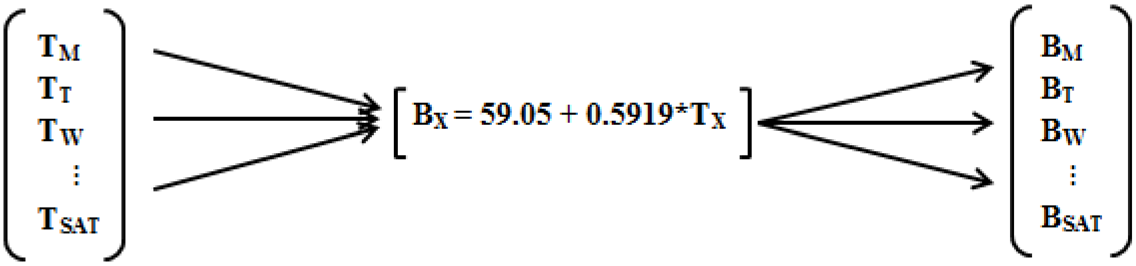


Figure 3-3: Weekly production forecast

All the different equipment to be ordered for the production of the beverage plant will also depend on the results of equation (3.1). The number of bottles to be ordered from the bottle manufacturer as well as the quantity of ingredients for the production will be estimated as follows:

Weekly bottles number to be ordered for production is expressed as $\mathbf{B_N}$ in equation (3.4):

$$B_N = \sum_{i=1}^{6} B_x(i) + \mu \tag{3.4}$$

- µ: is a safety margin factor that will be added to the calculated beverage sale of the week to reduce possible error.

Equation (3.4) can be explained in words as the sum of beverage sale from day 1 of the week (Monday) to day 6 of the week (Saturday). As mention earlier, production weeks days are Monday to Saturday.

Weekly beverage Ingredients to be ordered for production BI is expressed in equation (3.5) as:

$$B_I = \sum_{i=1}^{6} B_x(i) * \beta + \mu \tag{3.5}$$

- **ß**: is a factor used to find the corresponding weight of ingredient based on the number of beverages to be produced at the factory.

The beverage company supplies bottles of 500ml. The total amount of bottles to be sold in a week gets converted into litres to find the equivalent amount of ingredient to be ordered from the supplier. Based on the recipe of the beverage in question (Which recipe remains confidential), the following relationship was established:

1 litre of water = 1kg of water

➔ If **Θ** liter of water = **η** kg of Concentrate (Ingredient for the beverage)

➔ 1 liter of water = $\frac{\theta}{\eta}$ Kg of concentrate

➔ For any **8** amount of water, the corresponding amount of ingredient ß will be defined in equation (3.6) as:

$$\text{ß} = \frac{\theta}{\eta} \times 8 \text{ Kg of concentrate} \tag{3.6}$$

If the daily number of bottles to be ordered is B_x, the conversion to find the corresponding amount of water needed **8**, knowing that each bottles contains **500 ml** and that **1 ml = 10^{-3} litre,** will be equal by equation (3.7) expressed as:

$$8 = 500 \times \text{Bx} \times 10^{-3} \tag{3.7}$$

- In equation (3.5) **µ** is a safety margin factor that will be added to the calculated beverage sale of the week to reduce possible error.

Note: To avoid cases where bottles and ingredients run out before end of production, the actual daily production (sale) is also included as another case into the production method:

- Let us consider Δ to be the actual daily sale of the production. The following observations are made:

  For each daily bottle count ordered $B_x \neq B_{SAT}$; B_{SAT} being the number of bottles ordered for the last day of production:

➔ If $\Delta > B_x \rightarrow$ Prepare an additional order ς for bottles and ingredients to supplier

➔ $\varsigma = \Delta - B_x$; the order will be placed through the server at the end of the day and compensate the stakeholders (bottles and ingredients) used.

The first section of the theoretical modelling was intended to find a mathematical correlation between parameters involved in the beverages' production to forecast production and get on time all necessary components. This was achieved through simple linear regression after successful collection of daily temperature versus daily sales. A mathematical expression representing this correlation was generated and several rules made based on the expression to later optimize plant's production with Industry 4.0 rules.

3.2 Motor vibration velocity threshold limits for predictive maintenance: theoretical modeling

As per Collin Sanders (Sanders C., 2011), the simple form of vibration is a single frequency system as shown in Figure 3-4:

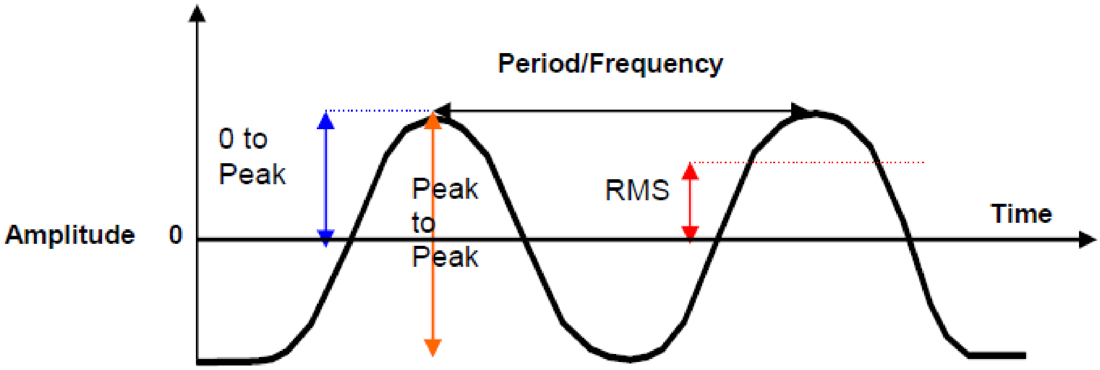
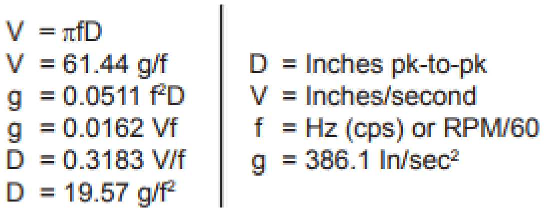


Figure 3-4: Vibration as a frequency system (Sanders C., 2011)

The velocity is the first derivative of displacement as a function of time; it is the rate of change in displacement (the speed of the vibration). Based on Figure 3-4, the displacement is equal to the amplitude of the vibration. The displacement or amplitude of a vibration is measured in inches, mils, micrometres or mm. The velocity or speed of vibration will therefore be measured in inches/sec or mm/sec. The acceleration is the second derivative of displacement; it is the rate of change of velocity (Keefer, 2008). Figures 3-5 shows some important relationship factors between sinusoidal velocity, acceleration and displacement.



$V = \pi f D$
$V = 61.44\ g/f$
$g = 0.0511\ f^2 D$
$g = 0.0162\ Vf$
$D = 0.3183\ V/f$
$D = 19.57\ g/f^2$

D = Inches pk-to-pk
V = Inches/second
f = Hz (cps) or RPM/60
g = 386.1 In/sec^2

Figure 3-5: Relationships of Sinusoidal Velocity, Acceleration and Displacement (American Environments Company, Inc., 1999)

As mentioned in the previous section, according to ISO 10816 there is an acceptable limit of vibration a motor shouldn't exceed when operating. Figure 3-6 shows a trend of machine vibration as per ISO 10816. When constantly exposed to undesired vibration, the motors lifespan is reduced and the chance of failures becomes high. One of the aims of our research study is to detect in advance all possible impairment due to motor vibration and schedule proper maintenance before failures occur. This is called predictive maintenance.

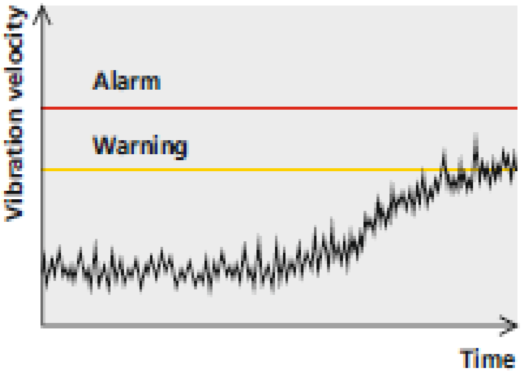


Figure 3-6: Machine vibration trend according to ISO 10816 (Ifm Electronics, 2013)

The size of the motor driving the conveyor will impact alarm and warning limits to the system. We will refer to Table 3-2 to select our alarms and warning thresholds that will be later on programmed in the PLC.

Table 3-2: Vibration Severity Criteria based on ISO 2372

RMS Overall Velocity Level in 1000 Hz Bandwidth		Vibration Severity Criteria			
mm/s	**In/s**	**Class I**	**Class II**	**Class III**	**Class IV**
0.28	0.01	Good	Good	Good	Good
0.45	0.02				
0.71	0.03				
1.12	0.04	Satisfactory			
1.8	0.07		Satisfactory		
2.8	0.11	Unsatisfactory		Satisfactory	
4.5	0.18		Unsatisfactory		Satisfactory
7.1	0.28	Unacceptable		Unsatisfactory	
11.2	0.44		Unacceptable		Unsatisfactory
18	0.71			Unacceptable	
28	1.10				Unacceptable
45	1.77				

Table 3-2 shows different vibration severity level as described by ISO IS2372: Good (displayed in green), Satisfactory (displayed in Yellow), Unsatisfactory (displayed in Orange) and Unacceptable (displayed in Red). The different classes on the table are divided as follows:

Class I: Small-sized machines (from 0 to 15KW)

Class II: Medium-sized machines (from 15 to 75KW)

Class III: Large-sized machines (powered > 75KW) mounted on “Rigid Support” structures and foundations

Class IV: Large-sized machines (powered > 75KW) mounted on “Flexible Support” structures.

There are many classifications for vibration severity criteria depending on applications and motors' sizes. In (Pereira R.R., 2016), one of those criteria is used for vibration analyses. To make operators' life easier, calculations and conditions to initiate the predictive maintenance of the conveyor motor is done in the background (in the PLC program). The only action that will be required from the operator is the selection on the SCADA system of the motor size range that will map it to the corresponding predictive maintenance rule.

Depending on one application to another, there are several types of safety, health criteria or classes that a system should respect to function properly. Table 3-2 is an example for one of them. The RMS vibration speed on Table 3-2 is read from a vibration sensor that gets mounted on the motor fin. The other side of the sensor is then connected to an intelligent controller (in our research to a Siemens S7-1200 PLC) or any other device that will convert the vibration speed into understandable digits. Figure 3-7 displays how the vibration sensor is mounted on a motor.

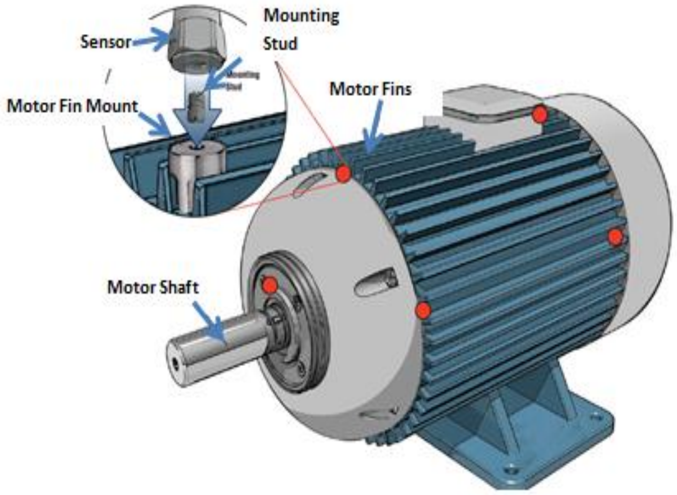


Figure 3-7: Vibration sensor mounted on motor
(IMI sensors, 2013)

Let's assume the vibration criteria of a system with **η number of classes** divided in **b ranges for health statuses** of a system:

- $b_1 – b_2$: healthy status
- $b_2 – b_3$: satisfactory status
- $b_3 – b_4$: intermediate status_1

- $b_4 - b_5$: intermediate status_1

•
•
•

- $b_s - b_m$: unsatisfactory status
- $b_m - b_n$: unacceptable status

Looking at the b ranges above as well as those on Table 3-2, it is very noticeable that the end of one range is the beginning of another, in other words two health ranges share a border. In mathematical this will be explained as follows:

$$(b_1 - b_2) \cap (b_2 - b_3) = \{b_2\},\ (b_2 - b_3) \cap (b_3 - b_4) = \{b_3\},\ \ldots\ ,(b_s - b_m) \cap (b_m - b_n) = \{b_m\}$$

For this project, we use the vibration criteria on Table 3-2 to program the predictive maintenance rule as follows:

The number of class ***Ƞ*** is equal to 4 and the number of health ranges is also equal to 4:

- Range $X_1 - X_2$→ Healthy status
- Range $X_3 - X_4$→ Satisfactory
- Range $X_5 - X_6$→ Unsatisfactory
- Range $X_7 - X_8$→ Unacceptable

The vibration velocity of the motor read from the vibration sensor by the PLC at a time ***t*** is equal to **V (t):**

Case 1:

If $V(t) = V(X_7 - X_8)$

→ Initiate reactive maintenance with immediate stop.

Case 2

If $V(t) = V(X_5 - X_6)$

→ Initiate predictive maintenance on next machine stop.

Case 3:

If V (t) = V ($X_3 – X_4$)

- ➔ V (temp) = $\sum_{t \in N} V[X_3 - X_4](t)$; with t time elapsed every hour.

  If V (temp) = {V($X_5 – X_6$) U V($X_7 – X_8$)}

→ Initiate predictive maintenance on next machine stop

The three cases above can be explained as follows:

Case 1 is not considered as a preventive maintenance rule because the range of velocity obtained is within the unacceptable range and requires immediate attention for safety of the whole system. It is therefore called reactive maintenance. Case 1 is also considered as an alarm of first degree.

Case 2, if the motor vibration velocity obtained at a specific time falls within unsatisfactory range, the system will call for a predictive maintenance at the next machine stop. The motor is supposed to always run in a healthy state. As soon as vibration starts to affect motor run, predictive maintenance needs to be done to recover healthy state.

Case 3, Prevention is always better than cure. When vibration starts to affect the motor, though still in the satisfactory range it is more likely to be converted into an unacceptable state later on. Case 3 detects velocity in the satisfactory range and sum every hour its value until equal to a velocity in the unsatisfactory and/or unacceptable range. In reality, the actual velocity vibration will not be equal to a value in the unacceptable yet, but this operation is done to prevent any future failure and recover healthy state.

Case 2 and 3 are considered as warnings. They are not as severe as case 1 but came back a danger if not attended to as soon as possible.

Using a vibration sensor that will be installed on the conveyor motor, we will collect directly vibration velocity data to the S7-1200 Siemens PLC; vibration sensor being connected as 4 – 20mA analogue input.

3.3 Tracking production output as a time function: theoretical modeling

The minimum number of machines needed in a production or manufacturing system is given on equation (3.8) as follows (Kar D. &Singh D. Kr, 2015):

$$N = \frac{Q \times t}{\frac{\min}{day} avail \times Mach..\mathrm{Re}\, l} \qquad (3.8)$$

Where:

$N =$ Minimum number of machines for production

$Q =$ Daily production in parts/day

$t =$ Total machining time per parts

$Mach.\mathrm{Re}\, l =$ Machine Reliability - percentage of successful operation with no errors or faults

$\frac{\min}{day} avail$ = amount of time that the machine runs daily or is available.

One of the aims of our research study is to develop a bottling process that is able to push the production output very close to the daily target despite inevitable faults and failures. From equation (3.8), let's pull out another equation (3.9) that is the daily production Q:

$$Q = \frac{N \times \frac{\min}{day} available \times Mach.\mathrm{Re}\, l}{t} \qquad (3.9)$$

Analysing the variables of equation (3.9) based on a bottling process; N representing the minimum number of machines involved in a bottling process is usually equal to 1; 1 being the minimum number of machines a SME bottling plant can afford to have for its operations. In fact, in many bottling SMEs, only a single machine is used to fill, cap and package bottles.

In real life, Machine reliability, the percentage of successful operation with zero error cannot be really controlled or given in advance. This is a parameter that we unfortunately cannot manipulate. The machining time per bottle can also be known but is very much tight to the machine reliability, in other words quite difficult to control. The only factor remaining is the amount of time that the machine runs daily for production. This variable is flexible since it's up to operators to start the system at the beginning of the working hours and to stop it at the end of the day. Looking at equation (3.9) we see that Q, the daily production is directly proportional to min/day available, the time of daily machine run. In simple words, the higher daily machine run, the higher the daily production. Increasing the daily machine run time implies working beyond the machine normal hours that is working overtime.

Our work will focus in efficiently increasing the system working hours in order to meet the daily target, avoiding wastage of resources: time, money and energy. The decision to start an efficient overtime run is only feasible when tracking closely the production and comparing it to the daily target. Our design quantifies the production in terms of an hourly feedback. Below are the variables involved in the design:

Q = production target bottles/day;

E_T = Entered Target: bottles/day;

For programming purpose, production target bottles/day and Entered target are made equal as per equation (3.10):

$$Q = \mathrm{E}_\mathrm{T} \tag{3.10}$$

$H_T =$ Hourly Target in bottles/hour

m = Number of Normal Working Hours: hours/day. The Hourly target is calculated by equation (3.11) as:

$$H_T = \frac{Q}{m} \tag{3.11}$$

$(t_0 + t) =$ (Start-up time (t_0) + number of hours): a specific hour; t $\in \mathbb{N}$

$CC_{(t_0+t)}$ = Current Count of bottles from start-up to a specific hour of the day: bottles

$HC_{(t_0+t)}$ = Bottles produced (Counted) for an hour @ a specific hour of the day

OP = Overtime Percentage

ORS = Overtime Run Start Condition

$AHC_{(t_0+t)}$ = After Hour Count @ a specific hour of the day

f = Final or last normal working hour

$CC_{(f)}$ = Current Count of bottles at the last working hour of the day

In the PLC, we will program the hourly count of bottles to keep close track of the production. At each specific hour during the system working hours, the count will be computed and saved as follows:

$$HC_{(t_0)} = CC_{(t_0)}$$

$$HC_{(t_0+1)} = CC_{(t_0+1)} - HC_{[t_0+(1-1)]} = CC_{(t_0+1)} - HC_{(t_0)}$$

$$HC_{(t_0+2)} = CC_{(t_0+2)} - HC_{[t_0+(2-1)]} = CC_{(t_0+2)} - HC_{(t_0+1)}$$

$$HC_{(t_0+3)} = CC_{(t_0+3)} - HC_{[t_0+(3-1)]} = CC_{(t_0+3)} - HC_{(t_0+2)}$$

$$\vdots$$

$$HC_{(t_0+10)} = CC_{(t_0+10)} - HC_{[t_0+(10-1)]} = CC_{(t_0+10)} - HC_{(t_0+9)}$$

$$\vdots$$

$$HC_{(t_0+t)} = CC_{(t_0+t)} - HC_{[t_0+(t-1)]}$$

$$\vdots$$

$HC_{(t_0+f)} = CC_{(t_0+f)} - HC_{[t_0+(f-1)]}$; Hourly count at the final normal working hour

$$\vdots$$

$$HC_{(t_0+t)} = \sum_{t\in N}[CC(t_0+t)]][-HC[t_0+(t-1)]] \tag{3.12}$$

In simple words, formula (3.12) can be explained as: the hourly count at a specific hour is equal to the current count at that hour minus the previous hour count. The total count of daily bottles is live in the PLC through sensor feedback but can theoretically be calculated by equation (3.13):

$$CC_{(f)} = \sum_{t\in N}^{m} HC(t_0+t) \tag{3.13}$$

Knowing the production count at each hour as well as at the final working hour, the condition for overtime run to meet daily target can be computed as equation (3.14):

$$ORS = E_T - [(E_T \times OP)] \tag{3.14}$$

If

$$CC_{(f)} \leq ORS \tag{3.15}$$

Equation (3.15) will apply the parameters for Overtime at full speed and make Overtime Run to Start.

During overtime, our design will still track at the hourly production to know exactly when to stop the system (unless forced by the operators) and to avoid wasting resources by over running the machine. The after-hours hourly bottles count expression is similar to formula (3.12) and is calculated by equation (3.16):

$$AHC_{(t_0+t)} = \sum_{t\in N}\left[CC(t_0+t)\right]\left[-HC\left[t_0+(t-1)\right]\right] \tag{3.16}$$

The total count of production from start-up shift including overtime is calculated in equation (3.17) as follows:

$$\sum_{t\in N}\left[AHC(t_0+t)\right]\left[CC_{(f)}\right] \tag{3.17}$$

If Equation $(3.17) \geq Q$

Overtime at Full Speed Run stops, parameters automatically back to normal conditions.

The theoretical model above shows the relationship that exists between the daily machine working hours and the daily production at specific time of the day. Computing the right conditions for the operation of a bottling process based on these expressions is the heart of our new strategy. In the next section, we will transform these mathematical expressions into programming codes and configuration in the Siemens S7-1200 PLC to generate an automatic response for the bottling process.

Note: In this research, Q, the amount of bottles targeted will either be equal to the forecasted production value B_N computed in (3.1) and (3.4) or the entered targeted value on the HMI E_T depending on the one with a greater value. This is done in order to always have a good planning and avoid running short of production stakeholders. The process flow chart on the next chapter explains how this is performed in the programming process.

The aim of this chapter was to develop the theoretical modelling of the high-tech automated bottling process. The optimal theoretical modelling involved:

Generation of a mathematical expression to predict plant production through simple linear regression; this was achieved by collecting daily temperature versus daily sales and finding the correlation between them. This expression is used later in the PLC to acquire on time all production stakeholders and avoid downtime due to material shortage.

Establishment of predictive maintenance and early fault detection rules for critical components of the bottling plant such as motors and sensors; this was achieved by installing smart industry 4.0 sensors that would collect important data from the critical devices and also by analyzing vibration severity criteria of different types of motors and stating cases or rules that would inrush predictive maintenance within the plant.

Reduction of human intervention within the plant’s operation by establishing rules for automatic configuration of parameters based on several cases and scenario; and by closely tracking the production output as a function of time.

Chapter 4 : HIGH-TECH AUTOMATED BOTTLING PROCESS: IMPLEMENTATION

4.1 High-Tech automated bottling process overall architecture

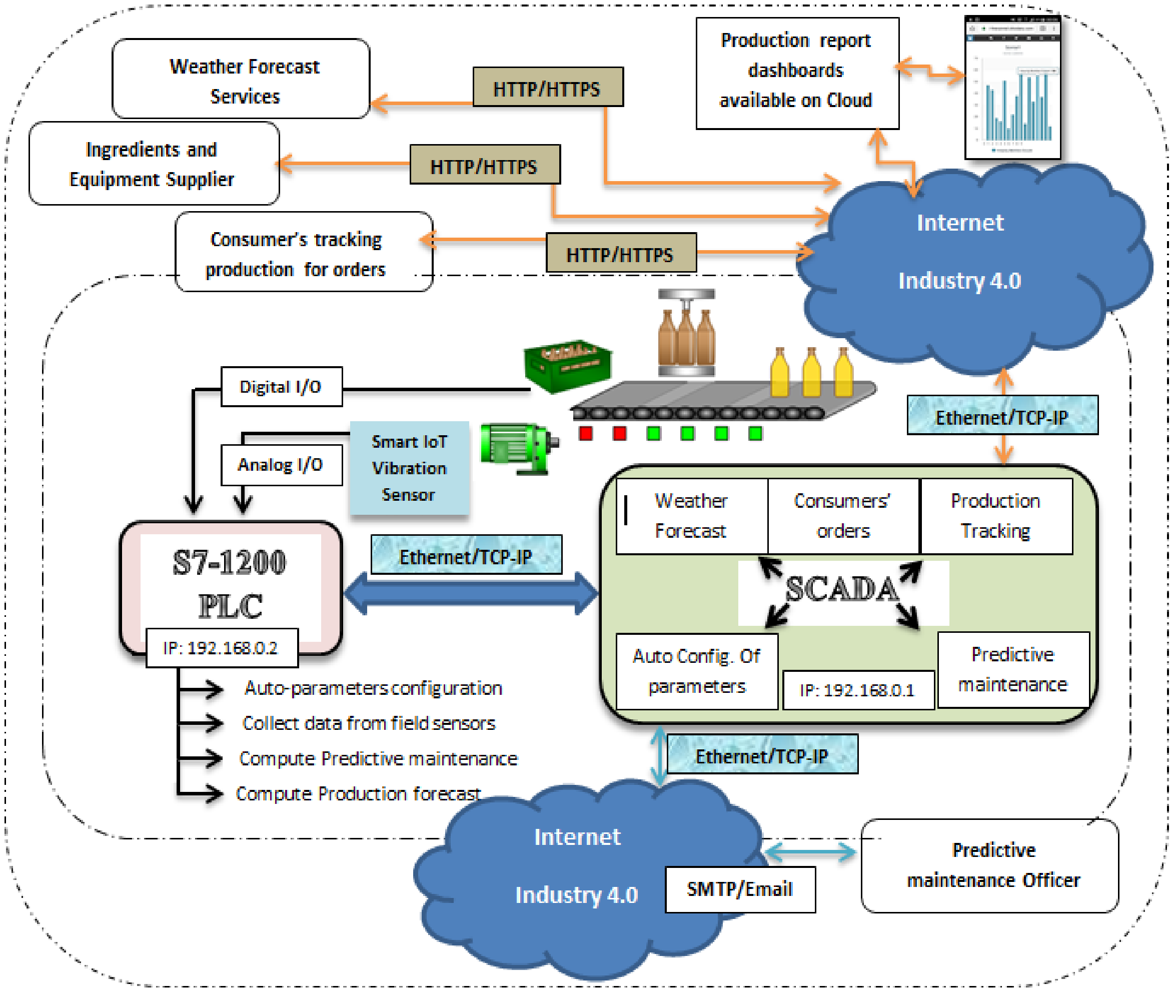


Figure 4-1: High-Tech automated bottling process overall system's architecture

Figure 4-1 is an illustration or a summary of all the components and all the functions that the high-tech automated bottling process has. The two main hardware components in the architecture are the PLC and the SCADA. The PLC is acting as the primary brain of the system because it computes, controls, monitors all useful functions for the operation of the bottling plant and all mathematical expression: linear regression formula used for production forecast, predictive maintenance rules and production output tracking, associated to process' optimization. The PLC transmits to the SCADA system all important information that users, operators, supervisors will need to closely monitor and control the bottling plant through the GUI. The communication between PLC and SCADA is done by TCP/IP protocol using their IP addresses and an Ethernet cable to link them.

The SCADA can be considered as the secondary brain of the system. As already mentioned, it is the graphical platform through which users interact with the system. It displays number of bottles forecasted for the week ahead; current bottles production count; overtime bottles count if any and total bottles count. It displays predictive maintenance schedule when generated and allows maintenance technicians to easily configure new maintenance rules based on motor’s size. The SCADA software is running on a panel pc used in this project as a gateway between internet application and the bottling plant. This is one of the Industry 4.0 standards our research brings in: an interconnection between bottling plant and internet applications as well as decentralized access of system’s information. E-mail notification to maintenance experts, beverages orders to customers or to suppliers, weekly weather forecast scripts are running on the panel PC and interact with the PLC and/or the SCADA.

The PLC is also directly connected to some field devices like vibration sensors mounted on conveyor motors as indicated on Figure 4-2. The vibration sensor is connected to the PLC through its analogue input card. The PLC reads analogue values and converts them to understandable vibration speed that will be used to initiate predictive maintenance of the motor.

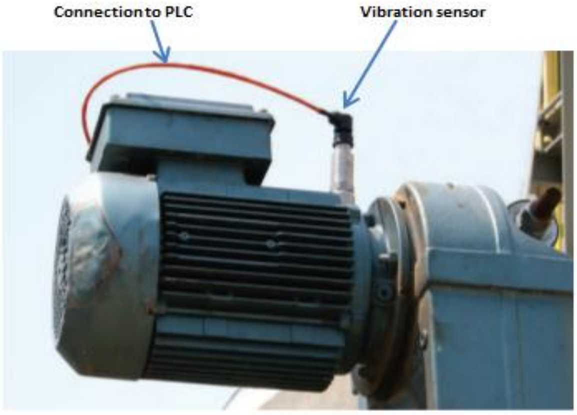


Figure 4-2: Vibration sensor mounted on motor and connected to PLC (IMI sensors, 2013)

The architecture of the overall system on Figure 4-1 takes into account the three values I40 concepts emphasize on predictive maintenance as per Kohler M (Kohler M, 2016). In our research, the three values are contextualized as follows:

- The Siemens S7-1200 PLC reads real-time data from the vibration sensor seating on the conveyor motor connected as an analogue input. This data is then processed in the PLC processor with the programmed predictive maintenance rule to generate proper actions. The PLC is also communicating real-time with a SCADA system that displays on its GUI actions to be taken. The predictive maintenance rule is programmed in a flexible way, gathering for easy future changes and editing of the system by technicians and maintenance team without necessarily experts intervention. This description relates to the real-time condition monitoring.

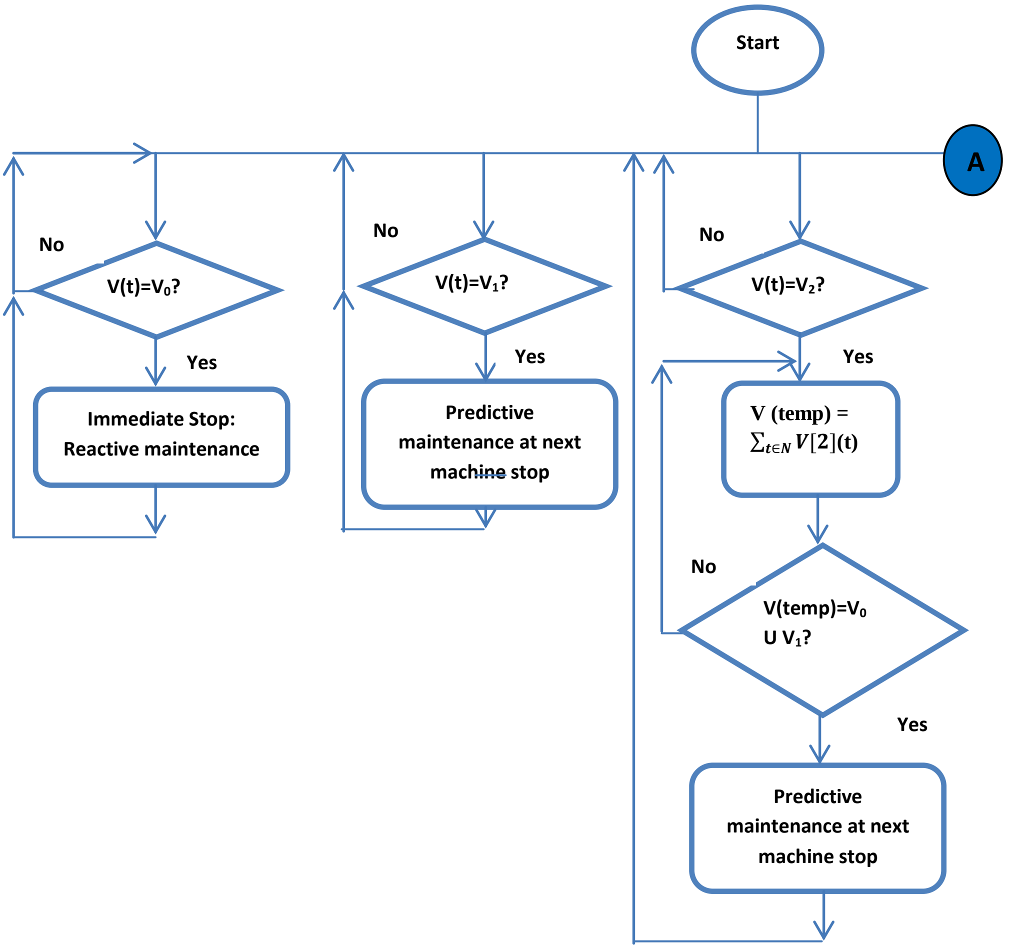
- When the maintenance team or the technicians are ready to configure a new rule for a specific motor, they only select the right class range of motor size and enter it into the PLC processor via the SCADA interface. The program received the value entered in the buffer, filter the entered data to make sure it falls within the acceptable range of value to enter, match it using a mapping table to the corresponding class of motor size and then apply the rules to the PLC program. With this structure in place, there is no need for an expert to develop again very complicated programs when a new rule needs to be edited or configured. The flexible evaluation and analysis option is offered by this section.

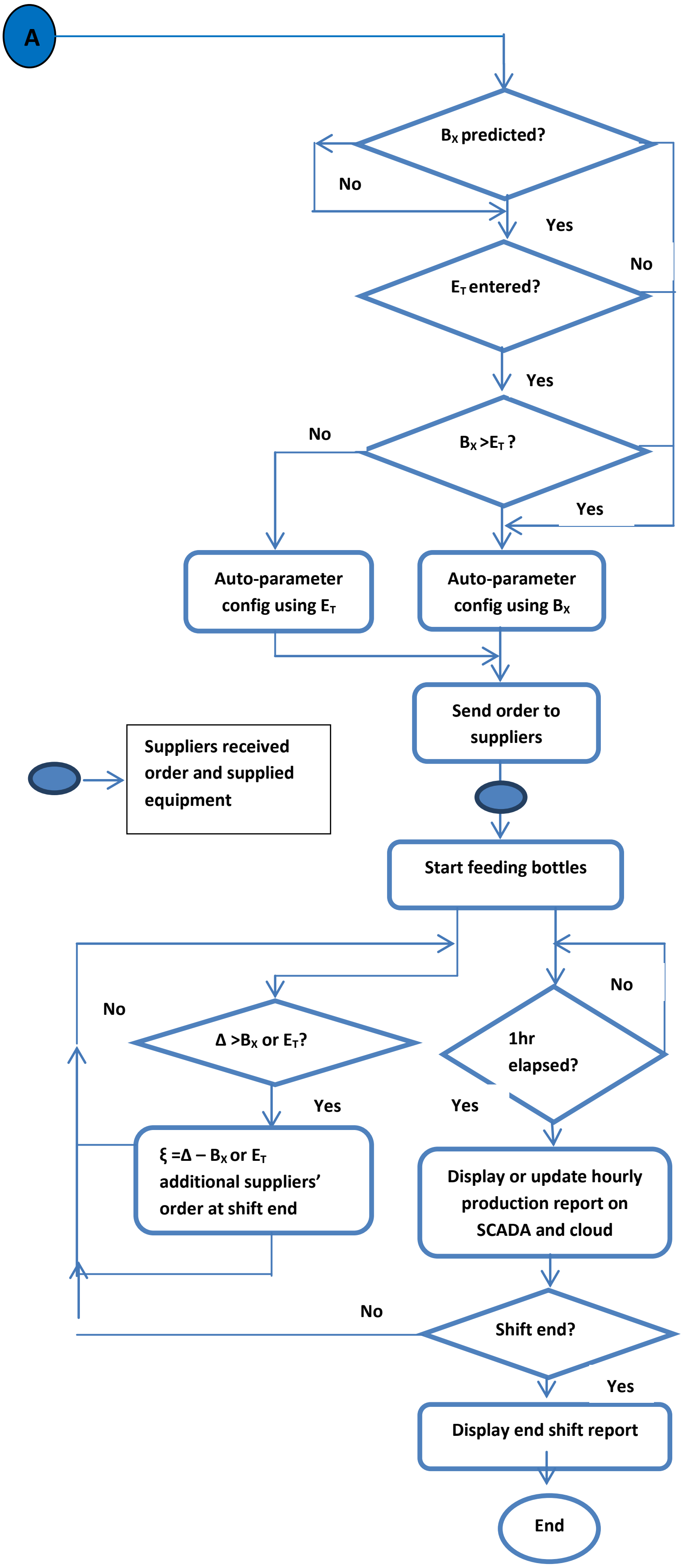
- As soon as a predictive maintenance rule has been generated, the system directly contacts via SMTP email, the expert in charge of maintaining the motor without waiting for him to rely on the central SCADA information. Traditional standards only used central PLCs and SCADAs to convey any important message: faults, alarms, and statuses to users or operators. In other words to be aware of system health statutes, one had to be physically in front of the SCADA system or connected to the PLC to monitor important information; this could result in unnecessary delays and late reactions to important system errors. Our research study is solving this issue by transmitting real-time the predictive maintenance schedule once generated to the expert in charge for him to take actions as soon as possible. This section takes care of giving Targeted notification of experts.

4.2 Overall process flow chart

The overall process flow chart is designed based on mathematical models and functions implemented in the previous chapter.The flow chart uses the following parameters:

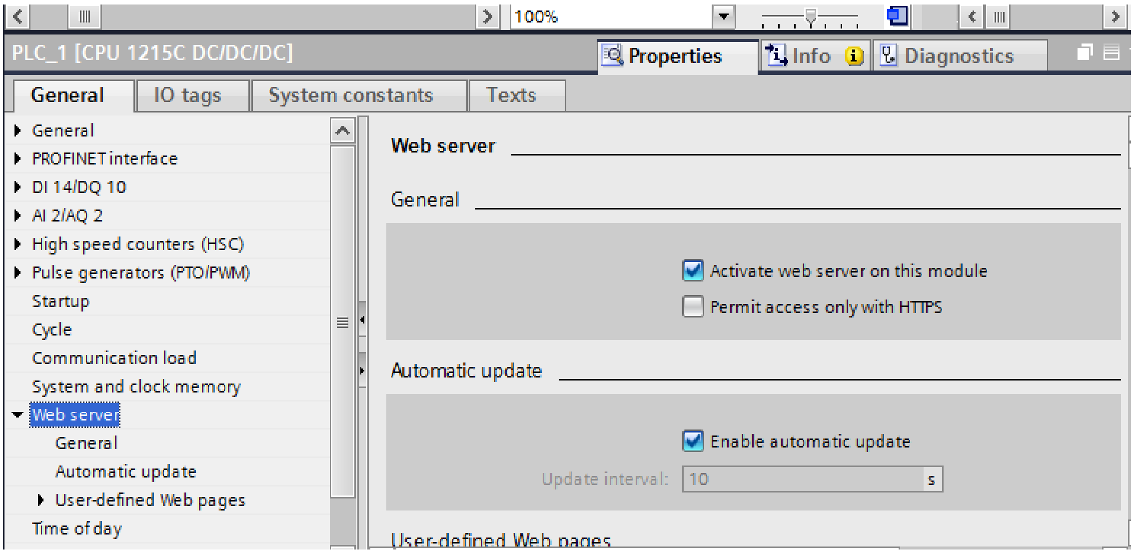
- E_T = Daily targeted bottles entered
- B_X = Predicted daily bottles target
- Δ = Actual daily bottles count
- ξ = Difference between targeted production and actual production
- $V(t) = V(X_7 – X_8) = V_0$= Unacceptable motor vibration speed range
- $V(t) = V(X_5 – X_6) = V_1$= Unsatisfactory motor vibration speed range
- $V(t) = V(X_3 – X_4) = V_2$ = Satisfactory motor vibration speed range




A
Bx predicted?
No
Yes
ET entered?
No
Yes
No
Bx >ET ?
Yes
Auto-parameter config using ET
Auto-parameter config using Bx
Send order to suppliers
Suppliers received order and supplied equipment
Start feeding bottles
No
Δ >Bx or ET?
Yes
ξ =Δ – Bx or ET additional suppliers' order at shift end
No
1hr elapsed?
Yes
Display or update hourly production report on SCADA and cloud
No
Shift end?
Yes
Display end shift report
End

4.3 S7-1200 PLC configuration for interconnection to internet services

The Siemens S7-1200 PLC has a web server function that allows it to be accessible through a web browser using its IP address. This function makes possible for a PLC to be integrated into an internet network with multiple other devices bearing IP addresses. Enabling this function is done in the PLC hardware configuration on the appropriate web server menu as displayed on Figure 4-3.



☑ Activate web server on this module
☐ Permit access only with HTTPS
☑ Enable automatic update
Update interval: 10 s

Figure 4-3: S7-1200 Siemens PLC web server functions activation

A proper download of the PLC hardware configuration to the actual PLC needs to be done every time changes are made on the PLC configuration.

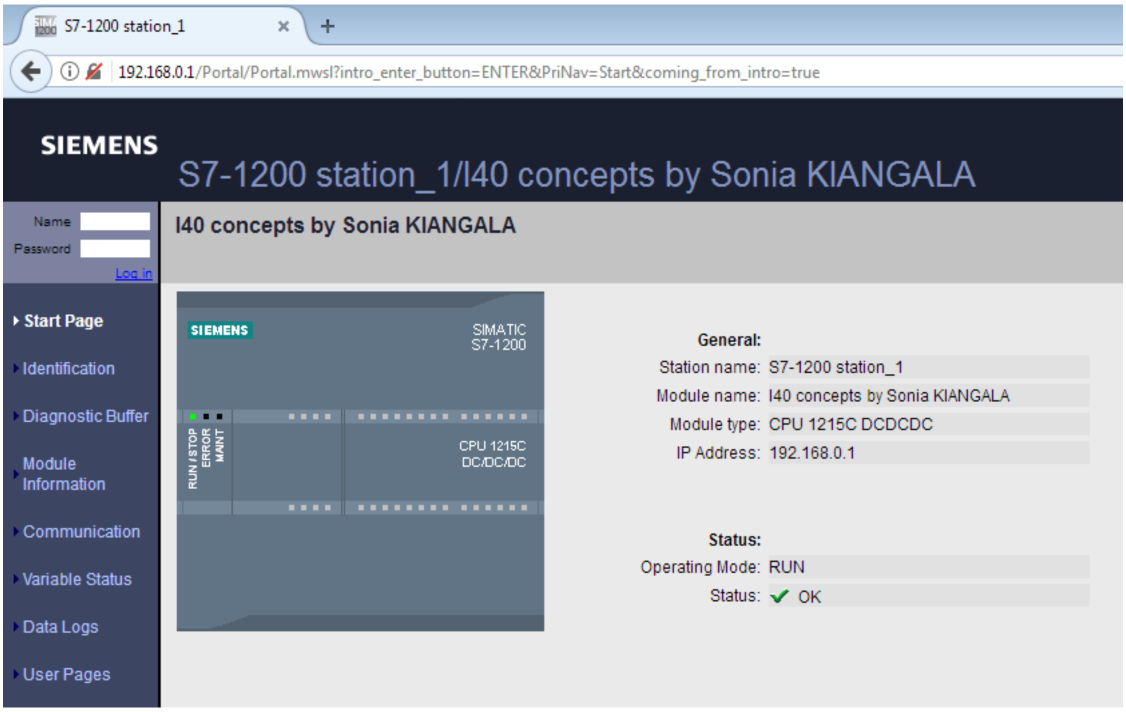
For this research our PLC IP address is: 192.168.0.1**.** A couple of sub functions are also available when accessing the PLC through its web server:

- **Identification**: view PLC order number, serial number, firmware version running on the PLC;
- **Diagnostic buffer**: check the PLC statuses, errors and faults; module information: view PLC and cards types connected onto the rack;
- **Communication**: view PLC MAC address, IP address, Ethernet port speed and statuses, data packages sent, received etc.;
- **Variable status**: monitor any PLC variable address.
- **Data logs**: Another very important web server function used in this research is the “Data logs” function which allows saving any data from the PLC time based into a

csv file format and accessible anytime for data analyses. The python script that will be used to send email notification to experts will be reading PLC data through this csv file;

- **User pages**: via few HTML configuration and programming blocks, read and/or write data from a webpage to/from the PLC.

Figure 4-4 is the S7-1200 PLC web server dashboard with all its sub functions as described above:



S7-1200 station_1/I40 concepts by Sonia KIANGALA

I40 concepts by Sonia KIANGALA

General:
Station name: S7-1200 station_1
Module name: I40 concepts by Sonia KIANGALA
Module type: CPU 1215C DCDCDC
IP Address: 192.168.0.1

Status:
Operating Mode: RUN
Status: ✓ OK

Figure 4-4: S7-1200 Siemens PLC Web server functions

As per the above web server functions, the data logs will be effectively used in this research for the creation of csv files. Data written in these csv files are the paths through which the PLC will be interacting with some internet functions like python script for e-mail notifications, production data accessibility in the cloud database etc. In order to generate csv files from the PLC, the following function blocks need to be configured and called in the PLC program:

- **Data Log create block:** This function generates the actual log file when instructed by the PLC program. Figure 4-5 shows the data log create block.

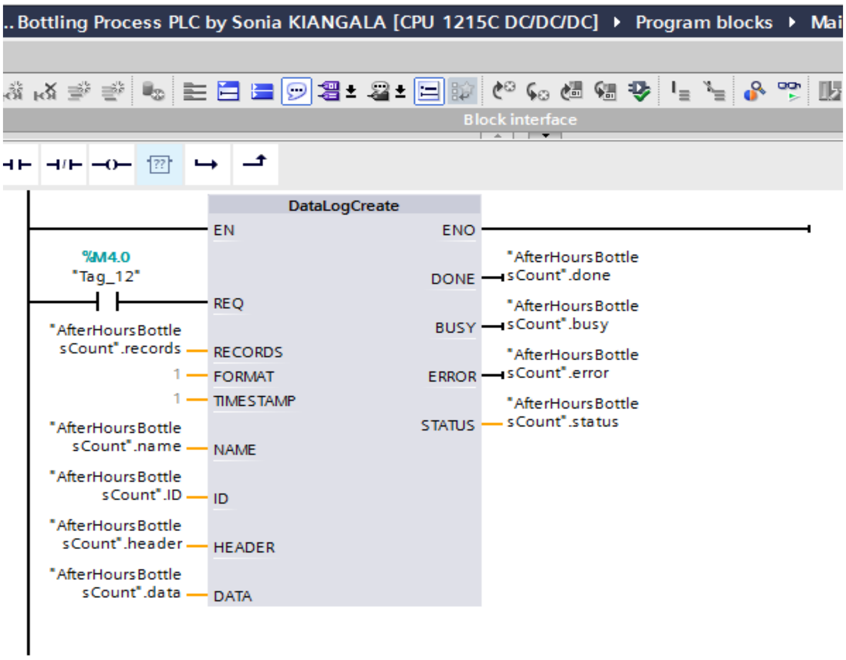


Figure 4-5: PLC data log create function called

As observed on Figure 4-5, some important parameters are to be configured on the function:

- **RECORDS**: Determine the number of entries the log file will have.
- **FORMAT**: Determine the format of the file, in this research csv format is used.
- **TIMESTAMP**: Choose whether or not to include date and time in the file.
- **NAME**: Give a name to the file
- **ID**: This is the unique identity of each created log file. This is very important when manipulating (opening or writing) more than one log file. The specific ID of each item will have to be used for operations to succeed.
- **HEADER**: Determine names of headers in the log file
- **DATA**: Point to address location of data to be saved in the log file

After creating a log file, one should be able to write and/or open it. This is done using two other functions similar to the data log create**: Data log write is displayed on Figure 4-6 and Data log open is displayed on Figure 4-7.**

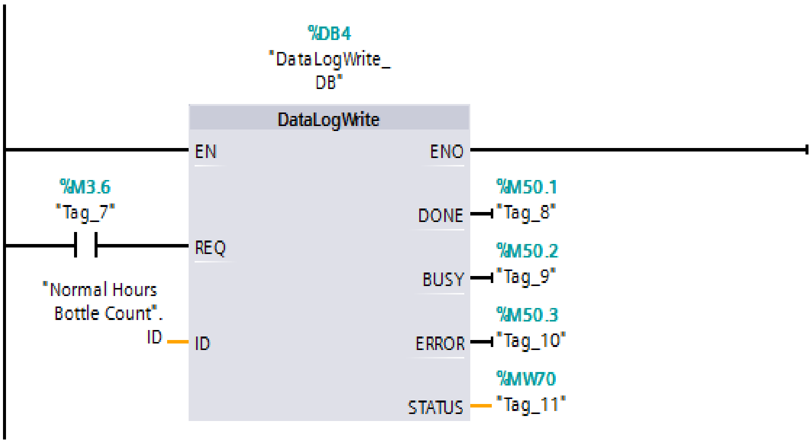


Figure 4-6: PLC data log write function called

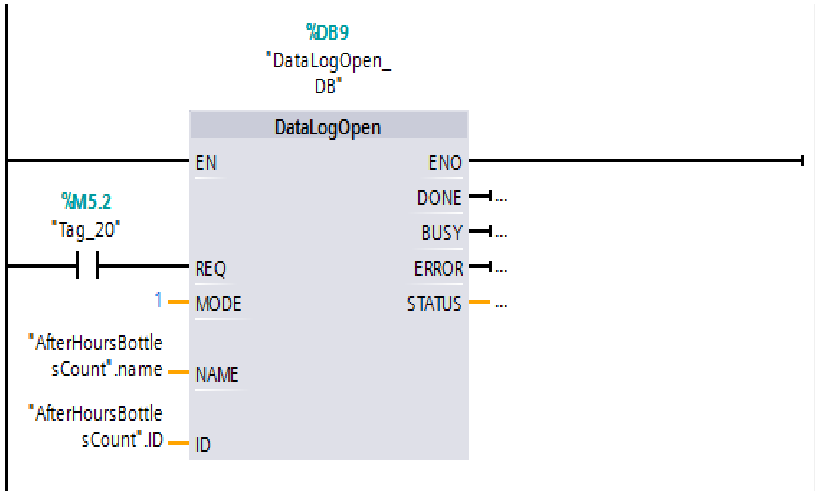
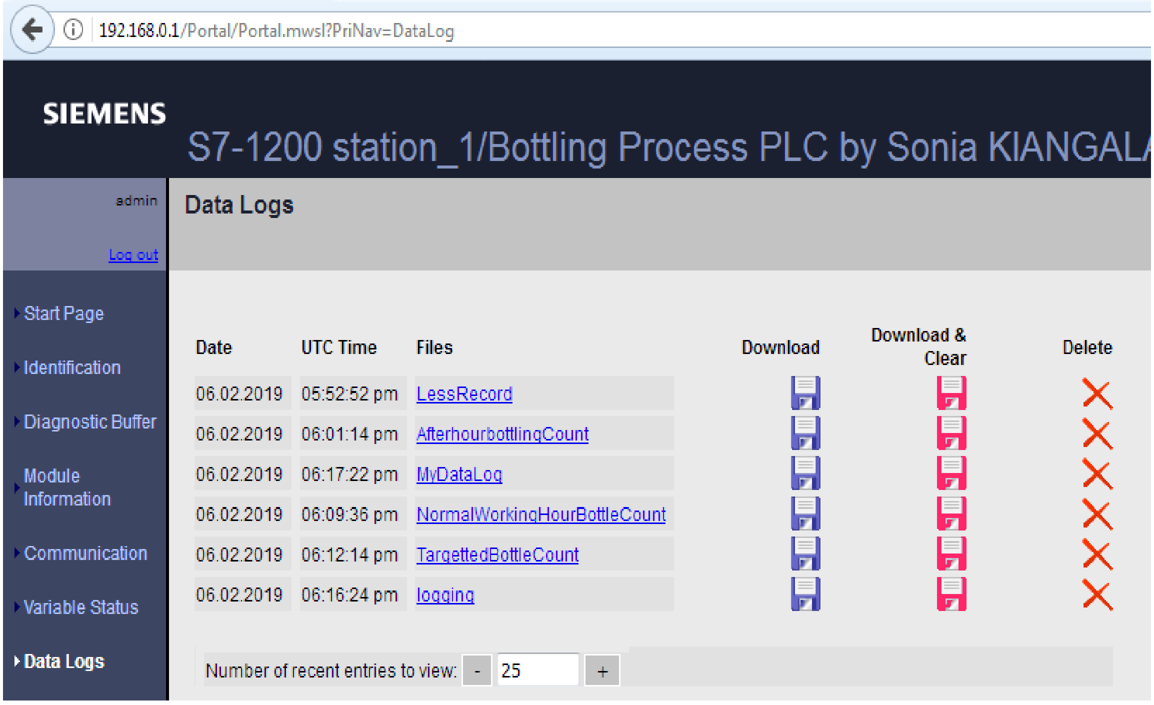


Figure 4-7: PLC data log open

The two main parameters that are used to write in the file and to open it are the: file name and the file ID that should match the ones of the created log file.

Figure 4-8 is a view of all the data log files created in the PLC after using all the above functions:



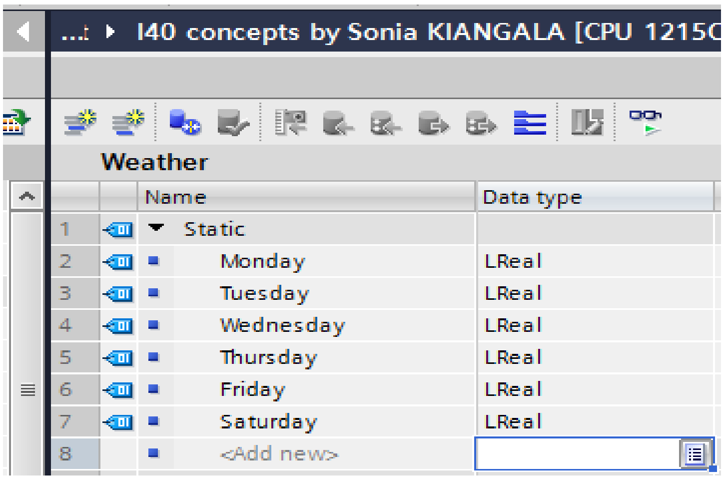
Data Logs

Date	UTC Time	Files	Download	Download & Clear	Delete
06.02.2019	05:52:52 pm	LessRecord			✕
06.02.2019	06:01:14 pm	AfterhourbottlingCount			✕
06.02.2019	06:17:22 pm	MyDataLog			✕
06.02.2019	06:09:36 pm	NormalWorkingHourBottleCount			✕
06.02.2019	06:12:14 pm	TargettedBottleCount			✕
06.02.2019	06:16:24 pm	logging			✕

Number of recent entries to view: - 25 +

Figure 4-8: S7-1200 PLC data logging csv files created

The web server function does not only limit in the creation of log files but also allow the PLC to be accessible in a web browser, to edit values of certain variables in websites written in HTML codes, to receive instructions or values from HTML websites etc. From figure 4 – 4 displaying all the S7-1200 web server functions, this option is available under the user pages tab (after proper configuration).In this research, we will use the HTML function to collect weekly weather forecast to predict weekly production for our bottling process. Before getting to the HTML configuration part, **η**number of variables will be created in the PLC database as displayed on Figure 4-9. **η** represents the number of weekday's production for which production need to be forecast. In our case **η** is equal to 6; production being from Monday to Saturday:

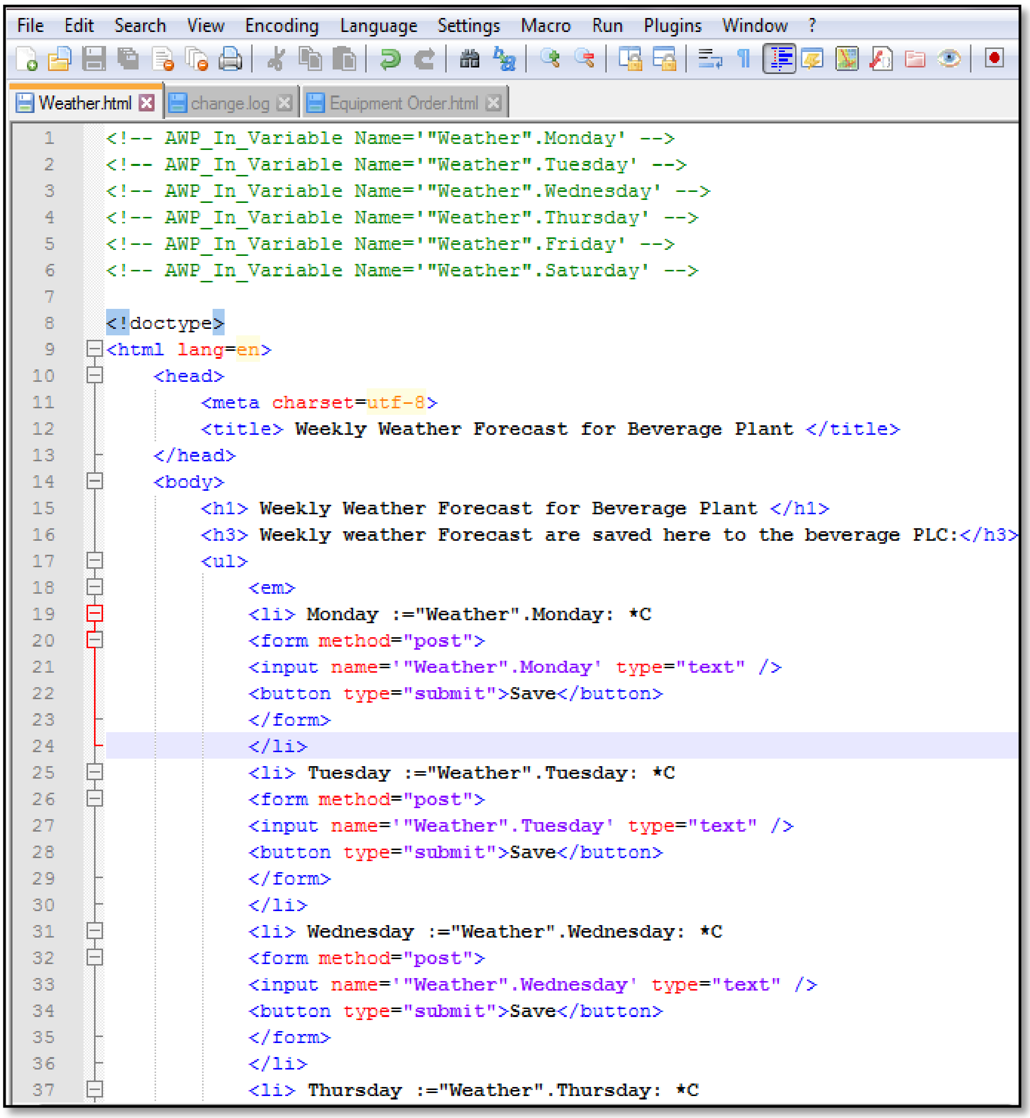


Weather

	Name	Data type
1	▼ Static	
2	Monday	LReal
3	Tuesday	LReal
4	Wednesday	LReal
5	Thursday	LReal
6	Friday	LReal
7	Saturday	LReal
8	<Add new>	

Figure 4-9: Weather forecast database variable

Variables have been created of type LREAL (Long Real) to accommodate the decimal places used in the regression equation (3.1). The HTML configuration to access web pages information starts with the creation of an HTML file as shown on Figure 4-10 for the user page of extension .html. Our file will be called **weather.html** and will contain the following:



```html
<!-- AWP_In_Variable Name='"Weather".Monday' -->
<!-- AWP_In_Variable Name='"Weather".Tuesday' -->
<!-- AWP_In_Variable Name='"Weather".Wednesday' -->
<!-- AWP_In_Variable Name='"Weather".Thursday' -->
<!-- AWP_In_Variable Name='"Weather".Friday' -->
<!-- AWP_In_Variable Name='"Weather".Saturday' -->

<!doctype>
<html lang=en>
    <head>
        <meta charset=utf-8>
        <title> Weekly Weather Forecast for Beverage Plant </title>
    </head>
    <body>
        <h1> Weekly Weather Forecast for Beverage Plant </h1>
        <h3> Weekly weather Forecast are saved here to the beverage PLC:</h3>
        <ul>
            <em>
            <li> Monday :="Weather".Monday: *C
            <form method="post">
            <input name='"Weather".Monday' type="text" />
            <button type="submit">Save</button>
            </form>
            </li>
            <li> Tuesday :="Weather".Tuesday: *C
            <form method="post">
            <input name='"Weather".Tuesday' type="text" />
            <button type="submit">Save</button>
            </form>
            </li>
            <li> Wednesday :="Weather".Wednesday: *C
            <form method="post">
            <input name='"Weather".Wednesday' type="text" />
            <button type="submit">Save</button>
            </form>
            </li>
            <li> Thursday :="Weather".Thursday: *C
```

Figure 4-10: PLC Weather forecast web page HTML code

The file on Figure 4-10 contains very simple HTML code created based on the 6 variables previously created in the PLC database. The only thing that might be new to HTML web designers is the top comment part:

<!-- AWP_In_Variable Name='"Weather".Monday' -->

<!-- AWP_In_Variable Name='"Weather".Tuesday' -->

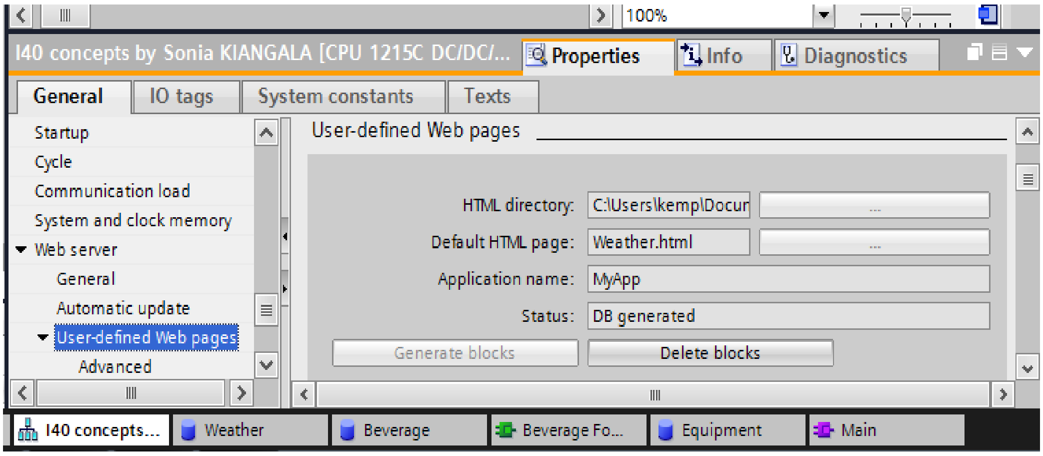
<!-- AWP_In_Variable Name='"Weather".Wednesday' -->

<!-- AWP_In_Variable Name='"Weather".Thursday' -->

<!-- AWP_In_Variable Name='"Weather".Friday' -->

<!-- AWP_In_Variable Name='"Weather".Saturday' -->

These comments have been added for the PLC to allow these variables to be externally modified; without these comments in the HTML file, variables will not allow any changes. After the file has been created, a path is added to the PLC hardware configuration to access it as displayed on Figure 4-11:



HTML directory: C:\Users\kemp\Docur

Default HTML page: Weather.html

Application name: MyApp

Status: DB generated

Figure 4-11: PLC hardware configuration for HTML webpage function

An important block in the PLC to enable web page services is the “**www**” block that needs to be called in the program to allow use of HTML web pages. Figure 4-12 shows how the block is called in the PLC program.

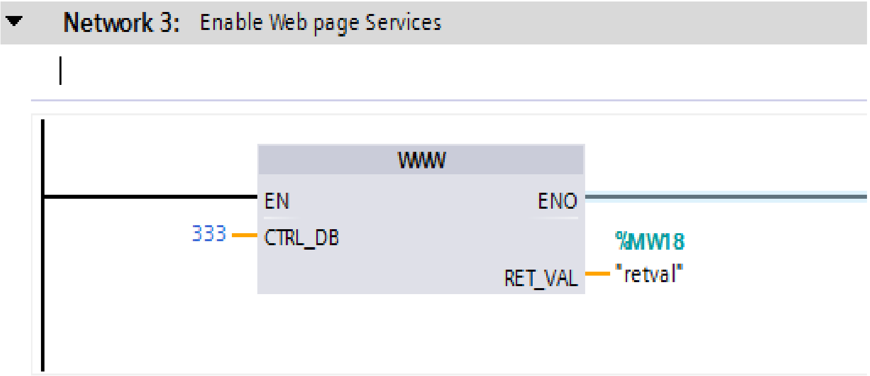


Figure 4-12: www function block to enable web services on PLC

Note: For security purposes, a user will need to log into the PLC web server application before making any change into the variables. The login portal is displayed on Figure 4-13.

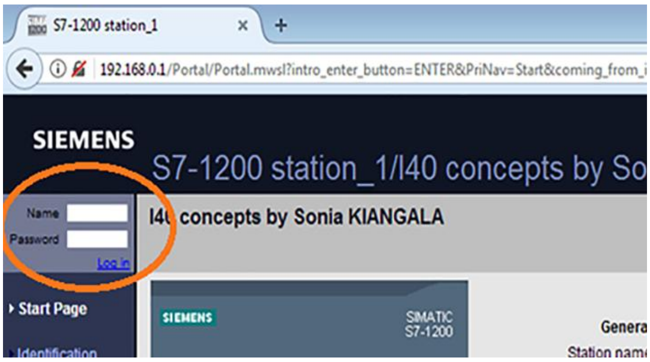
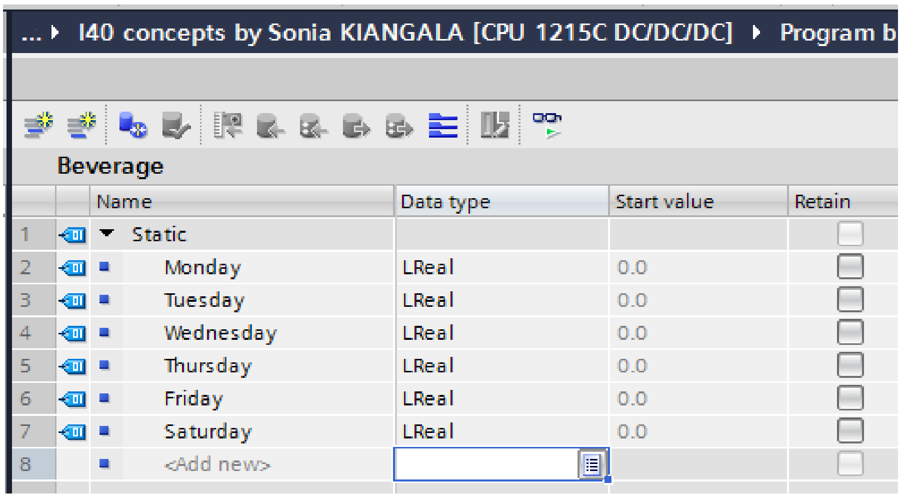


Figure 4-13: PLC web server Log in window

4.4 S7-1200 PLC program key routines and functions for design of high-tech automated bottling process

4.4.1 Bottling production forecast using linear regression formula

In chapter 3, the regression equation (3.1) for production forecast was developed. Below is its programming representation in PLC ladder language. Figure 4-14 is another database similar to the weather forecast one, created in Figure 4–9, added in the PLC to record beverage production forecast for each day of the week.

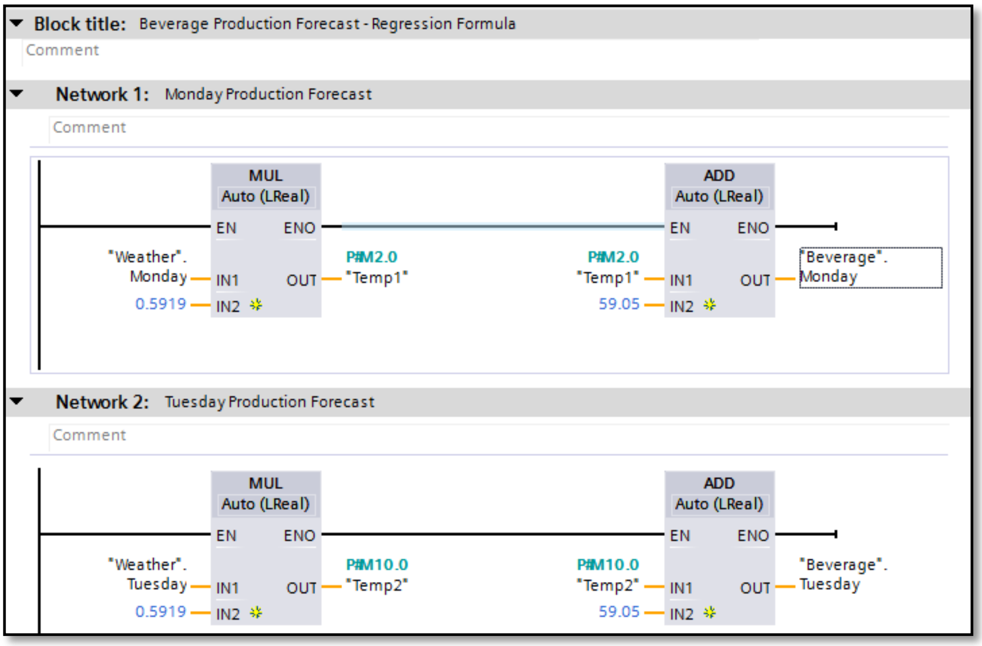


Beverage

		Name	Data type	Start value	Retain
1		▾ Static			☐
2		▪ Monday	LReal	0.0	☐
3		▪ Tuesday	LReal	0.0	☐
4		▪ Wednesday	LReal	0.0	☐
5		▪ Thursday	LReal	0.0	☐
6		▪ Friday	LReal	0.0	☐
7		▪ Saturday	LReal	0.0	☐
8		▪ <Add new>			☐

Figure 4-14: Production forecast database variables

Figure 4-15 is a representation of the regression equation (1) in PLC ladder programming language. The illustration is done for Monday and Tuesday only but the same PLC code applies for each production from Monday to Saturday.



Block title: Beverage Production Forecast - Regression Formula

Comment

Network 1: Monday Production Forecast

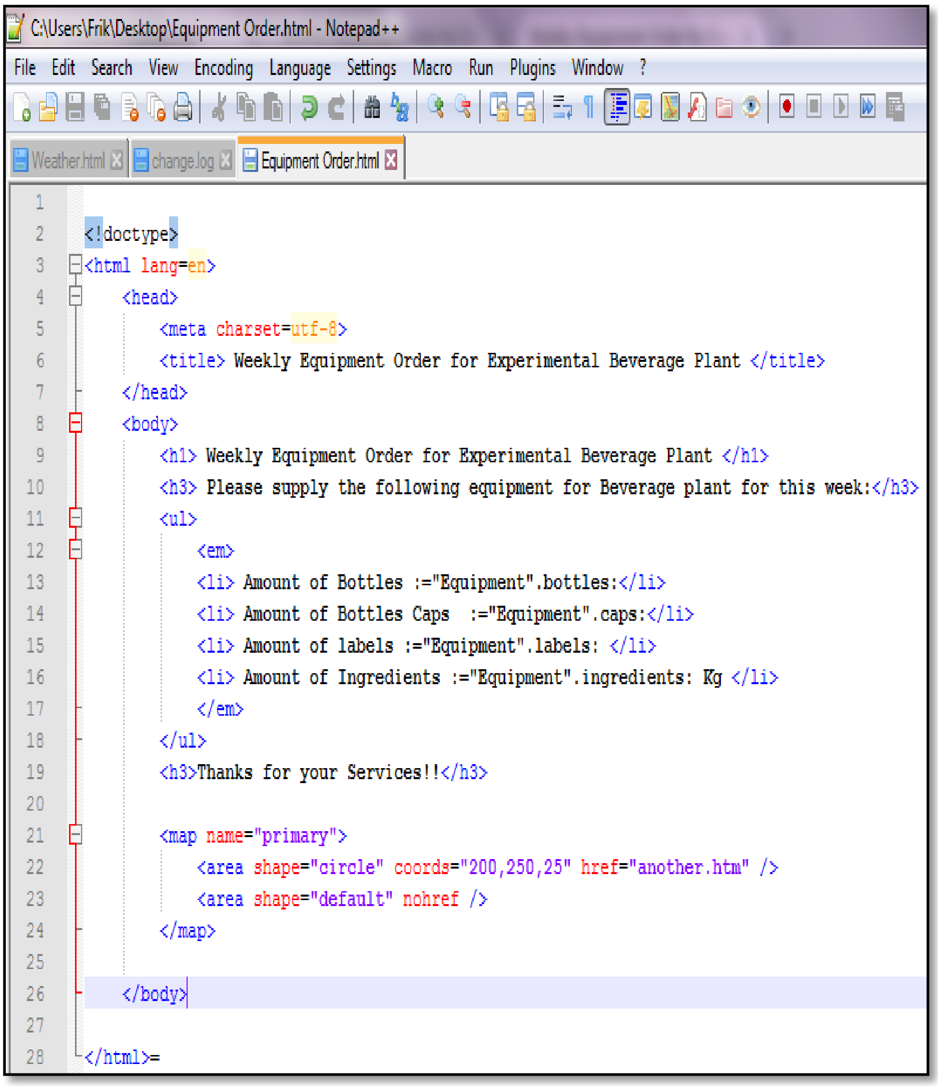
Comment

Network 2: Tuesday Production Forecast

Comment

Figure 4-15: Regression equation in PLC ladder programming language

As we can see on Figure 4-15, the input to each block is one the variables created on the database for the weather forecast and the output is saved into the beverage forecast database. In the result section, we will see how the readings are done from the weather forecast webpage to the PLC data block. The results from the regression formula are the production forecast for each day of the week: from Monday to Saturday. These results are later being used through equations (3.4) and (3.5) to order equipment and ingredients from respective suppliers. Once more, a PLC HTML application will be used to send information out from PLC to suppliers and made available on a webpage as shown on Figure 4-16.



```

<!doctype>
<html lang=en>
    <head>
        <meta charset=utf-8>
        <title> Weekly Equipment Order for Experimental Beverage Plant </title>
    </head>
    <body>
        <h1> Weekly Equipment Order for Experimental Beverage Plant </h1>
        <h3> Please supply the following equipment for Beverage plant for this week:</h3>
        <ul>
            <em>
            <li> Amount of Bottles :="Equipment".bottles:</li>
            <li> Amount of Bottles Caps  :="Equipment".caps:</li>
            <li> Amount of labels :="Equipment".labels: </li>
            <li> Amount of Ingredients :="Equipment".ingredients: Kg </li>
            </em>
        </ul>
        <h3>Thanks for your Services!!</h3>

        <map name="primary">
            <area shape="circle" coords="200,250,25" href="another.htm" />
            <area shape="default" nohref />
        </map>

    </body>

</html>=
```

Figure 4-16: Equipment order webpage HTML code

The same principal (PLC HTML application) is used to design a webpage that displays available beverage stock to consumers when production has started.

4.4.2 Predictive maintenance rules

As explained in Chapter 3, the predictive maintenance rule of our research relies on a close monitoring of the conveyor motor health states. From figure 4-2, a vibration sensor is mounted on the motor from which vibration needs to be monitored and the other hand of the sensor is connected to the PLC which will convert analogue values to understandable vibration speed. In this research, the vibration sensor is wired as a current (4-20mA) analogue input to the PLC. The sensor is wired to the analogue input card attached to the PLC CPU. Figure 4-17 is an overview on the PLC hardware as displayed on the programming software tool with all the analogue cards.

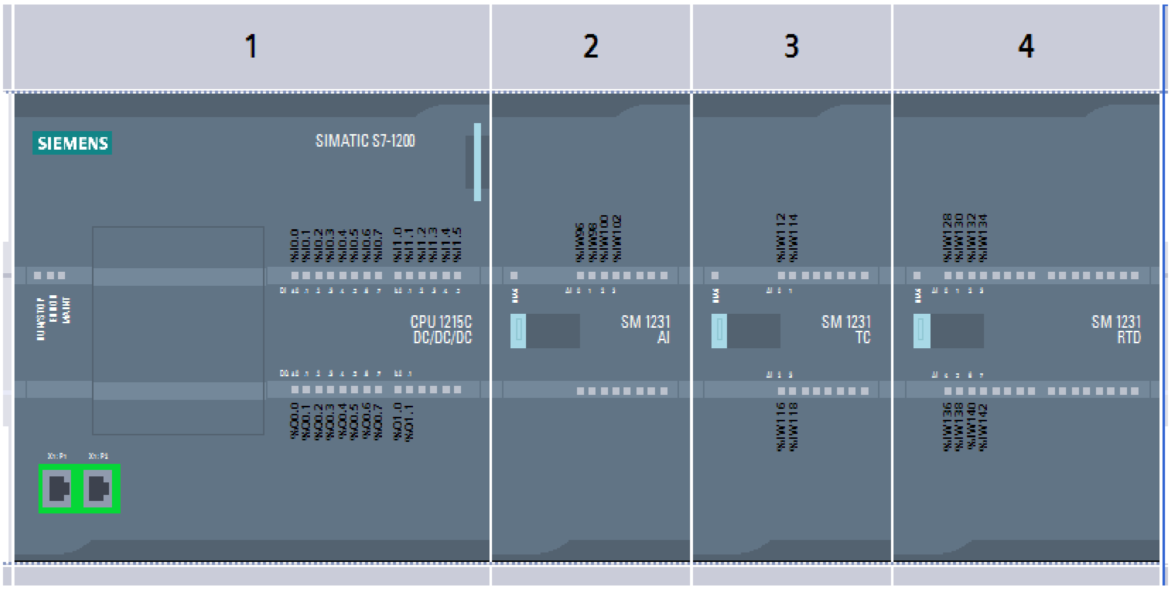
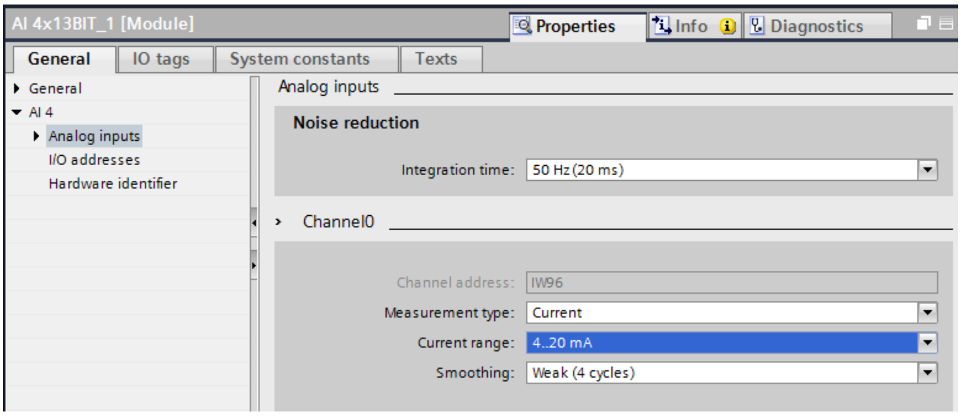


Figure 4-17: PLC hardware view in programming software with CPU and analogue input modules

A setting needs to be done on the PLC hardware configuration for the PLC to get reading from the analogue current input. This setting is displayed on Figure 4-18:



Analog inputs

Noise reduction

Integration time: 50 Hz (20 ms)

Channel0

Channel address: IW96

Measurement type: Current

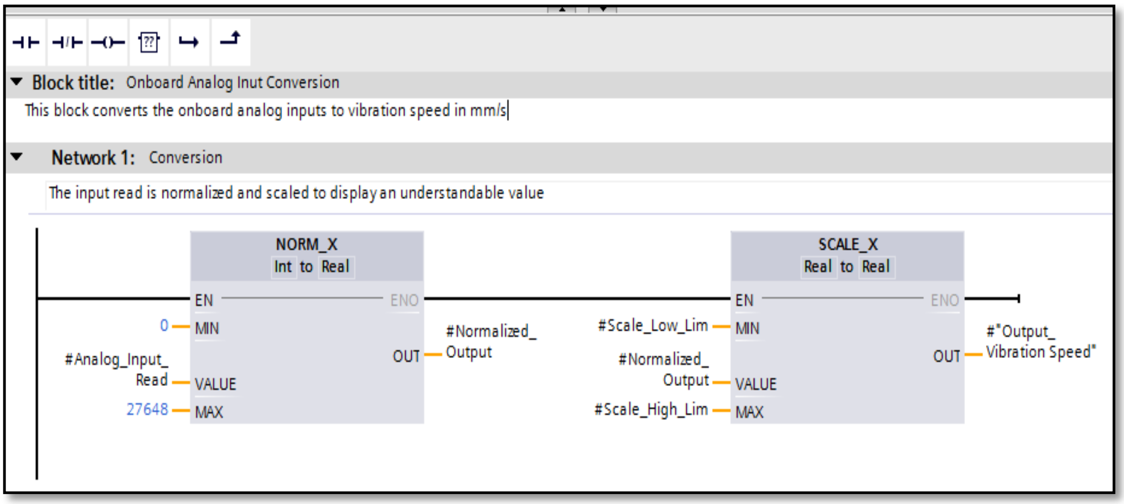
Current range: 4..20 mA

Smoothing: Weak (4 cycles)

Figure 4-18: Analogue input configuration in PLC hardware config

The current rating of the vibration sensor is an important detail to be used in the PLC hardware configuration. This information can be seen on the sensor’s datasheet.

In the PLC code, to convert the read analogue input value into understandable vibration speed values, we configure two Siemens blocks: NORM_X (normalizing) and SCALE_X (scaling) as displayed on Figure 4-19 that are used to convert raw analogue inputs to logical values.


Block title: Onboard Analog Inut Conversion

This block converts the onboard analog inputs to vibration speed in mm/s

Network 1: Conversion

The input read is normalized and scaled to display an understandable value

Figure 4-19: Analogue input conversion to a real value through NORM_X and SCALE_X

NORM_X block converts the Integer value of the analogue input to a real value and feed it into the SCALE_X block. SCALE_X converts the real value to a desired range of vibration speed for this application. As per Table 3-2, our low limit for the Vibration speed in mm/s will be 0.28 and higher limit will be 45. Putting these two values as parameter of the SCALE_X block narrow the range of the output vibration speed that is the actual value we will be constantly monitoring. The output of the SCALE_X block which is the vibration speed in mm/s is then used by other PLC blocks to program the actual predictive maintenance rules as grouped in cases 1, 2 and 3 in chapter 3. Figure 4-20 is an illustration done using case 1:

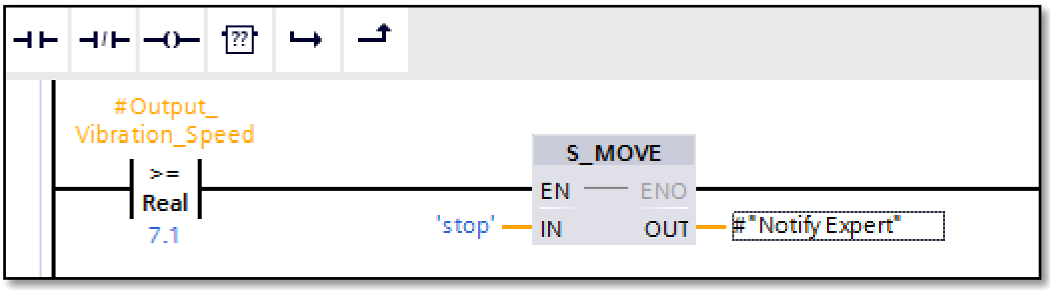


Figure 4-20: Vibration speed unacceptable range detection

The PLC code block on Figure 4-20 continuously monitors vibration speeds to check if the unacceptable vibration speed range (above 7.1 mm/s in this example) is detected. The second block of the rung is the action taken as soon as the unacceptable range of speed is detected. As previously mentioned, this should result in an immediate stop for reactive maintenance and a maintenance expert should be alerted accordingly. The S_MOV block writes the string ‘stop’ to a variable that will be recorded in a CSV file and later on read by a python script to send an email notification to the expert. The previous section on S7-1200 PLC internet services configuration already explained how to create and manipulate CSV files in the PLC through the web server. After a delay of 30 seconds the variable gets reset to avoid the script sending continuous emails until a new condition will be detected. Figure 4-21 shows how this variable is reset.

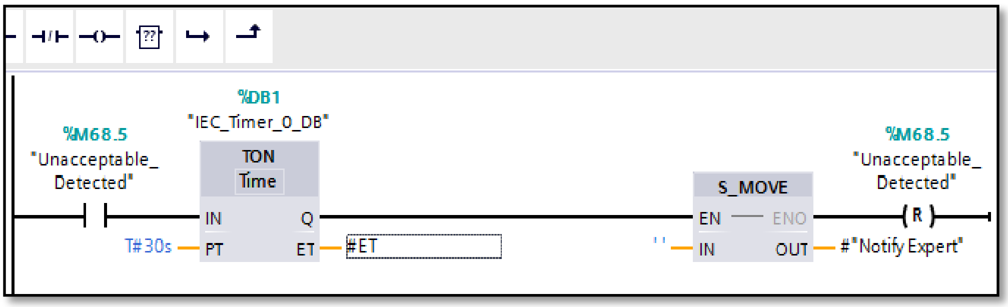
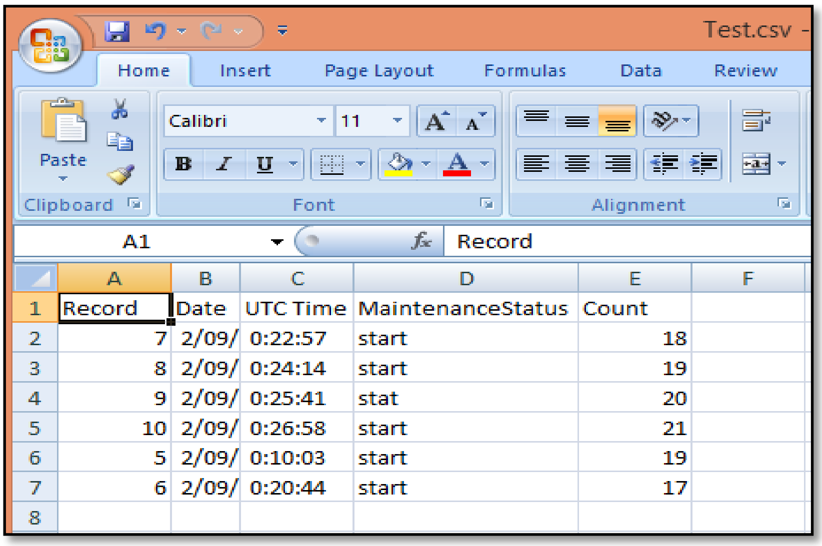


Figure 4-21: Reset expert notification variable

The same programming logic will apply for case 2 and 3 of the predictive maintenance rules.

4.4.2.1 Python script configuration for email notification to maintenance experts

As early mentioned, the PLC generates a csv file which contains notifications of predictive maintenance schedule for the conveyor motor. One of our research’s aims is to move away from a centralized system where any important information is only accessible via the central PLC or SCADA and directly notify maintenance experts in charge, real-time, anytime, when a schedule is generated without them having to physically move to the plant. This is achieved by combining a running python script continuously reading the notification in the PLC csv file and conveying through the internet the corresponding message (e-mail) to the expert. Figure 4-22 is an example of the PLC csv file format:



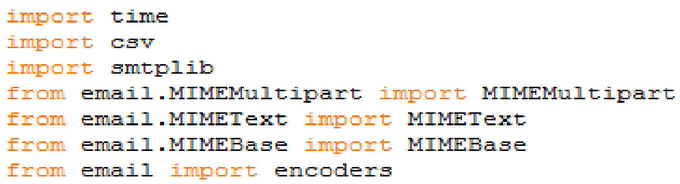
	A	B	C	D	E	F
1	Record	Date	UTC Time	MaintenanceStatus	Count	
2	7	2/09/	0:22:57	start	18	
3	8	2/09/	0:24:14	start	19	
4	9	2/09/	0:25:41	stat	20	
5	10	2/09/	0:26:58	start	21	
6	5	2/09/	0:10:03	start	19	
7	6	2/09/	0:20:44	start	17	
8						

Figure 4-22: PLC CSV file template for email notification via python script

The most important parameter in this file is the fourth column (maintenance status). It contains information that the PLC writes based on Figure 4-20 logic when a certain condition is met. The python script uses this parameter to trigger the sending of emails.

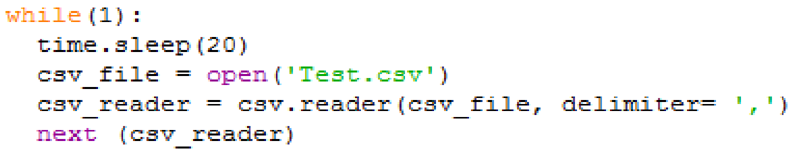
Here is the skeleton of the python algorithm used to send notifications to experts:

```
Import all necessaries libraries and modules

While (1) // Continuous script run
   Delay (20 seconds) // Small processing time to synchronize with PLC
     Open CSV file // this file has the same name and location as saved on the PLC
 If fourth row = ‘stop’
→ Reactive maintenance – Immediate stop
→ Send specific message for reactive maintenance to supervisor // Email address and settings to be inserted
       Elif (second if) fourth row = ‘start’
→ Predictive maintenance – On next machine stop
→ Send specific message for predictive maintenance to supervisor // Email address and settings to be inserted
         Else: print on compiler “normal operation”
     Close CSV file
```



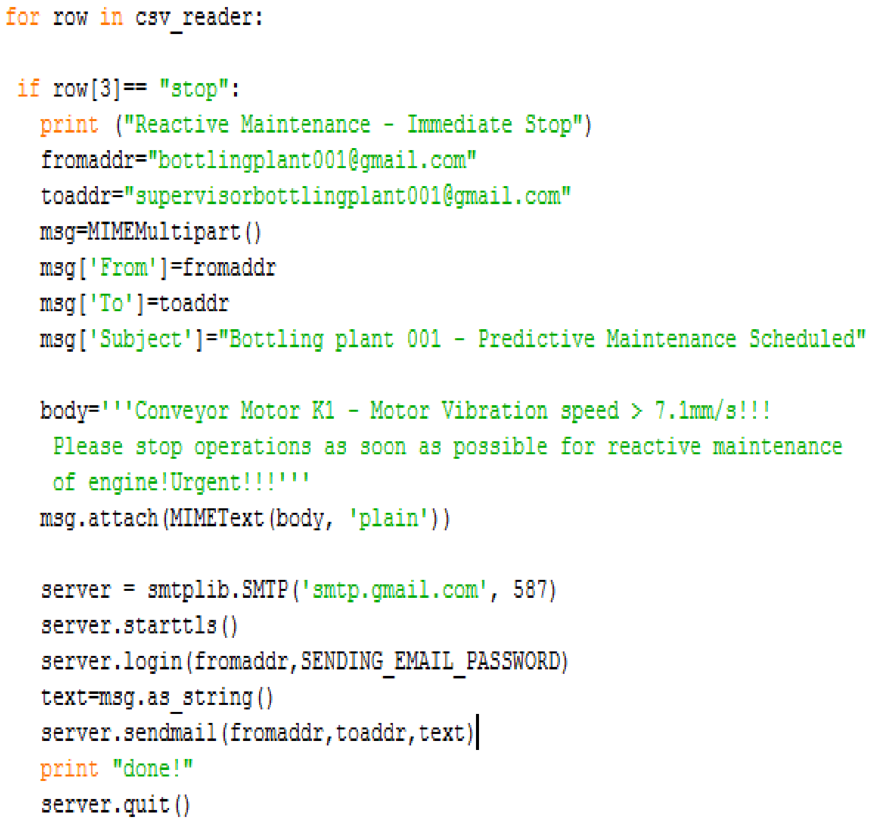
```
import time
import csv
import smtplib
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders
```

Code 4-1: Import Python libraries



```
while(1):
  time.sleep(20)
  csv_file = open('Test.csv')
  csv_reader = csv.reader(csv_file, delimiter= ',')
  next (csv_reader)
```

Code 4-2: Open CSV file



```
for row in csv_reader:

 if row[3]== "stop":
  print ("Reactive Maintenance - Immediate Stop")
  fromaddr="bottlingplant001@gmail.com"
  toaddr="supervisorbottlingplant001@gmail.com"
  msg=MIMEMultipart()
  msg['From']=fromaddr
  msg['To']=toaddr
  msg['Subject']="Bottling plant 001 - Predictive Maintenance Scheduled"

  body='''Conveyor Motor K1 - Motor Vibration speed > 7.1mm/s!!!
   Please stop operations as soon as possible for reactive maintenance
   of engine!Urgent!!!'''
  msg.attach(MIMEText(body, 'plain'))

  server = smtplib.SMTP('smtp.gmail.com', 587)
  server.starttls()
  server.login(fromaddr,SENDING_EMAIL_PASSWORD)
  text=msg.as_string()
  server.sendmail(fromaddr,toaddr,text)
  print "done!"
  server.quit()
```

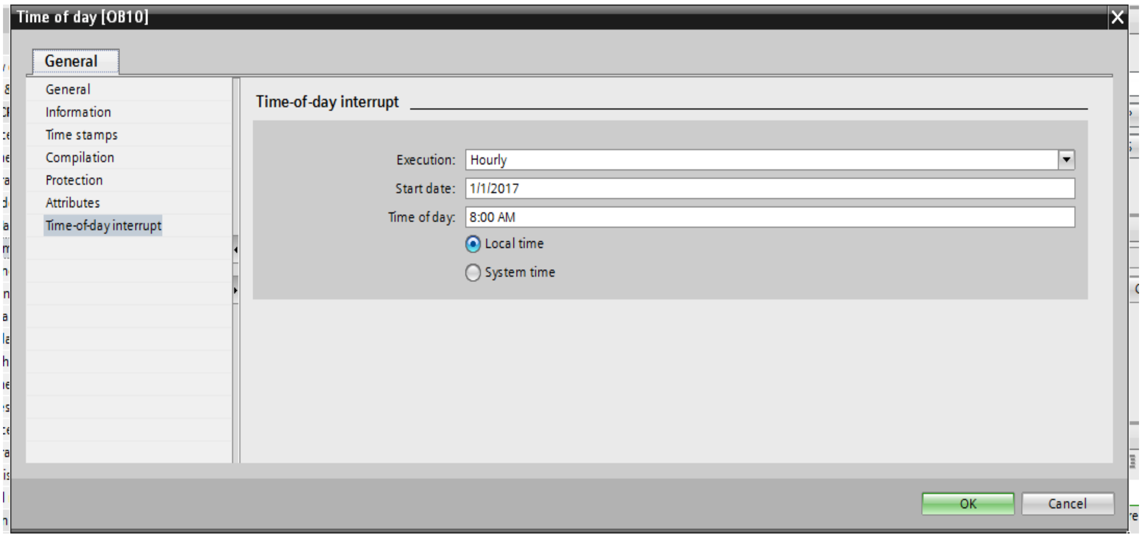
Code 4-3: Send email to maintenance expert

4.4.3 Hourly production tracking

In the S7-1200 PLC program, we also create a function block in which all the operations will be made for the production tracking of the bottling process which refers to getting an hourly report of the bottling count, at the end of the shift compare the result to the daily predicted or entered target (as per our flow chart operation) and compute automatically an action based on the result. This is called the parameters auto-configuration. Some internal functions of the S7-1200 PLC like the time of day interrupts will be used in our solution.

In programming, the interrupts refer to a notification, communicated to the controller, by a hardware device or software, on receipt of which controller momentarily stops and responds to the interrupt. Whenever an interrupt occurs the controller completes the execution of the current instruction and starts the execution of an Interrupt Service Routine (ISR) or Interrupt Handler (Choudhary H, 2012). In this research, the interrupt that will be programmed is based on time configuration.

We start by generating an Organization Block, known as an "OB" of the category Time of Day. We configure the block to be executed hourly from $\mathbf{t_0 = 08{:}00}$ which is the beginning of the first shift. In other words, at each hour $\mathbf{t_0+t}$**,** $t \in \mathbb{N}$ from the beginning of the shift we will be tracking our production. Figure 4-23 displays a Time of day OB.

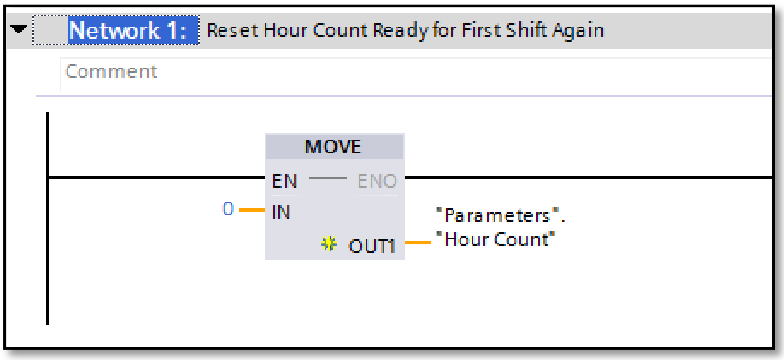


Time-of-day interrupt

Execution: Hourly
Start date: 1/1/2017
Time of day: 8:00 AM
● Local time
○ System time

Figure 4-23: Time of day interrupt PLC configuration

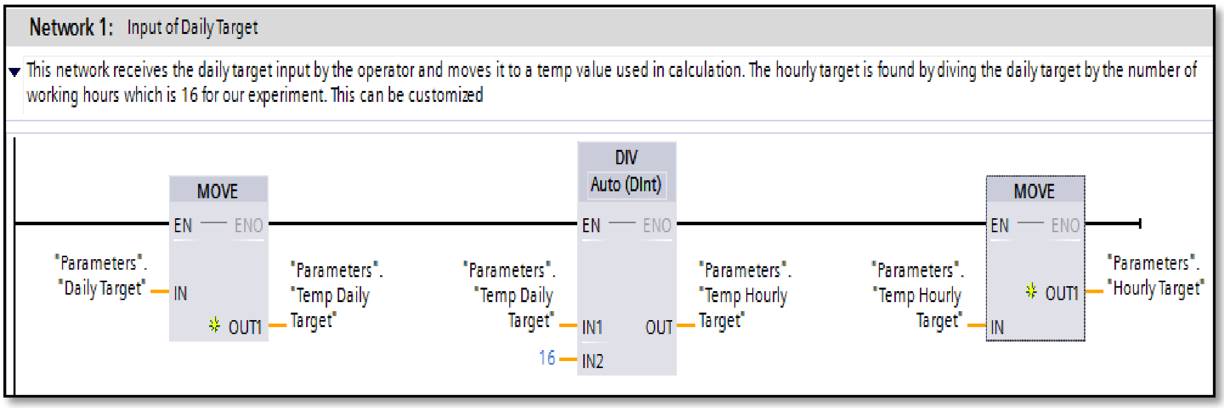
We will also run two other time interrupts to notify the end of each shift and generate the end of shift's report on the SCADA. At the end of the last shift, the hour count is reset to zero and ready to start-up the first shift again. Figure 4-24 displays the reset function.



Network 1: Reset Hour Count Ready for First Shift Again

Figure 4-24: Reset hour count at the end of last shift

In chapter 3, we generated a mathematical expression (3.11) to find the hourly target to be achieved by the system. The formula (3.11) depends on the daily target **Q** and on the number of normal working hour's **m** which should be input by the plant technicians or supervisors. Figure 4-25 is the equivalent of formula (3.11) in PLC ladder language. In the PLC code below, the number of working hours is equal to 16 hours corresponding to two shifts. The user will only input the daily target on the SCADA, if not automatically predicted yet by the system, and in a real time will get the feedback of the hourly target.

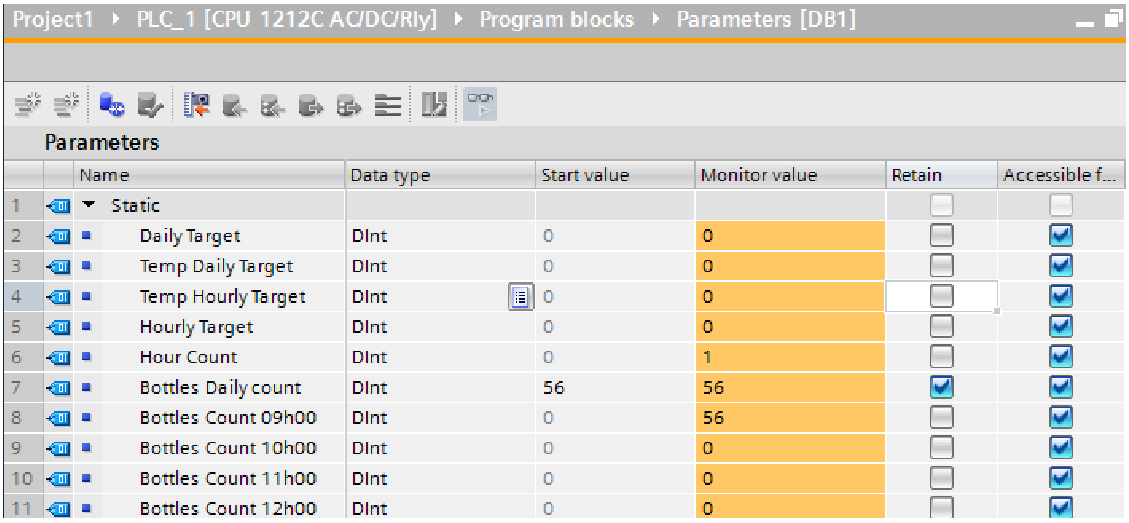


Network 1: Input of Daily Target

This network receives the daily target input by the operator and moves it to a temp value used in calculation. The hourly target is found by diving the daily target by the number of working hours which is 16 for our experiment. This can be customized

Figure 4-25: From daily bottles target input to bottles Hourly target

Note: Not to lose any value in case of power loss or Hardware configuration download, all parameters and values are saved in a Data Bock. Data Block has got the ability to retain values (when configured to do so) unlike the normal PLC memory bits. Data block variables are displayed on Figure 4-26.



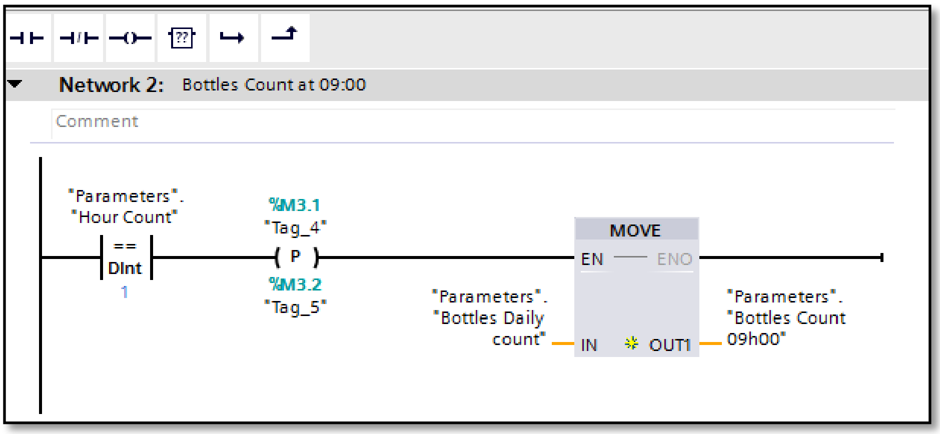
Project1 ▸ PLC_1 [CPU 1212C AC/DC/Rly] ▸ Program blocks ▸ Parameters [DB1]

Parameters

	Name	Data type	Start value	Monitor value	Retain	Accessible f...
1	Static				☐	☐
2	Daily Target	DInt	0	0	☐	☑
3	Temp Daily Target	DInt	0	0	☐	☑
4	Temp Hourly Target	DInt	0	0	☐	☑
5	Hourly Target	DInt	0	0	☐	☑
6	Hour Count	DInt	0	1	☐	☑
7	Bottles Daily count	DInt	56	56	☑	☑
8	Bottles Count 09h00	DInt	0	56	☐	☑
9	Bottles Count 10h00	DInt	0	0	☐	☑
10	Bottles Count 11h00	DInt	0	0	☐	☑
11	Bottles Count 12h00	DInt	0	0	☐	☑

Figure 4-26: Data Block Variables

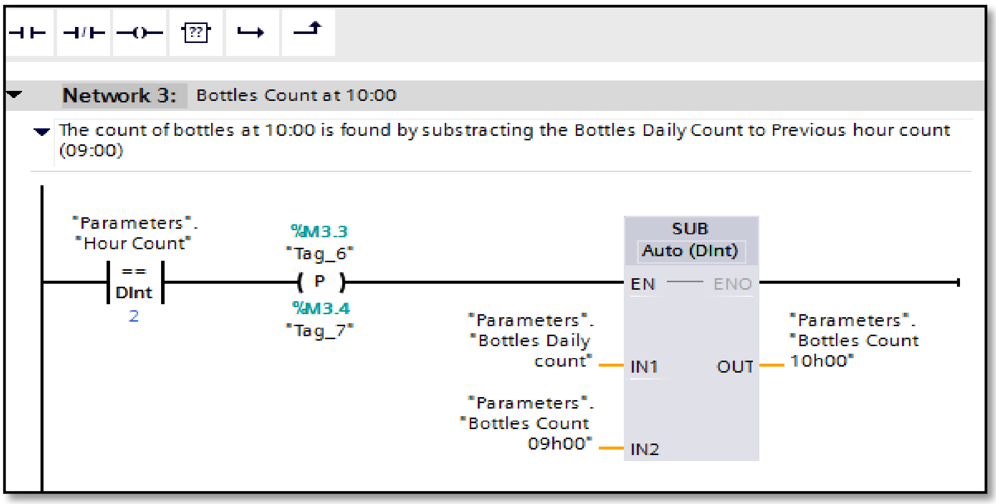
In this research, we are not much interested in the physical hardware methods used in the plant to get the final bottles count but rather in tracking that final count and computing an understandable production tracking report in order to take actions. One of the most common hardware methods used in bottling plants to count bottles is the reading of a digital proximity input sensor located at the end of the capping stage where all bottles going through are well filled and ready to be packed. The sensor counts every bottle passing through and transmits it to the PLC program. Figure 4-27 and Figure 4-28 are two samples expressions of the hourly bottle count in PLC programming code. They are equivalent to the mathematical model summarized for the hourly bottles count in chapter 3 as Equation (3.12):



Network 2: Bottles Count at 09:00

Comment

Figure 4-27: Network rung for bottles count at 09:00

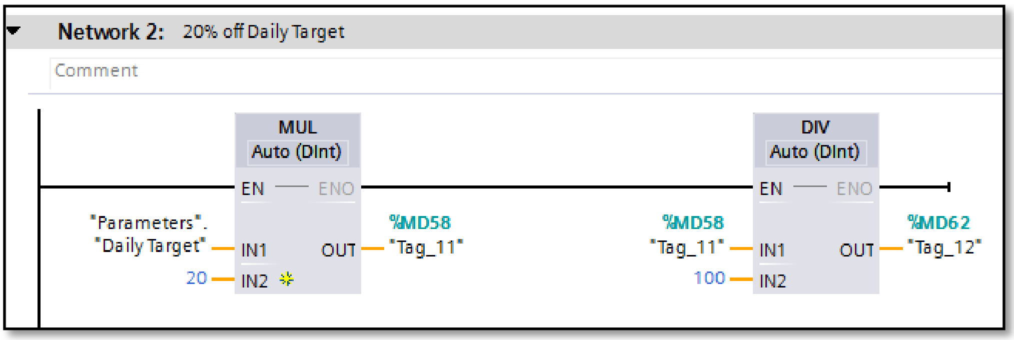


Network 3: Bottles Count at 10:00

The count of bottles at 10:00 is found by substracting the Bottles Daily Count to Previous hour count (09:00)

Figure 4-28: Network rung for bottles count at 10:00

One of the actions that could be taken at the end of a shift based on end of shift report is the initiation of an overtime production run. This is performed if the production is below a certain percentage set by plant technicians and supervisors. The overtime calculation expression was defined by formula (3.14). The actual condition for overtime run is represented by (3.15). Expression (3.15) can be interpreted as follows: if the current count of bottles at the final working hour is less or equal to the overtime percentage condition generate auto-parameters for full speed run and start overtime run. This operation is done automatically at the end of the shift, no additional parameters except the daily target input at the beginning of the shift is required. All calculations are performed in the PLC code. In this research the overtime percentage OP is pre-set to 20% and programmed in the PLC as displayed on Figure 4-29.

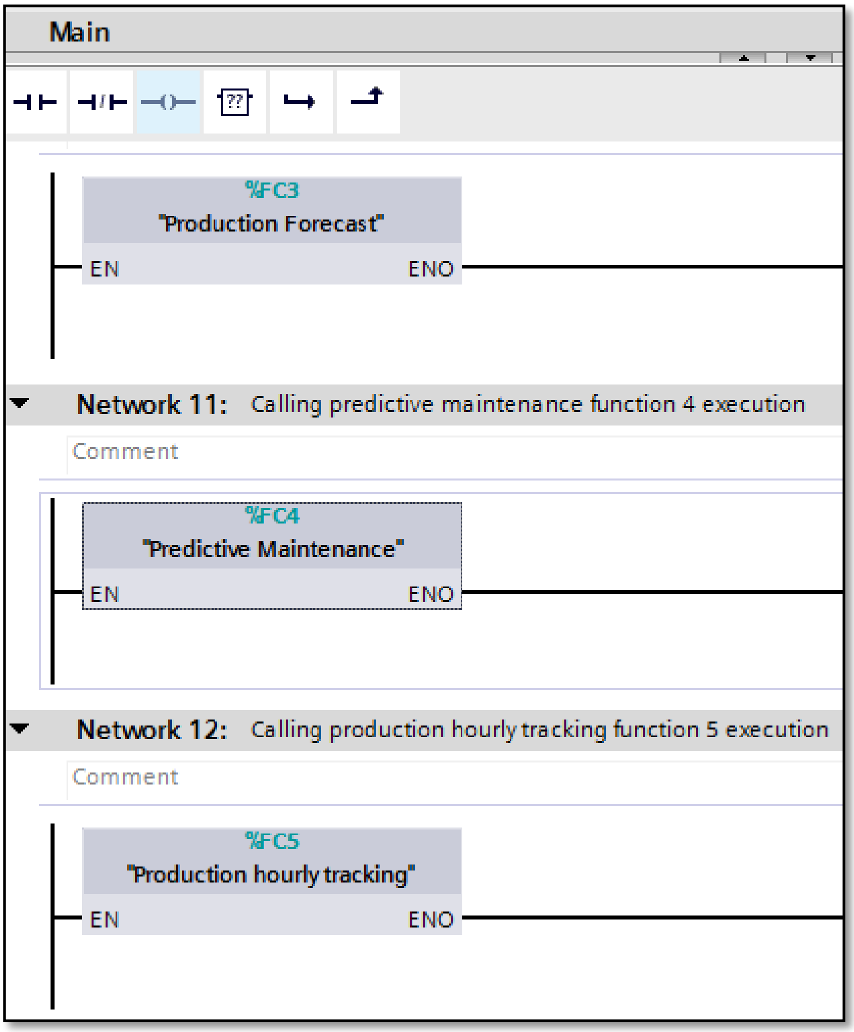


Network 2: 20% off Daily Target

Comment

Figure 4-29: Pre-set Percentage Calculations for actions on Machine Bottling Run

Note: Except the time of day OBs, all created functions need to be called from the OB1 in order to be executed by the PLC. The three main functions of our bottling process: production forecast, predictive maintenance and hourly production tracking have been programmed in three different functions with sub functions in each of them. Only the main functions are being called in OB1 as shown on Figure 4-30.



Network 11: Calling predictive maintenance function 4 execution

Network 12: Calling production hourly tracking function 5 execution

Figure 4-30: Main functions call in OB1

4.5 Bottling production data available on cloud

As mentioned in previous chapters, another important aim of this research is to move away from a centralized monitoring system and add a decentralized data access platform. This added feature is mainly profitable for management levels which do not always have permanent access to physical running plants but have to closely track production when needed. In the olden days and currently in some SMEs, management relies on information provided by their supervisors. This information is time depend, because they only get them at a specific time of the day or even days of the week, and sometimes inaccurate, because the information comes from a human being. In this research, we develop a live cloud-based dashboard report that management can access anytime when connected to the internet, using their smartphones, tablets or personal computers. The online dashboard report receives automatic updates on the bottling plant from a local MySQL server which is the panel PC on which the SCADA software is running. As displayed on the overall system's architecture on Figure 4-1, the SCADA panel PC is connected to the PLC and the PLC directly to the bottling plant. Figure 4-31 displays how the bottling plant information is made available on the cloud.

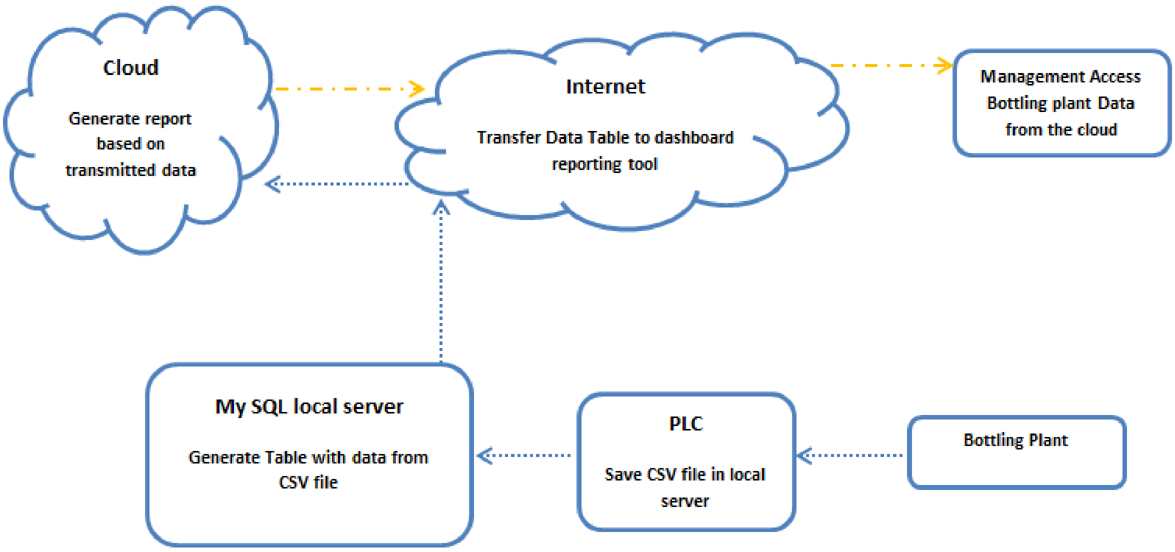
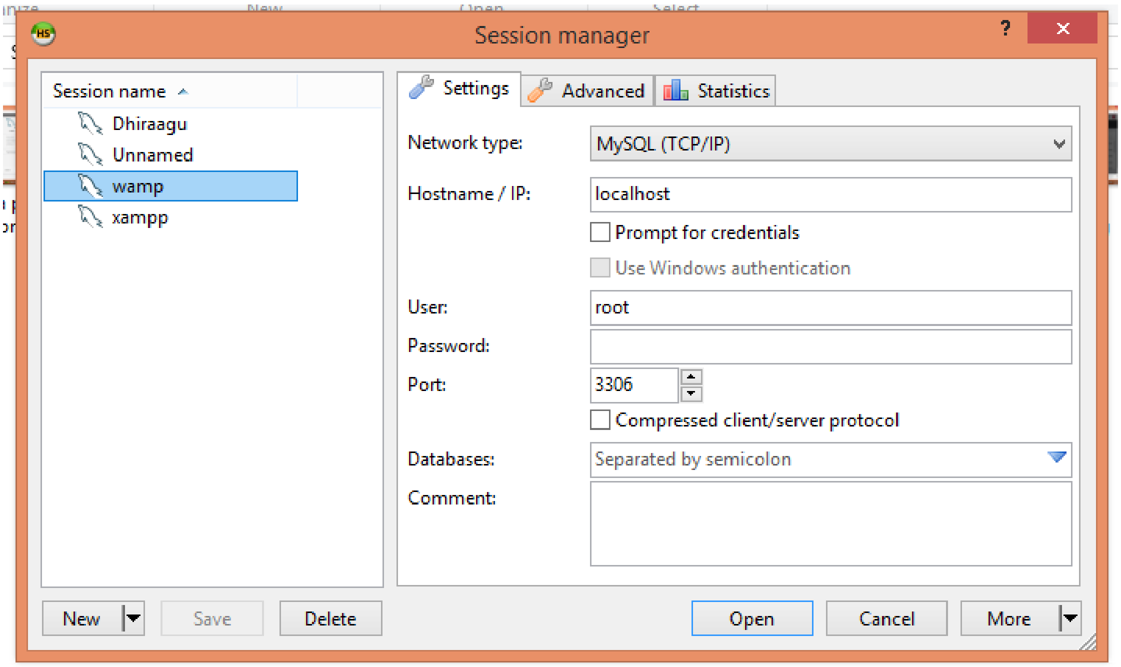


Figure 4-31: Bottling plant Data Access through Cloud

Only some key information, useful for management, is displayed on the dashboard:

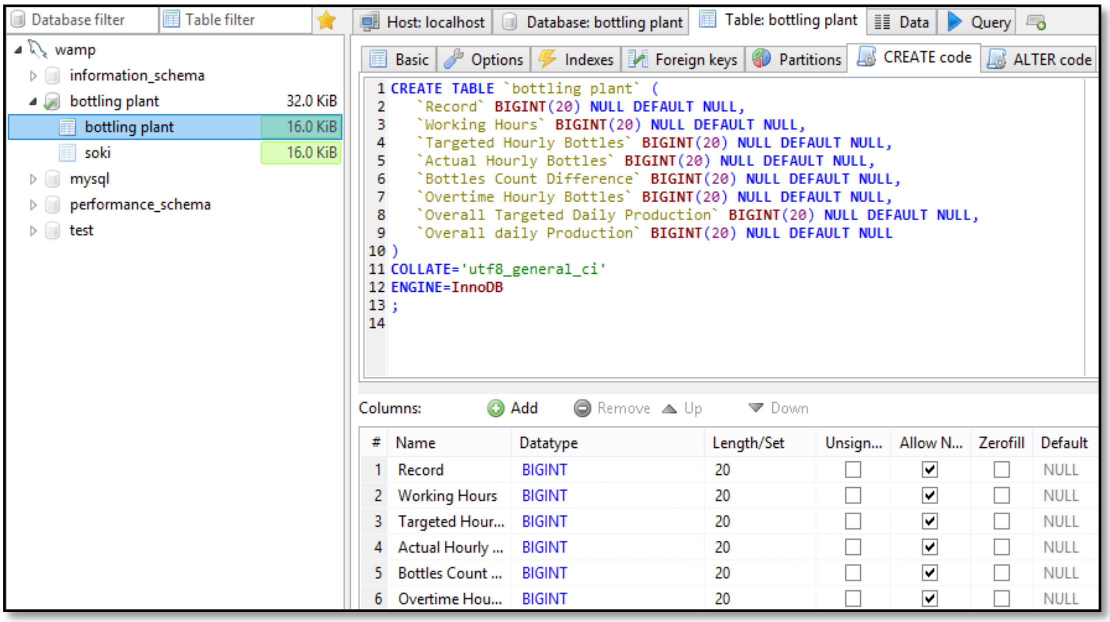
- The actual hourly bottling production as implemented by Section 4.4.3
- The targeted hourly bottling production versus the actual bottling production
- The overtime hourly bottling production (if any) versus the two previous counts

The above listed data are saved from the running PLC to the local MySQL server in a csv file. A database is then created in the local server to group all those data in a single Table. The creation of the table is performed via a query in the database. The table containing data is later directly linked to the dashboard reporting tool. Figure 4-33 displays the created database named “bottling plant” and the code written to generate a database table. After running the code, the actual table is generated as shown on Figure 4-34. Some important settings of the local server: hostname, Port number, will have to be used in the online reporting tool for the two entities to be connected. Figure 4-32 displays these two settings as used on the local server.



Network type: MySQL (TCP/IP)
Hostname / IP: localhost
☐ Prompt for credentials
☐ Use Windows authentication
User: root
Password:
Port: 3306
☐ Compressed client/server protocol
Databases: Separated by semicolon
Comment:

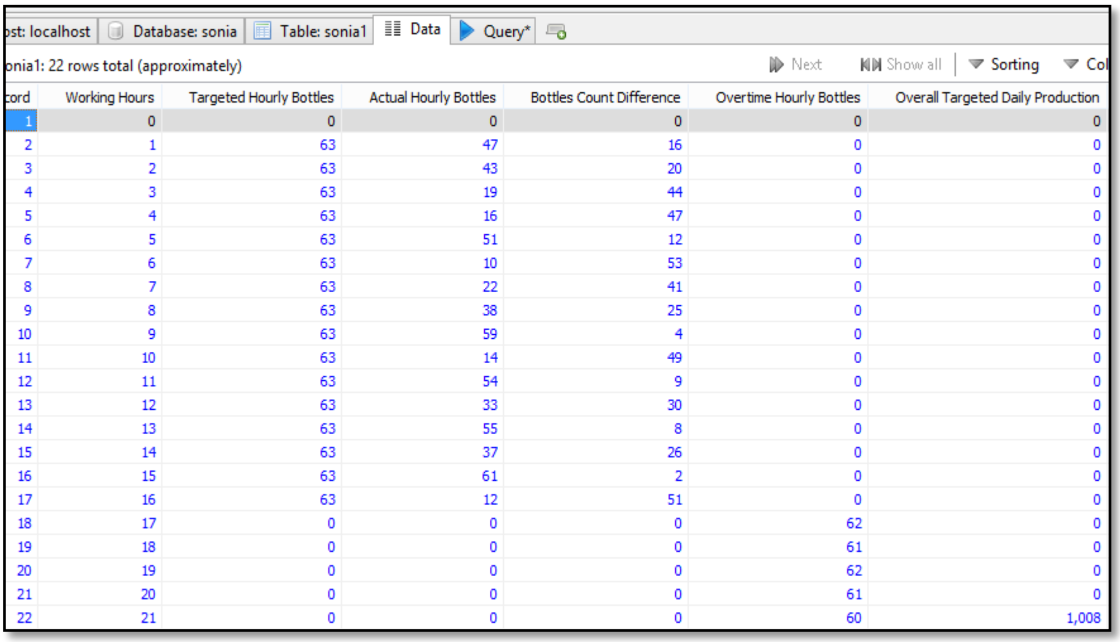
Figure 4-32: Local server parameter settings



```sql
CREATE TABLE `bottling plant` (
	`Record` BIGINT(20) NULL DEFAULT NULL,
	`Working Hours` BIGINT(20) NULL DEFAULT NULL,
	`Targeted Hourly Bottles` BIGINT(20) NULL DEFAULT NULL,
	`Actual Hourly Bottles` BIGINT(20) NULL DEFAULT NULL,
	`Bottles Count Difference` BIGINT(20) NULL DEFAULT NULL,
	`Overtime Hourly Bottles` BIGINT(20) NULL DEFAULT NULL,
	`Overall Targeted Daily Production` BIGINT(20) NULL DEFAULT NULL,
	`Overall daily Production` BIGINT(20) NULL DEFAULT NULL
)
COLLATE='utf8_general_ci'
ENGINE=InnoDB
;
```

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default
1	Record	BIGINT	20	☐	☑	☐	NULL
2	Working Hours	BIGINT	20	☐	☑	☐	NULL
3	Targeted Hour...	BIGINT	20	☐	☑	☐	NULL
4	Actual Hourly ...	BIGINT	20	☐	☑	☐	NULL
5	Bottles Count ...	BIGINT	20	☐	☑	☐	NULL
6	Overtime Hou...	BIGINT	20	☐	☑	☐	NULL

Figure 4-33: Database code to create bottling data table in local server

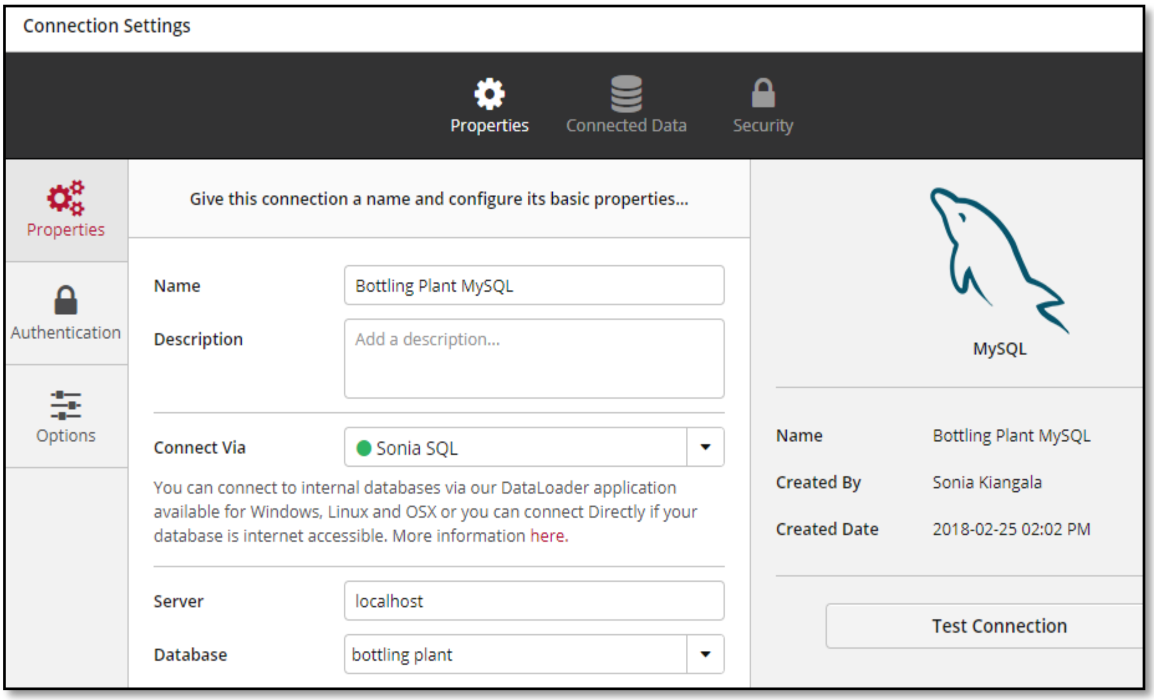


onia1: 22 rows total (approximately)

ord	Working Hours	Targeted Hourly Bottles	Actual Hourly Bottles	Bottles Count Difference	Overtime Hourly Bottles	Overall Targeted Daily Production
1	0	0	0	0	0	0
2	1	63	47	16	0	0
3	2	63	43	20	0	0
4	3	63	19	44	0	0
5	4	63	16	47	0	0
6	5	63	51	12	0	0
7	6	63	10	53	0	0
8	7	63	22	41	0	0
9	8	63	38	25	0	0
10	9	63	59	4	0	0
11	10	63	14	49	0	0
12	11	63	54	9	0	0
13	12	63	33	30	0	0
14	13	63	55	8	0	0
15	14	63	37	26	0	0
16	15	63	61	2	0	0
17	16	63	12	51	0	0
18	17	0	0	0	62	0
19	18	0	0	0	61	0
20	19	0	0	0	62	0
21	20	0	0	0	61	0
22	21	0	0	0	60	1,008

Figure 4-34: Generated database table

The dashboard reporting tool is called Clic Data. It is setup using the same parameters on Figure 4-32 for connection to take place. Figure 4-35 and Figure 4-36 shows how the two parameters are setup in Clic Data.



Give this connection a name and configure its basic properties...

Name: Bottling Plant MySQL

Description: Add a description...

Connect Via: Sonia SQL

You can connect to internal databases via our DataLoader application available for Windows, Linux and OSX or you can connect Directly if your database is internet accessible. More information here.

Server: localhost

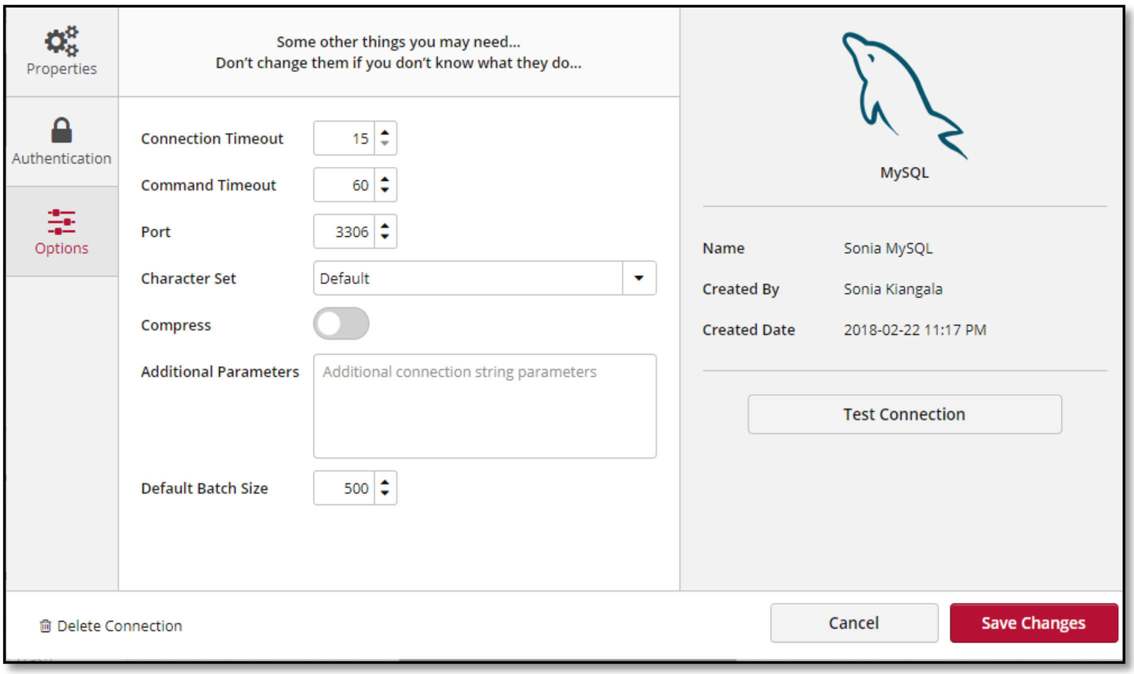
Database: bottling plant

Name: Bottling Plant MySQL

Created By: Sonia Kiangala

Created Date: 2018-02-25 02:02 PM

Figure 4-35: Linking bottling plant local server database to Clic Data database settings



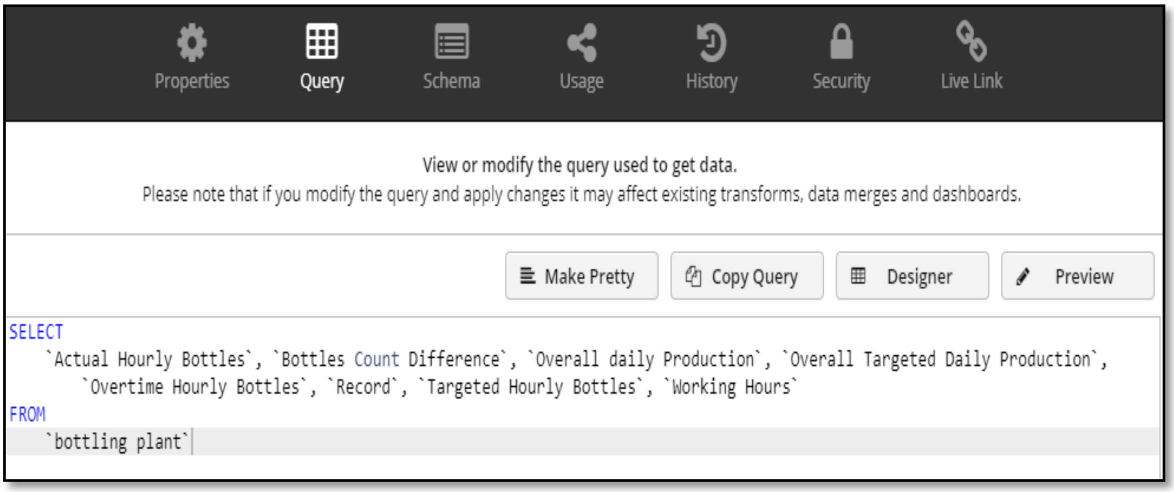
Some other things you may need...
Don't change them if you don't know what they do...

Connection Timeout 15
Command Timeout 60
Port 3306
Character Set Default
Compress
Additional Parameters Additional connection string parameters
Default Batch Size 500

Name Sonia MySQL
Created By Sonia Kiangala
Created Date 2018-02-22 11:17 PM

Figure 4-36: Port number connection setting on Clic Data

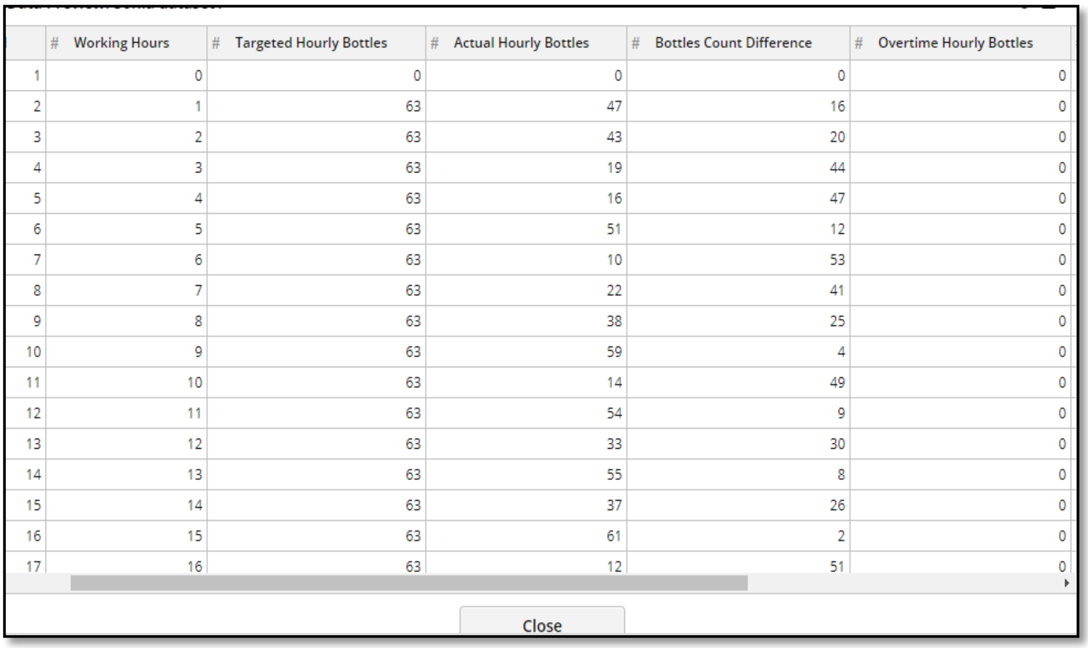
A similar query as the one on Figure 4-33 will be implemented in Clic Data to generate a data table for the creation of the actual dashboard report. Figure 4-37 displays the query in Clic Data and the created table is shown on Figure 4-38.



View or modify the query used to get data.
Please note that if you modify the query and apply changes it may affect existing transforms, data merges and dashboards.

```
SELECT
    `Actual Hourly Bottles`, `Bottles Count Difference`, `Overall daily Production`, `Overall Targeted Daily Production`,
        `Overtime Hourly Bottles`, `Record`, `Targeted Hourly Bottles`, `Working Hours`
FROM
    `bottling plant`
```

Figure 4-37: Clic Data query to generate data table



	# Working Hours	# Targeted Hourly Bottles	# Actual Hourly Bottles	# Bottles Count Difference	# Overtime Hourly Bottles
1	0	0	0	0	0
2	1	63	47	16	0
3	2	63	43	20	0
4	3	63	19	44	0
5	4	63	16	47	0
6	5	63	51	12	0
7	6	63	10	53	0
8	7	63	22	41	0
9	8	63	38	25	0
10	9	63	59	4	0
11	10	63	14	49	0
12	11	63	54	9	0
13	12	63	33	30	0
14	13	63	55	8	0
15	14	63	37	26	0
16	15	63	61	2	0
17	16	63	12	51	0

Figure 4-38: Data table created in Clic Data

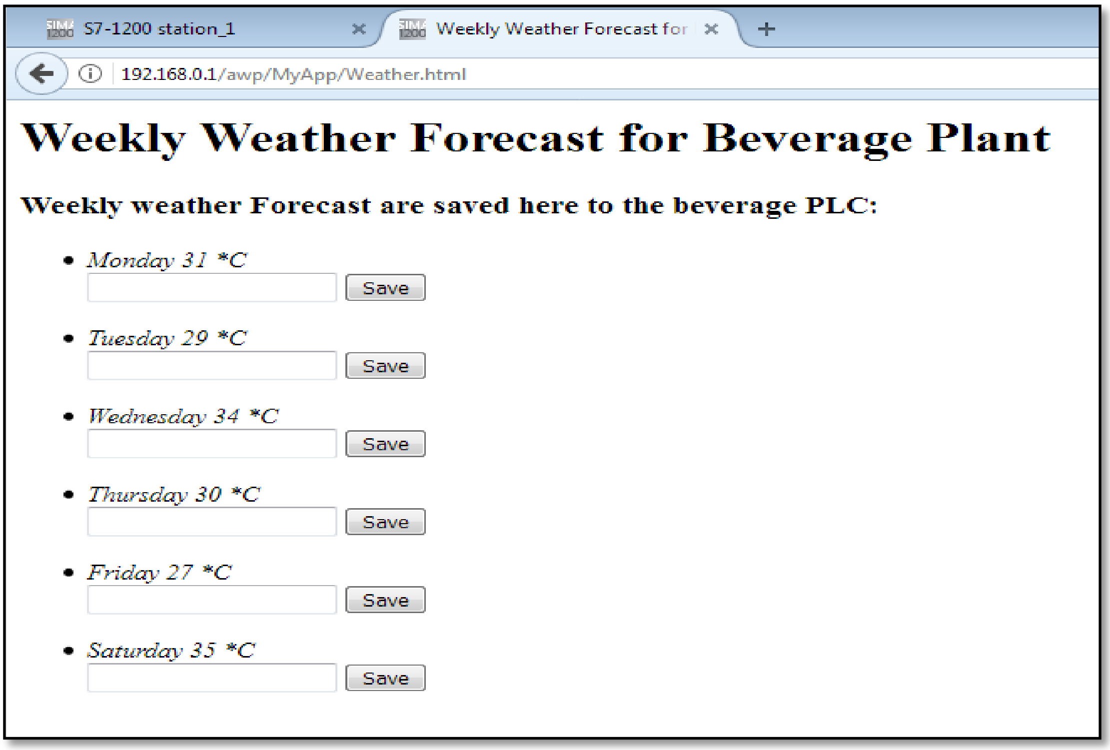
The reporting dashboard will be created and made available online for management based on the above data. Because the local server and the Clic Data database are linked, any change on any values present in the local server database will be automatically reflected on the dashboard.

The aim of this chapter was to practically implement the design of the high-tech automated bottling process by transforming all mathematical expressions generated in the theoretical modeling section into programming codes and structures within the PLC, python scripts, MySQL databases etc. A detailed overall architecture of the bottling process was created. The architecture clearly showed the relationship between different components of the system to paint a global idea of activities of the system. The flow chart of the production process was also designed with all stages for successful operations as well as provision for exceptions that could occur within the process.

Chapter 5 : EXPERIMENTAL RESULTS AND ANALYSIS

5.1 Bottling production forecast

Figure 4–16 in the previous chapter showed a HTML webpage code we wrote to receive weekly weather forecast from weather services to PLC webpage function and directly transmitted to the PLC code. Figure 5-1 displays the webpage created when running the PLC web server application.



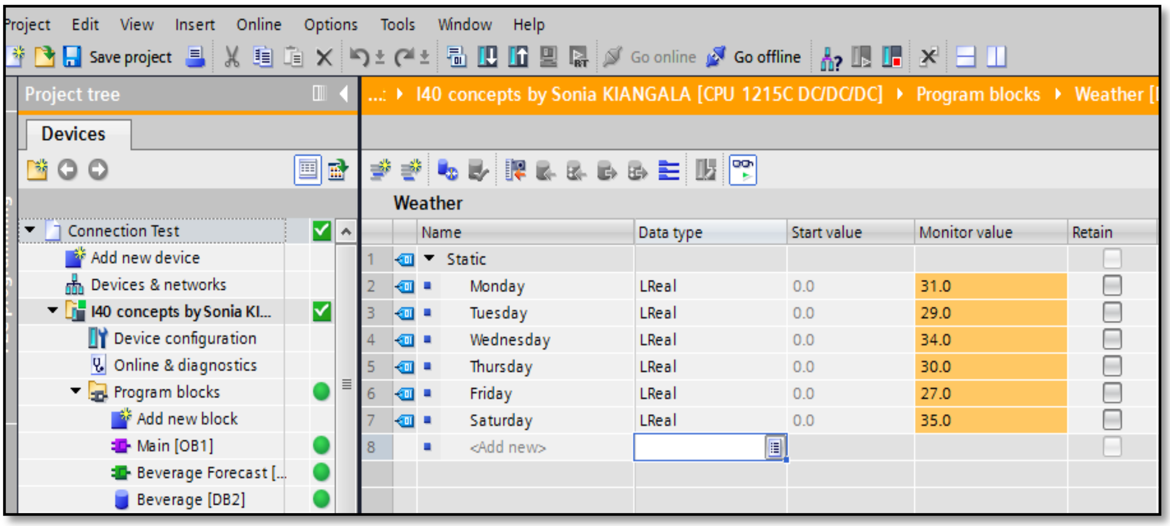
Weekly Weather Forecast for Beverage Plant

Weekly weather Forecast are saved here to the beverage PLC:

- *Monday 31 *C*
- *Tuesday 29 *C*
- *Wednesday 34 *C*
- *Thursday 30 *C*
- *Friday 27 *C*
- *Saturday 35 *C*

Figure 5-1: Result of weekly weather forecast written to PLC webpage

This webpage is accessible as one of the PLC web server functions via the PLC IP address: 192.168.0.1 through an internet browser. The weather forecast on Figure 5-1 for each production day of the week is then transferred to the PLC code database variables created on Figure 4-9. The live view on these variables as monitored on the PLC is displayed on Figure 5-2.

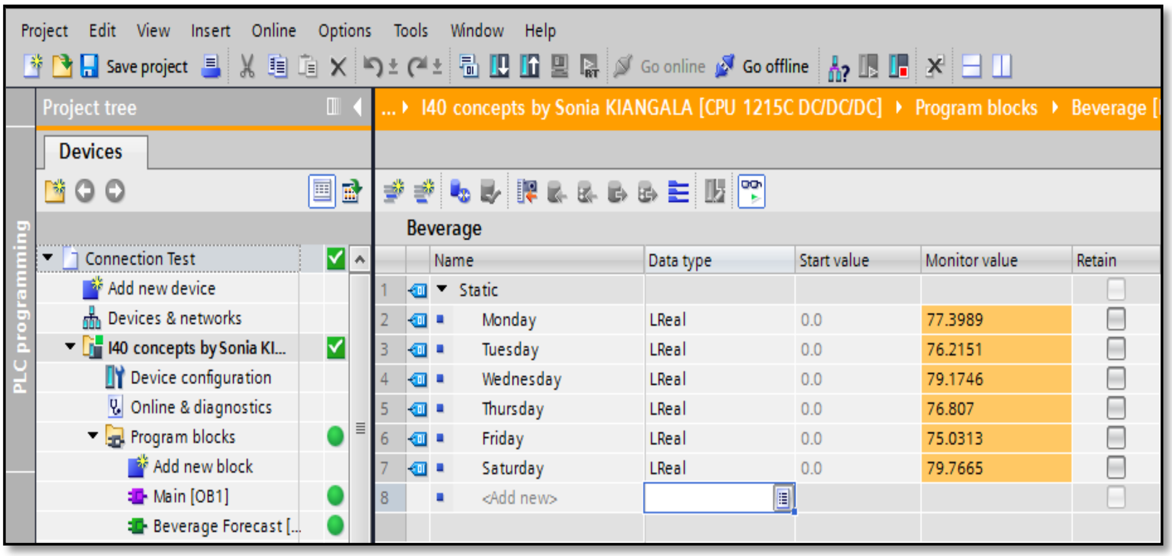


Weather

	Name	Data type	Start value	Monitor value	Retain
1	▼ Static				☐
2	Monday	LReal	0.0	31.0	☐
3	Tuesday	LReal	0.0	29.0	☐
4	Wednesday	LReal	0.0	34.0	☐
5	Thursday	LReal	0.0	30.0	☐
6	Friday	LReal	0.0	27.0	☐
7	Saturday	LReal	0.0	35.0	☐
8	<Add new>				☐

Figure 5-2: Weather forecast results transmitted to PLC database

Figure 5-2 is a live monitoring of weekly weather forecast as transmitted from the created webpage in Figure 5-1 to the programming PLC software. As we would notice, the same temperature forecasted are monitored in the running PLC. The regression equation (3.1) programmed in the PLC receives Figure 5-2 weather forecast and generates as a result the beverage production forecast for each respective weekdays into the created beverage database in Figure 4-14. Its results are displayed on Figure 5-3:

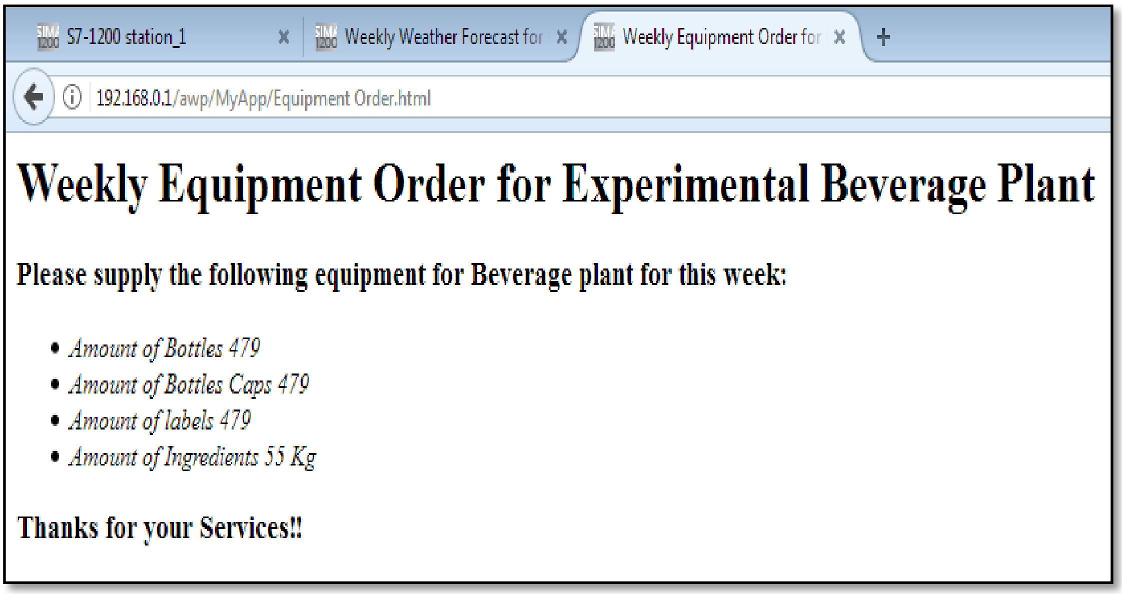


Beverage

	Name	Data type	Start value	Monitor value	Retain
1	▼ Static				☐
2	Monday	LReal	0.0	77.3989	☐
3	Tuesday	LReal	0.0	76.2151	☐
4	Wednesday	LReal	0.0	79.1746	☐
5	Thursday	LReal	0.0	76.807	☐
6	Friday	LReal	0.0	75.0313	☐
7	Saturday	LReal	0.0	79.7665	☐
8	<Add new>				☐

Figure 5-3: Production forecast results in PLC database from equation (3.1)

The result in Figure 5-3 is the aim of weekly production forecast as we are now able to predict what the production will be like for a specific week and we can foresee getting on time all the required stakeholders and avoid risk of running short while demand is high. Some of the production stakeholders are: bottles, bottles caps, labels and Ingredients. As shown on Figure 4-16, another HTML webpage was created for the beverage plant to acquire equipment and other stakeholders for production. As for weather services forecast, dedicated suppliers access the webpage through the internet using the PLC IP address and see how many bottles or ingredient should be prepared for the weekly production. In the background of the webpage in Figure 5-4, equations (3.4) and (3.5) are being run in the PLC code to send out requested quantities to suppliers:



Weekly Equipment Order for Experimental Beverage Plant

Please supply the following equipment for Beverage plant for this week:

- *Amount of Bottles 479*
- *Amount of Bottles Caps 479*
- *Amount of labels 479*
- *Amount of Ingredients 55 Kg*

Thanks for your Services!!

Figure 5-4: Equipment order webpage result from PLC HTML code

After running our production method for the same amount of days as during the design stage, 21 days, we noticed that the rate of production had increased; mainly due to reduction of errors caused by equipment shortage while demand is high. Production forecast allowed the beverage plant to have in advance a fairly approximate idea of demand quantity to expect and therefore get on time the necessary amount of stake holders. Figure 3-2 displays bottling production with errors due to short of stock. Figure 5-5 is the new production graph after running production forecast. This production graph has a reduced error rate displayed on the graph by concentration of dots around the trend line and less scattered ones as on Figure 3-2.

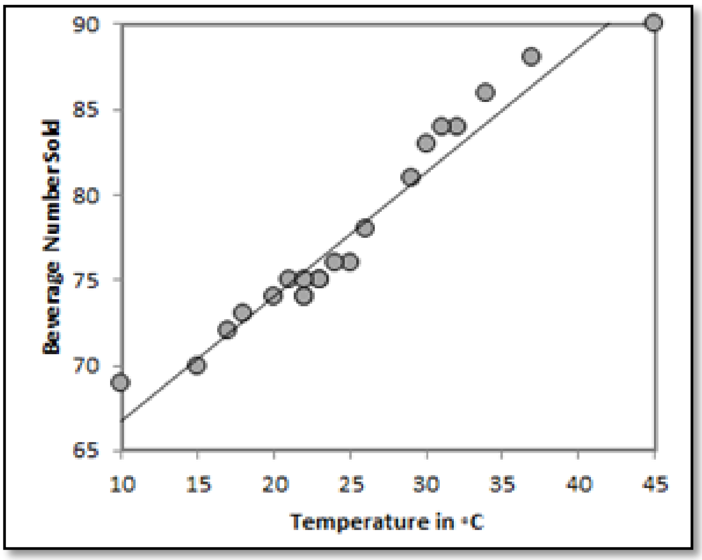


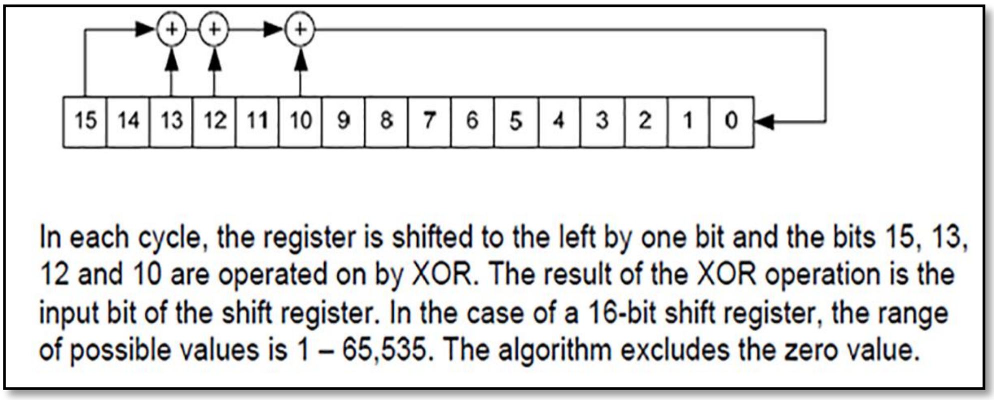
Figure 5-5: Beverage plant production after implementing optimization method

Figure 5-5 interprets the new weekly production output of the beverage plant after implementing the production forecast (using simple linear regression) formula designed in chapter 3 and implemented in chapter 4. This method has optimized the number of beverage sold in the plant by reducing previous existing errors (stop in production due to shortage of stakeholders while demand is high). Figure 3-2 displayed the production output before the production forecast method was implemented and Figure 5-5 shows how the production output has increased by reducing unnecessary stop in production. The main variable in this optimization method is the beverage production output or beverage number sold itself.

All figures in this section have interpreted how the first objective of the study which was to optimize production output by developing weekly production forecast to avoid production shortage while demand is still high, was achieved by displaying results of weather and production forecast experimental results in both web browser and PLC monitoring interface. A summary of the plant production after implementing the optimization method is also displayed as oppose to initial production results.

5.2 Hourly production tracking report

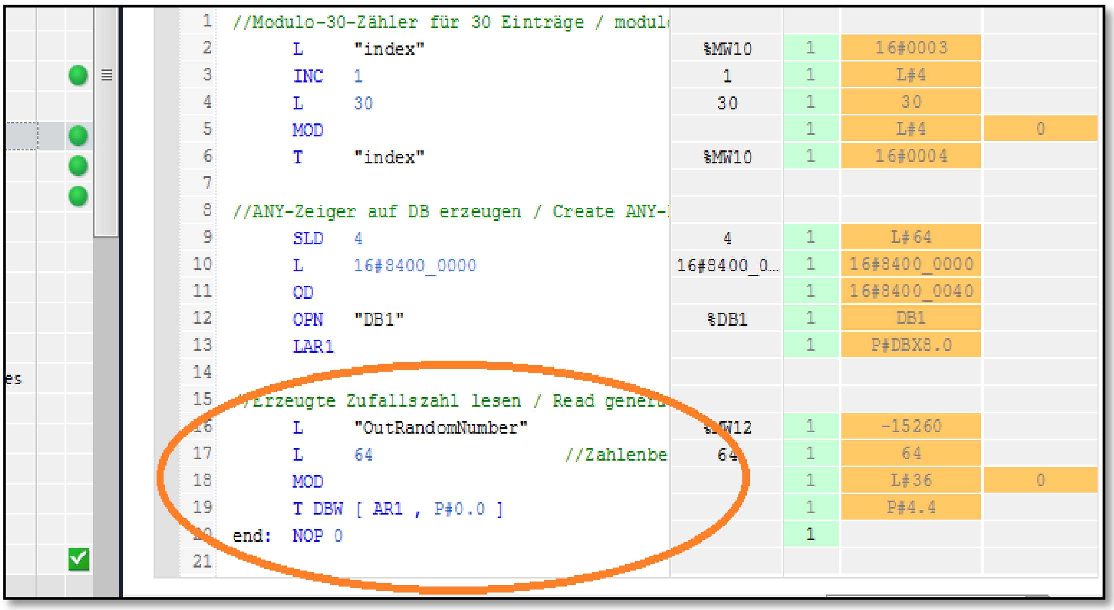
As previously mentioned, our research doesn’t focus in building any type of hardware for the automation of a bottling process but rather in computing intelligent strategies in the PLC to optimize production. For the hourly production tracking section, our aim is to reduce human intervention through auto-parameter configuration and to automatically get the production closer to the daily target despite faults, errors or delays that are almost always present. For testing and simulation of this method, we collected data corresponding to bottles hourly count from a random number generator function. This random number generator function is programmed in the PLC based on linear shift register feedback and modulo operations. This random number generator is programmed based on a 16 bits shift register and can output numbers between 1 and 65535. Figure 5-6 displays the random generator function principle.



In each cycle, the register is shifted to the left by one bit and the bits 15, 13, 12 and 10 are operated on by XOR. The result of the XOR operation is the input bit of the shift register. In the case of a 16-bit shift register, the range of possible values is 1 – 65,535. The algorithm excludes the zero value.

Figure 5-6: Siemens Random generator function principle

For testing purpose, we assumed the daily target of the plant Q to be equal to 1000 bottles. Using formula (3.11) for the theoretical hourly target H_T, we can calculate its actual value assuming a number of working hours equal to 16 (two shifts) as earlier mentioned. After substituting these two variables in (3.11), H_T is equal to 62.5 bottles an hour rounded off to 63 bottles an hour. In normal circumstances, the actual hourly count of the bottling plant will never be higher than this value at a daily target of 1000 bottles. The range of the random number generator needs to be narrowed, in the PLC code as shown on Figure 5-7, to only generate numbers between 0 and 63 which is the expected hourly count.



```
//Modulo-30-Zähler für 30 Einträge / modul
      L      "index"                      %MW10       1   16#0003
      INC    1                            1           1   L#4
      L      30                           30          1   30
      MOD                                             1   L#4            0
      T      "index"                      %MW10       1   16#0004

//ANY-Zeiger auf DB erzeugen / Create ANY-
      SLD    4                            4           1   L#64
      L      16#8400_0000                 16#8400_0…  1   16#8400_0000
      OD                                              1   16#8400_0040
      OPN    "DB1"                        %DB1        1   DB1
      LAR1                                            1   P#DBX8.0

//Erzeugte Zufallszahl lesen / Read gener
      L      "OutRandomNumber"            %MW12       1   -15260
      L      64              //Zahlenbe   64          1   64
      MOD                                             1   L#36           0
      T DBW [ AR1 , P#0.0 ]                           1   P#4.4
end:  NOP 0                                           1
```

Figure 5-7: Random number range code in PLC

The random number range code in the PLC, represented on Figure 5-7, is the tool used in the research to test the hourly production tracking function integrated in the bottling plant to minimize human intervention during the production process. This new feature was developed in Chapter 3 under the automatic parameter configuration section whereby some formulas were designed to closely track the actual hourly bottling production versus the targeted ones and activate overtime production run whenever needed. These formulas were implemented in the PLC in Chapter 4. The random number range code is used in this section to automatically generates multiple possibilities of hourly production output (without being manually forced) that are loaded into the designed model to study the system response; for example: whether or not overtime is automatically activated, how is the hourly count compared to the targeted one etc.

At each working hour of production day from t_0, beginning of shift, those random numbers are being saved into a data base and exported in the S7-1200 data logging function. The data logging saves each data into a file in a csv format accessible through the PLC web server described in Chapter 4. We generate data logs for the targeted bottles count, the actual bottles count during normal working hours, the bottles count during overtime and at the end, a compilation of all these values to demonstrate the merit of our strategy. The demonstration of the results is done graphically using the csv data logs information. Figures 5-8 and 5-9 are the statistics of these data log files.

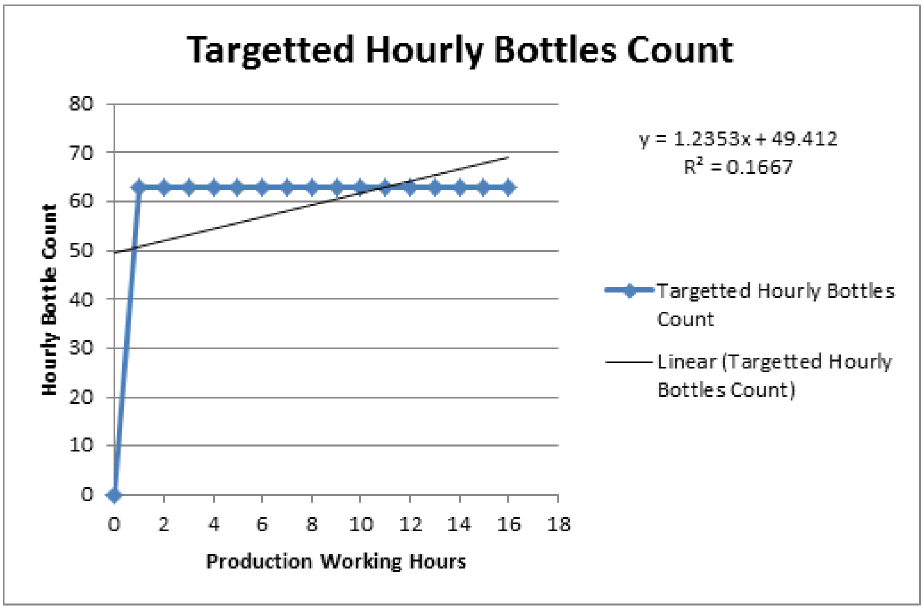


Figure 5-8: Targeted bottles count per hour

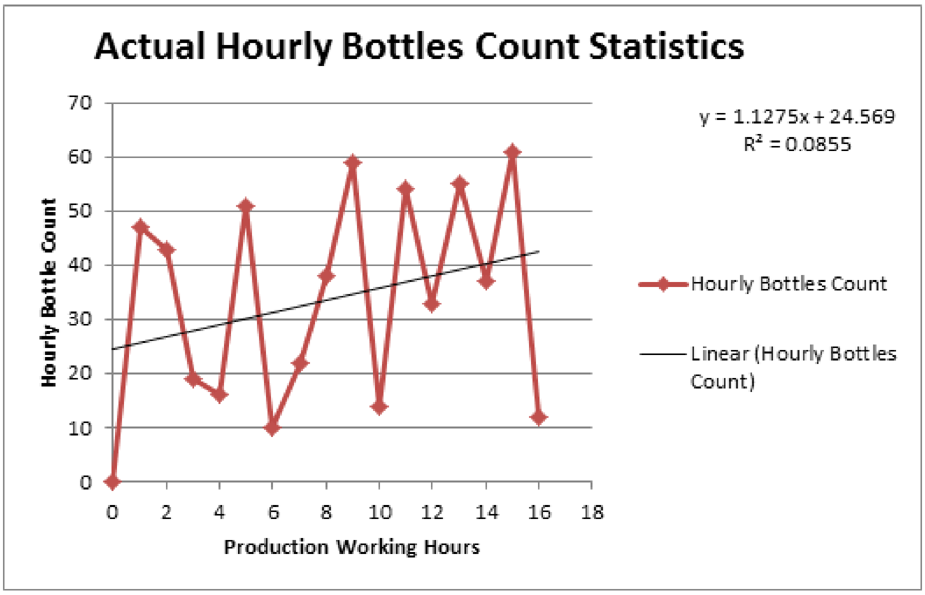


Figure 5-9: Actual bottles count per hour until end of production

Figures 5-8 and 5-9 represents the targeted bottles count per hour and the actual bottles count per hour until the end of the production respectively. The production tracking method to minimize human intervention and push production output as close as possible to the daily target, lay on the principle of closely monitoring the actual bottling production per hour as opposed to the target per hour automatically generated by the PLC (though the theoretical model programmed). The gap between these two variables, targeted hourly production and actual hourly production, is the heart of the method design principle. It gets assessed by the PLC at the end of the production to automatically activate a start of overtime run. Figures 5-8 and 5-9 are therefore essential in the design.

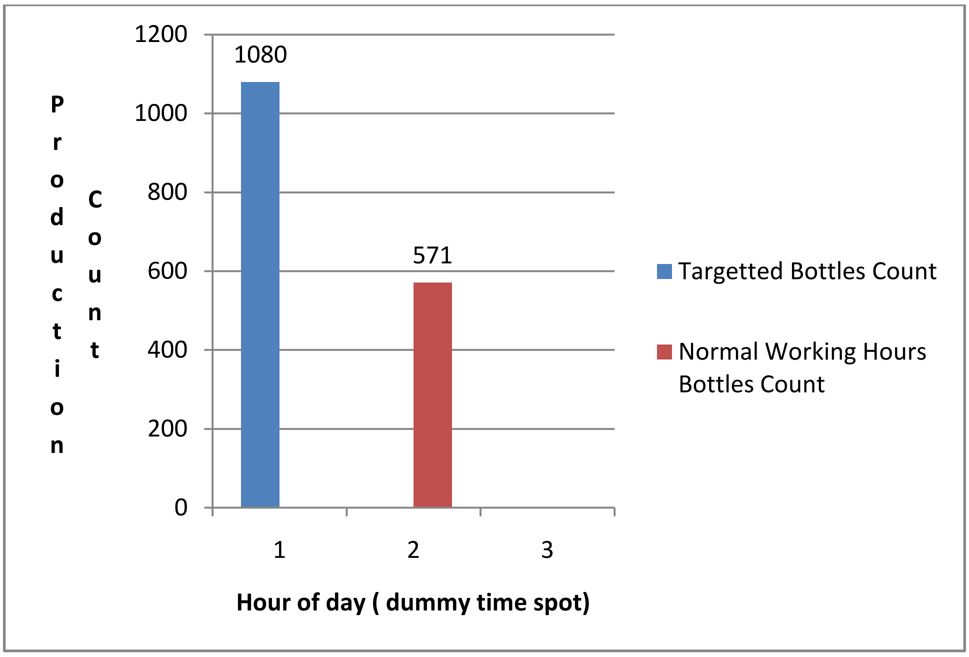


Figure 5-10: Summary of Targeted Bottles Count versus Actual Bottle Count

From Figure 5-10, we can clearly observe the difference between the daily target and the actual count of bottles at the end of the normal working hours. This difference can be caused by any unforeseen events, faults, errors, incompetent operators, bottles stolen etc. Towards the end of the shift, as part of our strategy, equations (3.14) and (3.15) will automatically run in the PLC program. Knowing experimental values of all parameters used in these equations as implemented in Chapter 4: $Q = E_T = 1000$, OP=20%, $CC_{(f)}$ read from the graph in Figure 5-10 as 571, the results of these two equations (3.14) and (3.15) are calculated as follows respectively:

$$ORS = 1000 - \left(1000 \times \frac{20}{100}\right) = 800$$

If $571 \leq 800$ parameters for overtime at full speed applies and overtime Run starts.

Figure 5-11 is the bottle count obtained during overtime hours:

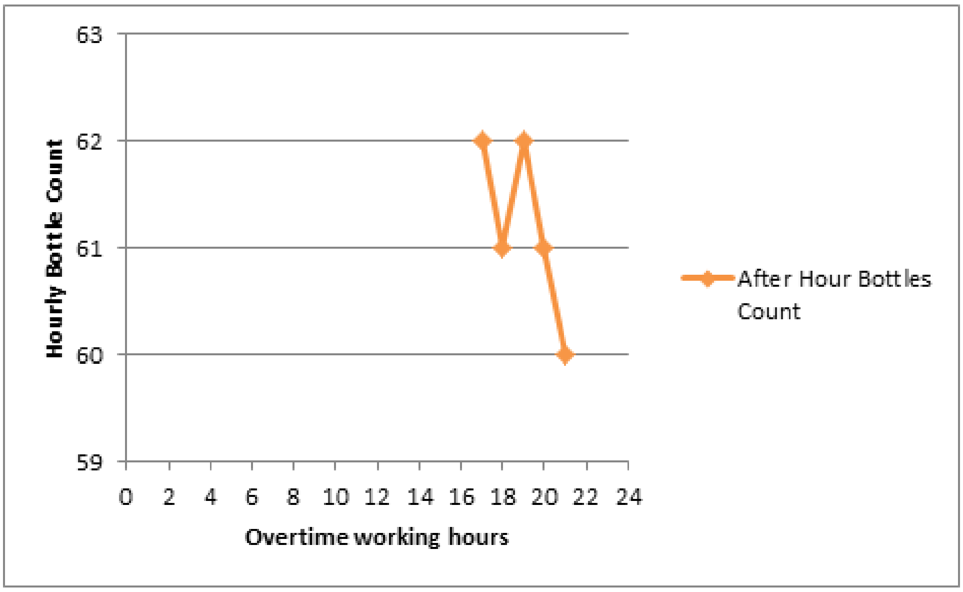


Figure 5-11: Overtime (After hours) bottles Count

Figure 5-11 represents the production hourly count during overtime run, automatically started at the end of the normal production shift. As oppose to other production count figures, e.g. Figure 5-9, the overtime run does not start at the beginning of the shift time t=0 but as soon as the normal shift ends; t=16 as per Figure 5-9. The overtime run stops either when the overall production count reaches a closer percentage to the targeted count or when the operator over rights the command and decides to manually stop the system. As for the actual hourly production count, it can be noticed that there is still a gap between the overtime hourly production count and the hourly targeted production count. This could be due to various factors such as unforeseen faults in the system, slow operators loading bottles etc.

Let's now look at Figure 5-12 which is a combination of all the bottles count computed for the day: targeted count, actual bottle count during normal working hours, bottles count during overtime and total daily production count. The total daily production count is the sum of the actual bottles count during normal working hours and the overtime bottles count.

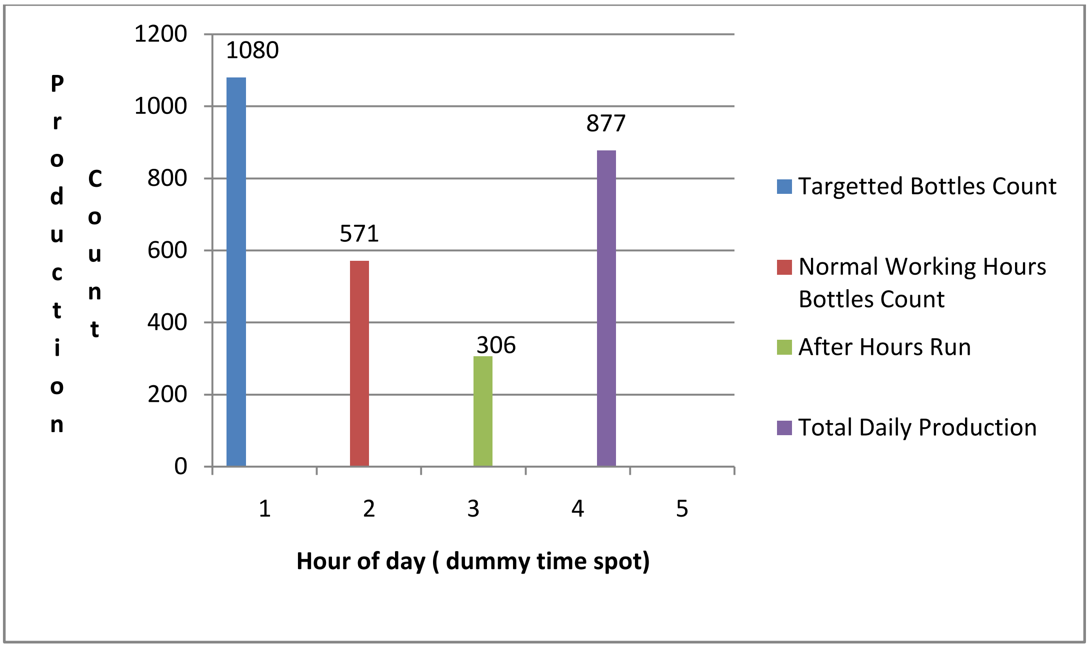


Figure 5-12: All bottles counts computed for the day

At the end of the day, after our strategy has been implemented, only two bottles count will really impact the production: the targeted bottles count and the total daily bottles count represented by Figure 5-13.

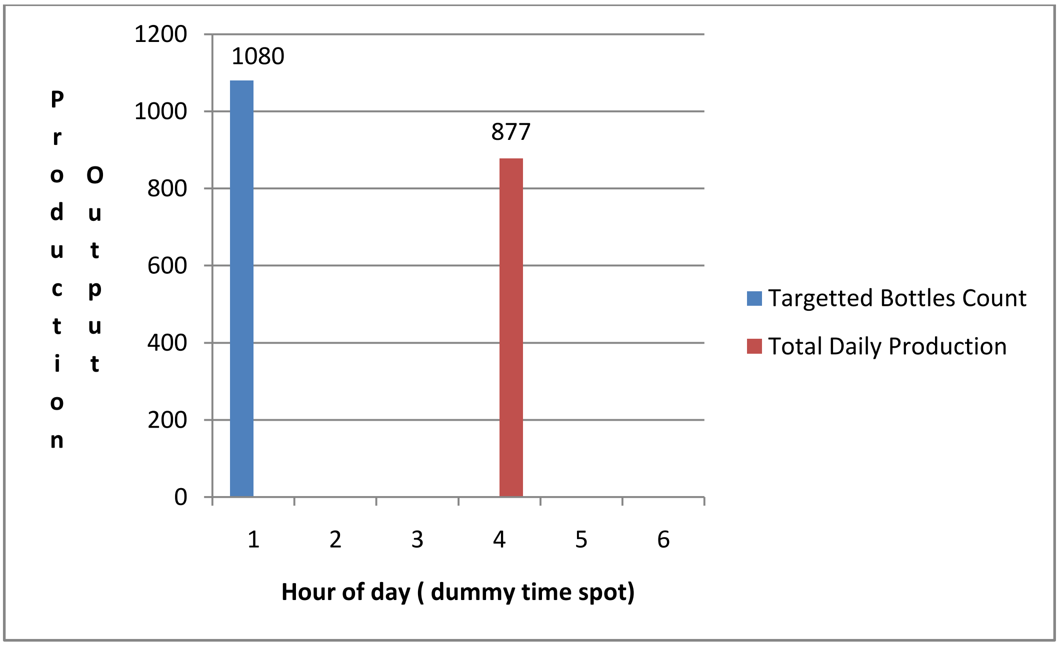


Figure 5-13: Targeted bottles count versus Total Bottles count after Overtime Run

There difference of what the production should have been for the day without implementing our strategy in Figure 5-10 and what it is actually after programming our PLC with our strategy in Figure 5-13 is very clear. Our strategy added value to the daily production, allowing it to get closer to its daily target. All these operations and calculations were done real time in the PLC. The decision for overtime run was done instantaneously and automatically after computation of results calculations. No need for a supervisor to interpret results before decision making. It saves considerably time to the process and gets the production closer to the daily target compared to what it should be without overtime run; and all this with no human intervention.

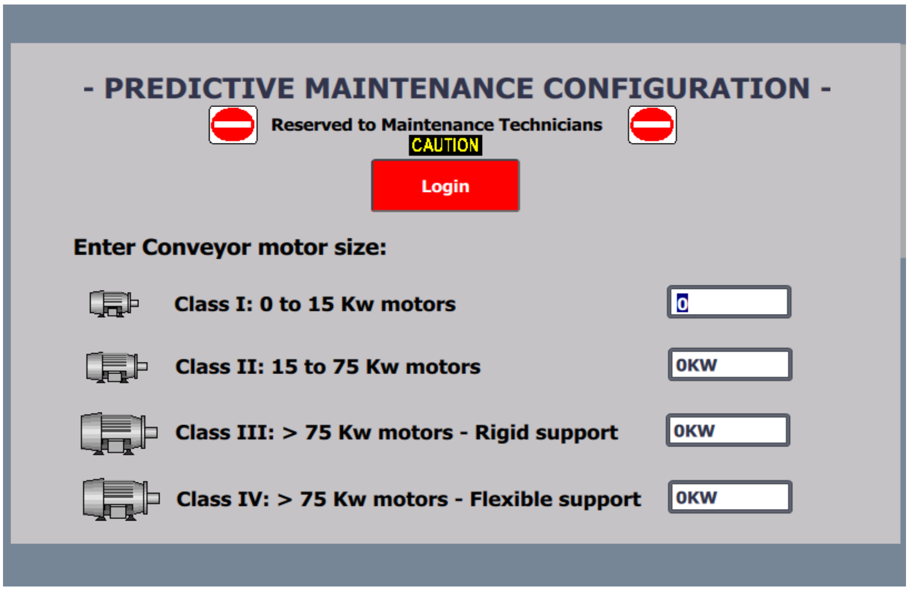
Figures in this section have interpreted one of the research goals of the study intended to minimize human intervention in the system by configuring automatic parameters configuration. Production was modelled as a time function to track closely the daily production in an hourly basis and activate on time adequate actions such as overtime run when production target is not met. Results of the implementation of auto parameter configures were graphically displayed.

5.3 Predictive maintenance on conveyor motor

In Chapter 4, we discussed how predictive maintenance rules for conveyor motors are flexibly programmed in PLC and also how a python script is configured for direct notification to experts. In order to view results of the predictive maintenance technique, we first developed a SCADA GUI for the plant. The SCADA system was used to both display externally PLC's program operation and to allow operators interact with the overall system: control operations, input values etc.

For predictive maintenance, we designed two main important screens in the SCADA:

- **The parameters setting screen (displayed on Figure 5-14)**: where the maintenance technicians edit or configure new predictive maintenance rules for different motors' size, different vibration criteria. For this study the vibration criteria on Table 1 is used.



- PREDICTIVE MAINTENANCE CONFIGURATION -

Reserved to Maintenance Technicians

CAUTION

Enter Conveyor motor size:

Class I: 0 to 15 Kw motors	0
Class II: 15 to 75 Kw motors	0KW
Class III: > 75 Kw motors - Rigid support	0KW
Class IV: > 75 Kw motors - Flexible support	0KW

Figure 5-14: Predictive maintenance settings for motors

Only maintenance technicians are entitled to do configurations, therefore user identification through the login button on Figure 5-14 is required to keep track of changes performed in the system.

For more flexibility when editing or configuring predictive maintenance rules for the system, PLC and SCADA are working together based on the programming structure in Figure 5-15 below:

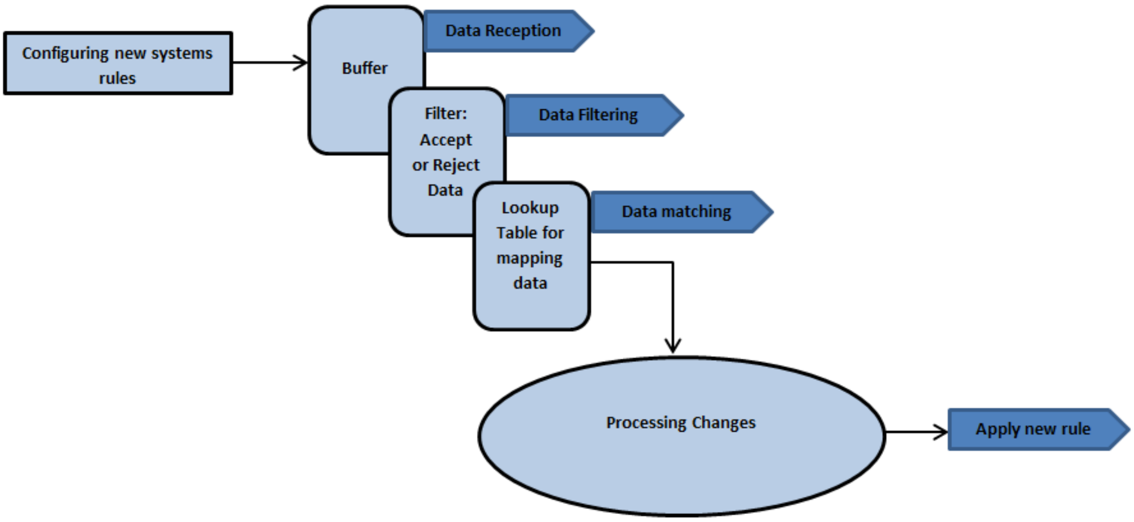


Figure 5-15: Flexible predictive maintenance rule programming structure

Data entered in Figure 5-14 are first received by the SCADA in a buffer, filtered, mapped to the corresponding class of motors predefined in the PLC software and then applied to the operations:

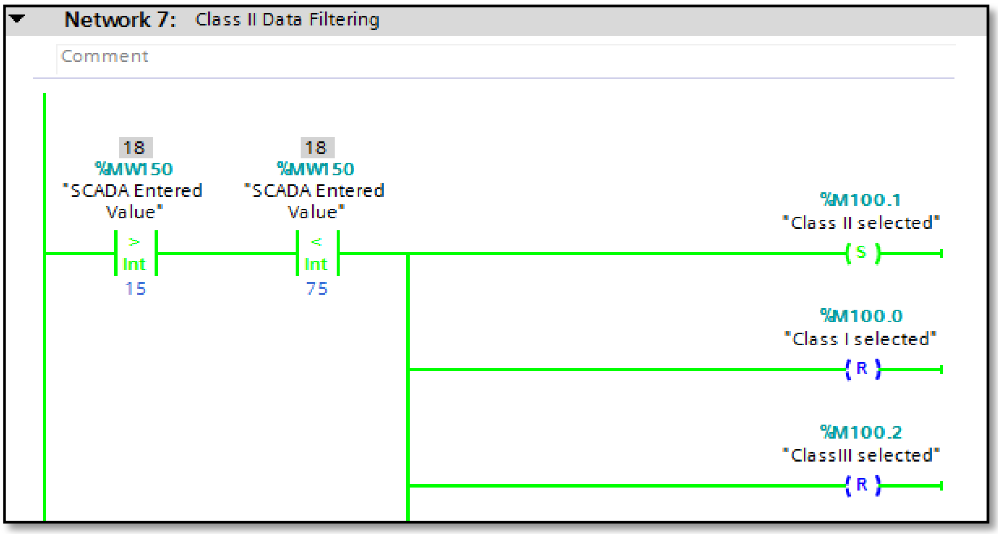


Figure 5-16 Class II Data Filtering in PLC software

Figure 5-16 is an example of data filtering in PLC language for class II as per Table 1. The entered data in the SCADA is equal to 18 KW which matches the second motor's class (motors sizes between 15 and 75 KW). After filtering of data, there is data mapping which is the selection of the corresponding class II function to be applied for PLC operations. Figure 5-17 shows how the data mapping is done in the PLC code.

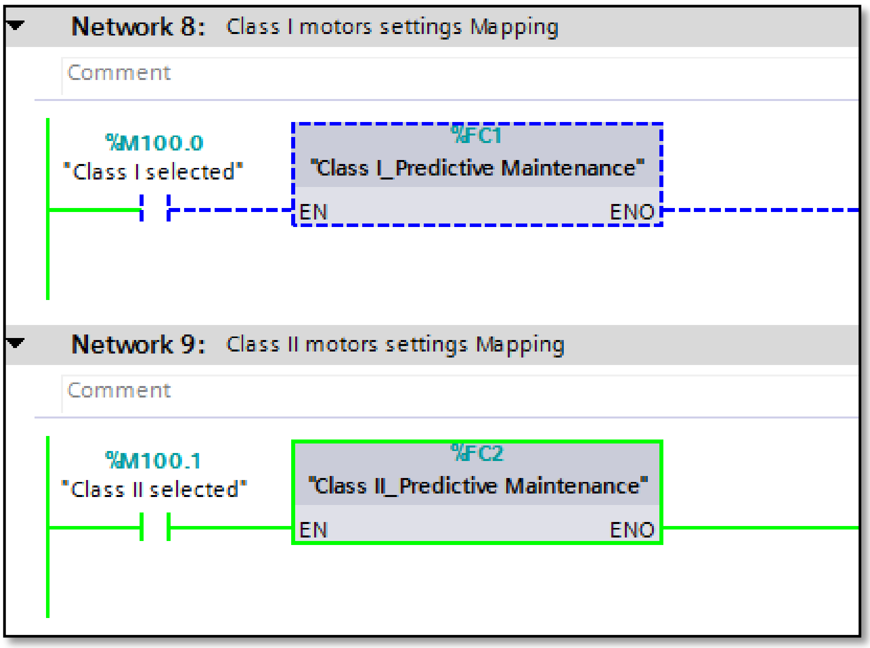
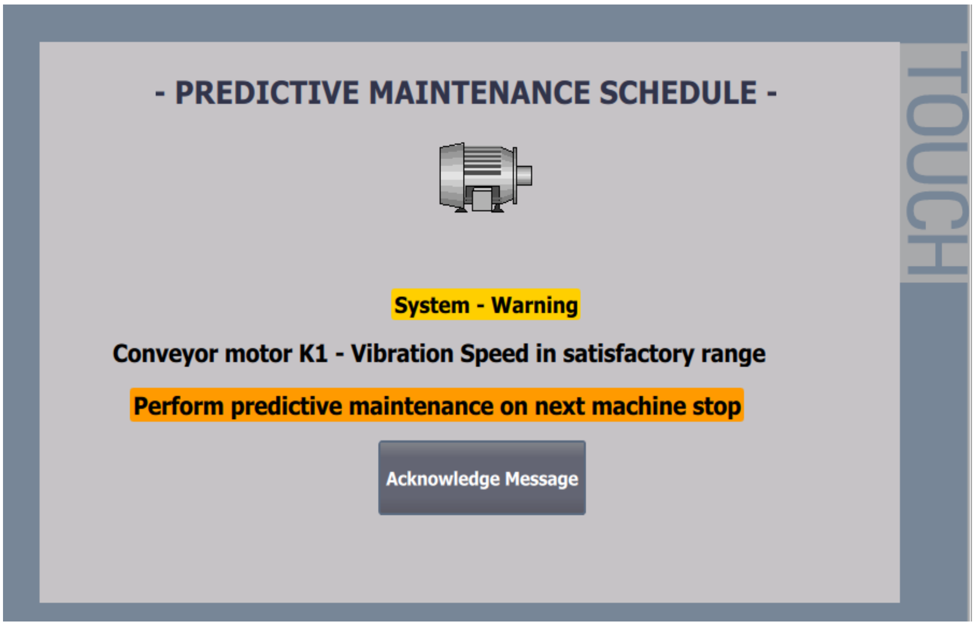


Figure 5-17: Class II motors data mapping

- **The message schedule screen**: where notifications of reactive or predictive maintenance are displayed for actions to be taken. Figure 5-18 shows one of the SCADA message screens' notifications for predictive maintenance.



- PREDICTIVE MAINTENANCE SCHEDULE -

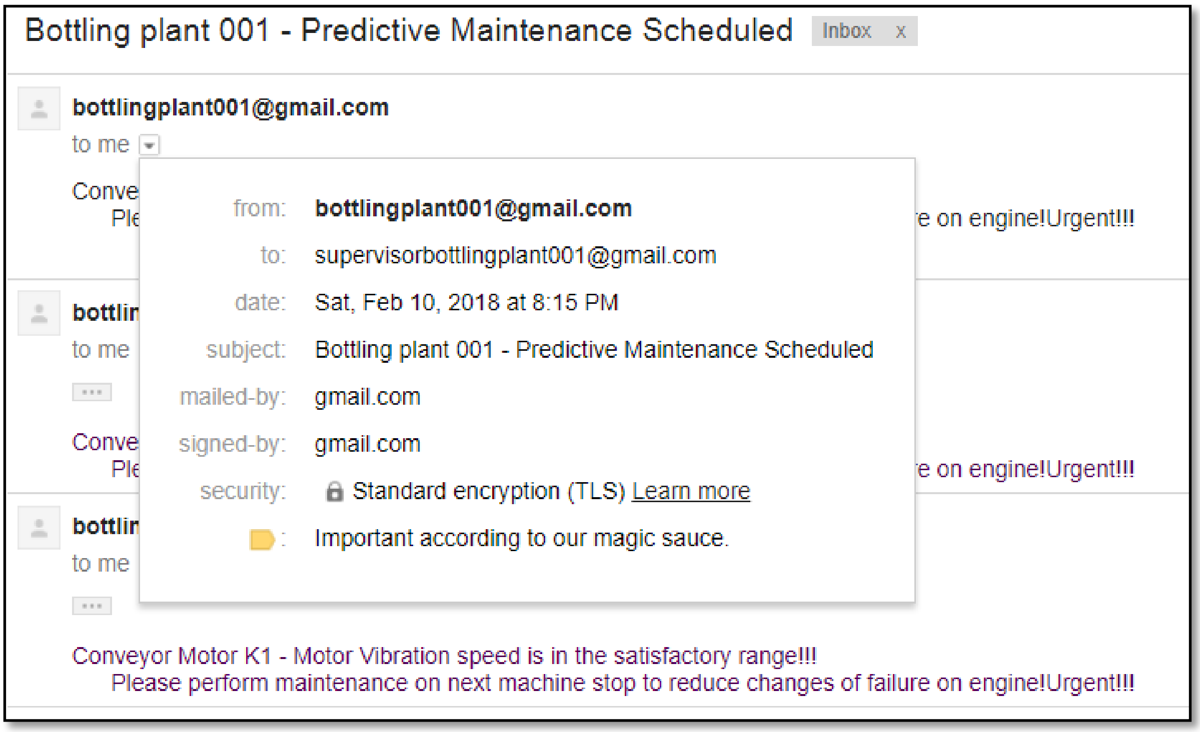
System - Warning

Conveyor motor K1 - Vibration Speed in satisfactory range

Perform predictive maintenance on next machine stop

Figure 5-18: Predictive maintenance schedule message screen

As soon as predictive maintenance is scheduled and displayed on the SCADA's screen, the running python script reads a value change in the csv file displayed in Figure 4-22 written by the PLC and automatically sends an email to the supervisor in change of maintenance for the bottling plant. In our platform, the email is sent more than once (six times) corresponding to the number of lines in the CSV file to make sure that the supervisor gets properly notified. The python compiler's reaction when predictive maintenance is detected is displayed on the Appendix (section 9.5). The compiler continuously runs on the SCADA panel pc. As soon as the compiler successfully runs, email-notifications are sent directly from the bottling plant to the intended supervisor as shown on Figure 5-19.



Bottling plant 001 - Predictive Maintenance Scheduled

bottlingplant001@gmail.com
to me

Conve
Ple
e on engine!Urgent!!!

bottlin
to me

Conve
Ple
e on engine!Urgent!!!

bottlin
to me

from: **bottlingplant001@gmail.com**
to: supervisorbottlingplant001@gmail.com
date: Sat, Feb 10, 2018 at 8:15 PM
subject: Bottling plant 001 - Predictive Maintenance Scheduled
mailed-by: gmail.com
signed-by: gmail.com
security: Standard encryption (TLS) Learn more
: Important according to our magic sauce.

Conveyor Motor K1 - Motor Vibration speed is in the satisfactory range!!!
Please perform maintenance on next machine stop to reduce changes of failure on engine!Urgent!!!

Figure 5-19: Predictive maintenance E-mail received by Supervisor from bottling plant

In this section, results of predictive maintenance and early fault detection were displayed. Figures in this section have interpreted results from the online PLC monitoring portal to the SCADA GUI as well as the targeted automatic emails sent by the system to the plant supervisor.

5.4 Bottling production key data and motor vibration speed accessible on cloud

In Chapter 4, databases were created in the local server as well as in the online reporting tool, Clic Data, to generate an online dashboard. A dashboard of type Column Chart was selected for the report. Three dashboards are created using the three previously highlighted data: the actual hourly bottling production, the targeted hourly bottling production and the overtime hourly bottling production. On Figure 5-20 there is view of the actual hourly bottling count. On Figure 5-21 a combination of the actual hourly count versus the targeted one is displayed. And finally all the hourly bottling counts of the day: the two previous ones including the overtime hourly count if available are shown on Figure 5-22. The view of the dashboards below is adapted to a laptop screen but is also available in smartphones and tablet view.

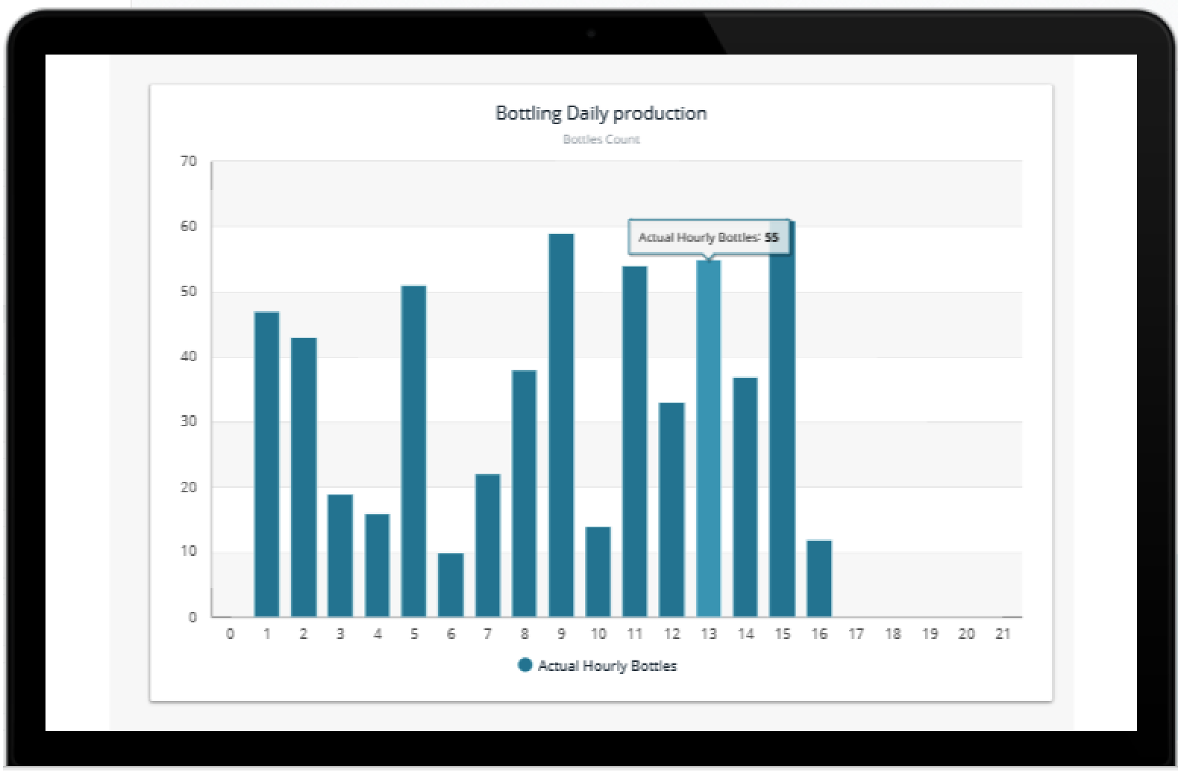


Figure 5-20: Actual hourly bottling production count online from Clic Data dashboard

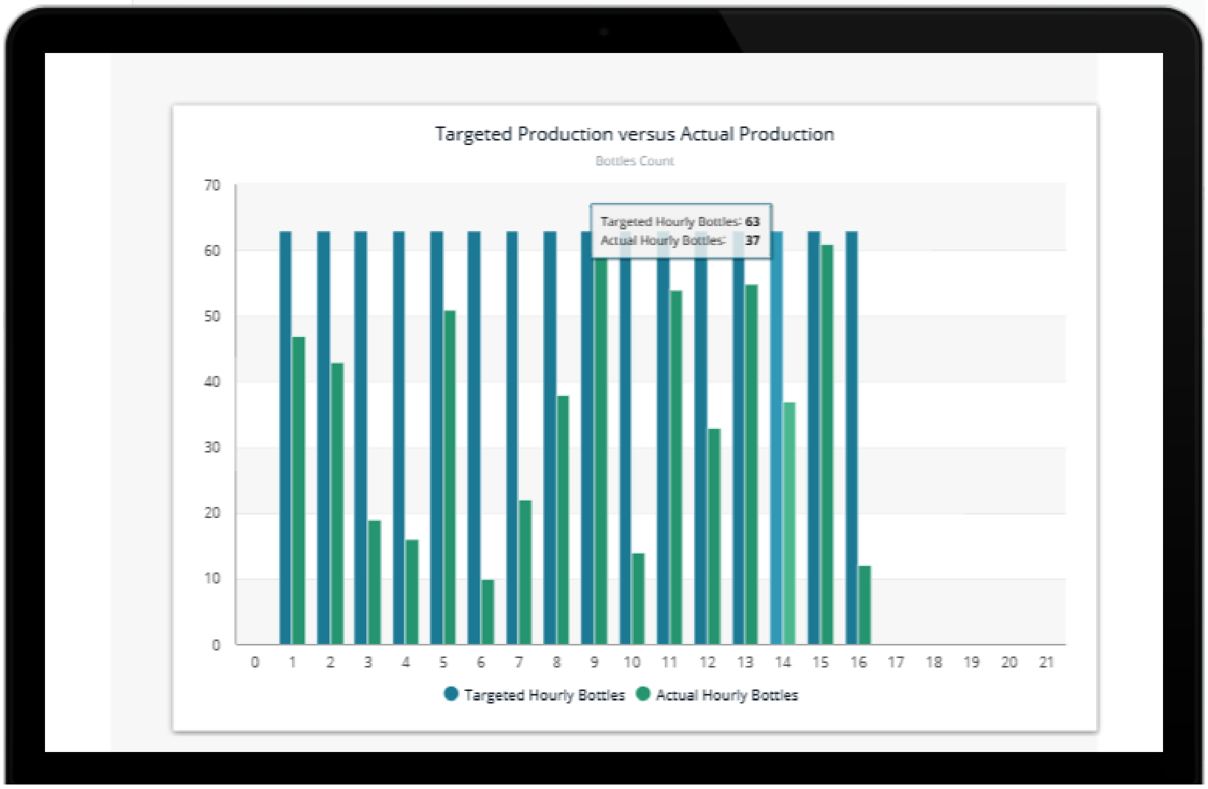


Figure 5-21: Actual hourly bottling count versus targeted hourly count online from Clic Data

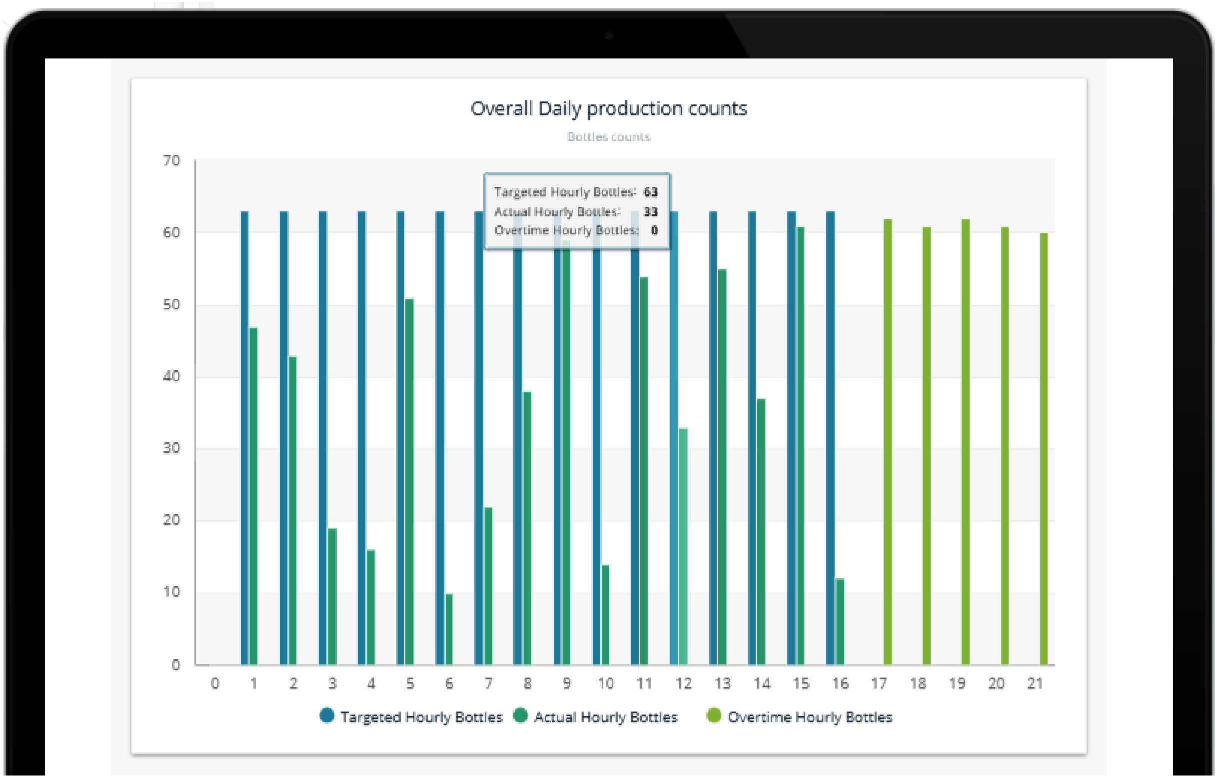


Figure 5-22: Overall daily bottling counts online from Clic Data

The dashboard is accessible simply by connecting on the internet and entering the live link: https://theraintel.clicdata.com/v/wViqpPElBI3K

Figure 5-20 is the first dashboard that appears on the above live link report. The first dashboard has been configured in a way that by double clicking on the graph, it drills down to the two dashboards and back to itself. This way management has a clear overview of all key information of the bottling plant from wherever they are.

Another decentralized monitoring dashboard of the motor vibration evolution is available in a cloud based reporting tool (ClicData). Wherever they are, when connected to the internet, supervisors can monitor the evolution of different motor vibration speed states and anticipate preparation of a maintenance schedule. The online dashboard represents some self-explanatory graphical gauges of the Motor vibration severity criteria in Table 3-2. Data on the dashboard is updated real-time from a local MySQL server which receives real-time vibration speed from the PLC CSV file database. The PLC itself reads vibration speed from the vibration sensor mounted on the conveyor motor (Figure 4-2). Figures 5-23, 5-24, 5-25 and 5-26 display the gauges states available on the online monitoring tool:

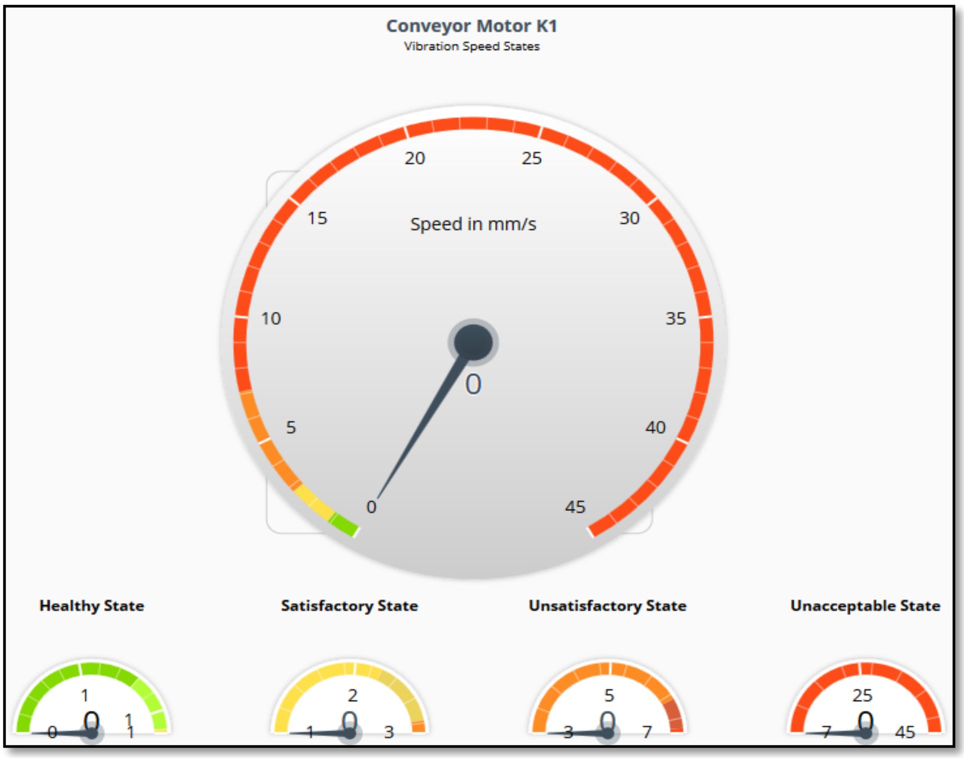


Figure 5-23: Conveyor Motor Vibration in Healthy State

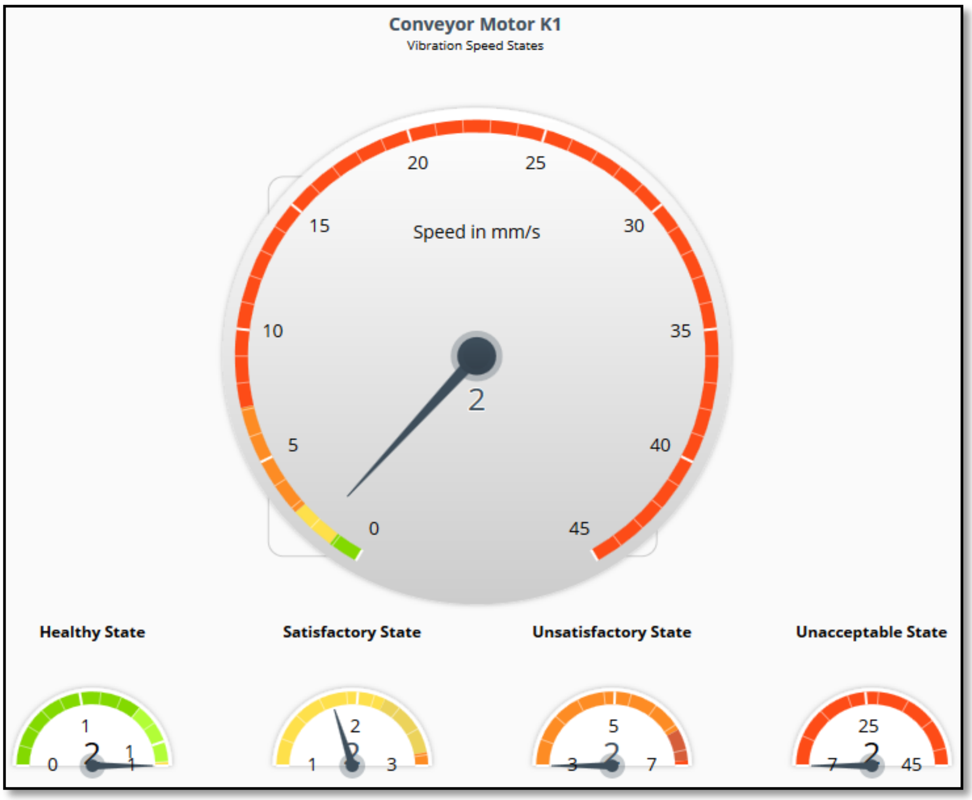


Figure 5-24: Conveyor Motor in Satisfactory State

Figure 5-25: Conveyor Motor in Unsatisfactory State

Figure 5-26: Conveyor Motor in Unacceptable state

The dashboard is accessible through an internet browser link:

https://theraintel.clicdata.com/v/YgPGRCamDflO

Last but not least, decentralization of bottling plant production was interpreted in this section with figures displaying different portals, cloud platforms available allowing an off-site monitoring of the production and critical machine components such as conveyor motors.

Chapter 6 : CONCLUSION AND FUTURE WORK

The aim of this chapter is to give a summary of the overall work done in the study, to discuss objectives achieved as well as possible direction for future work.

6.1 Summary of the study

This study was about the development of high-tech automation bottling process for SMEs using PLC, SCADA and basic concepts of Industry 4.0. To stay competitive in an ever growing market, the challenges of any industrial enterprise is to enhance product quality, reduce cost and improve on-time delivery. In the beverage industry, the increasing demand of products has boosted the need to introduce more advanced and efficient automation techniques to improve production processes. Small to medium scale beverage enterprises, forming a fairly high part of the market share, are also facing the same challenges. Unfortunately, most of them remain in a *statu quo* because of their perception that adopting high tech, current and advanced automation production techniques is reserved to big companies which can afford high expenses. One of the strategies used in this research to reduce cost involved in the development was to include most available infrastructure in the small bottling plant such as PLC and SCADA system, investing more on intelligent programming solution to integrate their advanced properties with current trend of automation.

Literature review conducted in this research has shown that, while moving toward the fourth industrial revolution, most previous researches done to improve the automation of SMEs bottling processes have focused in implementing older trends of automation: mechanical changes, mass production and digitalization, barely approaching the current one, I40 or smart factory. I40 focuses in cyber physical systems, Internet of things and cloud computing. These new concepts still in slow deployment in the African market will soon become the way to the future for all industrial automation companies high, medium or small. I40 brings several advantages to production processes without necessarily adding exorbitant cost. A very good knowledge of the concept, a combination of its basic principles with current production structures and components in place could make a very big difference in the overall plant's production.

Our research used this approach to optimize operation of a SME bottling process by using two basic principles of I40: interconnection and decentralization. The internet plays a very important role in I40. In this research, it allows all different stakeholders of the bottling process to be effectively interconnected throughout production stages and to share important information. Decentralization, where information accessibility is no longer limited to a central physical location of the bottling plant, is achieved through I40.

6.2 Objectives achieved

A similar methodology was used to achieve objectives of this research. A collection of data was first done in the small bottling plant over a period of time to get a common understanding of the system behaviour, understand weaknesses of the system and propose ultimate solutions. Afterward, a theoretical modelling of solutions was done for better accuracy before being practically implemented in different platforms.

The high tech automated bottling process we developed in this research for SMEs using the current trend of automation has achieved the following:

- **A Higher production rate**: The new production technique increases production output rate by implementing prediction of the weekly production. This allows the plant to order in advance correct quantities of all necessary stakeholders to production and reduces the risk of running short of goods while demand is still high. The ordering of equipment is done automatically through the internet with an instant connection between bottling production output and suppliers. By closely tracking the output production as a function of time, the system is also able to compensate ordering of goods by requesting additional equipment in instances where the actual production is higher than predicted ones.

  Our new production technique also reduces failure rates on plant conveyor motor by initiating predictive maintenance. The motor's vibration speed is constantly monitored and generates alarms through the PLC. A direct email notification is then sent by the system to the maintenance expert for expeditious actions to be taken reducing unnecessary message conveying delay.

- **Better delivery time**: Because unexpected errors or failures that could affect daily production target always occur in a plant, the high tech bottling process implements another technique to closely track production by comparing the daily target to the actual production output and automatically changing parameters to start an overtime production run when production is below a predefined unacceptable percentage. This action results in better production delivery time because the plant does not depend on operators to manually calculate production loss rate and decide after consulting management whether or not the plant needs to continue production.

- **Easy access of plant information:** In order to allow management level, who is not always physically present on the plant premises, to monitor bottling production, the high tech bottling process sends key information: actual hourly production, targeted production and overtime hourly production, to a cloud based reporting tool. Management is now able to view progress of bottling plant from wherever they are, once connected to the internet, without depending on supervisors' and operators' reports.

The development of this high tech is done at low cost. The very important element for the realization of this study is the Internet. PLC and SCADA panel pc are bottling plant available resources. Most programming software installed in the SCADA panel pc like python, MySQL database are open software. The additional hardware used is the speed vibration sensor. Though there are couple of open source reporting tool that could be used, we selected the cloud based reporting dashboard tool, Clic Data, which is one of the most affordable with possibility of trial period.

6.3 Future work

As mentioned earlier, we used only basic principles of I40 in our techniques to optimize production. I40, through IoT components, could effectively be used to allow customers place their orders directly to the bottling plant, specify their preferred flavour, and monitoring the progress of the order as going through the process. This is possible by using IoT devices, such as RFID tags, at different stages of the bottling process. The installed IoT devices will report actual positions of the ongoing bottles in the production lane and customers could be able to get approximate of delivery time for their orders. In this research we only considered the interaction between the bottling plant and stakeholders suppliers without directly including customers.

For better control of all incoming customers’ orders with diversified information like flavours, bottles sizes, quantities etc. an efficient use of cloud computing to manage all these data will have to be implemented to avoid overloading the central controller. In this research, the cloud application was only used for monitoring purpose. Maintenance experts are notified by email when predictive maintenance schedule is generated. It is only a one way communication from the bottling plant to the experts where a response or acknowledgement from the experts does not affect the bottling plant. In future, for better interaction and faster response time from experts, an acknowledgment feature could be implemented in the communication channel between bottling plant and experts to allow them react faster in every generated maintenance schedule.

7. LIST OF PUBLICATIONS

Kahiomba Sonia Kiangala and Zenghui Wang, "Initiating predictive maintenance for a conveyor motor in a bottling plant using Industry 4.0 concepts", The International Journal of Advanced Manufacturing Technologies (JAMT), August 2018, Volume 97, Issue 9-12, PP. 3251-3271 (ISI master Indexed accredited journal).

Kahiomba Sonia Kiangala and Zenghui Wang, "An Industry 4.0 approach to develop auto parameter configuration of a bottling process in a small to medium scale industry using PLC and SCADA", submitted to the International Conference on Sustainable Materials Processing and Manufacturing, SMPM 2019, Sun City Resort, South Africa, 08 – 10 March 2019

Kahiomba Sonia Kiangala and Zenghui Wang, "Implementation of Basic lean management and Industry 4.0 concepts to optimize production of small local beverage supply industry", submitted to the International Journal of Production Economics (Industry 4.0 & Production Economics) by the Elsevier Editorial publishers.

8. BIBLIOGRAPHY

Abashar A.I., Mohammedeltoum M.A. & Abaker O.D., 2017. Automated and Monitored Liquid Filling System using PLC Technology,*International Conference on Communication, Control, Computing, and Electronic Engineering (ICCCCEE), Khartoum, Sudan*, PP. 1-5

AB Elektronik, 2018, *Inductive proximity sensor/ cylindrical/ IP67/ digital* [online]
Available at:
http://www.directindustry.com/prod/ab-elektronik/product-16540-492145.html [Accessed 20 May 2018]

Adolph, S., Tisch, M. and Metternich, J., 2014. Challenges and approaches to competency development for future production. *Journal of International Scientific Publications–Educational Alternatives*: PP. 1001-1010.

Ahuja H., Singh A., Tandon S., Srivastava S. & Pal S., 2014. Automatic Filling Management System for Industries,*International Conference on Advanced Developments in Engineering and Technology (ICADET-14), INDIA, International Journal of Emerging Technology and Advanced Engineering* Website: www.ijetae.com (ISSN 2250-2459 (Online), Volume 4, Special Issue 1), PP. 241-244

Al-Hawari T., Aqlan F., Al-Buhaisi M. & Al-Faqeer Z., 2010. Simulation Based Analysis and Productivity Improvement of a Fully Automatic Bottle Filling Production System*:* A practical case study, *Second International Conference on Computer Modelling and Simulation*, PP. 195-199

American Environments Company, Inc., 1999. *Vibration Equations* [online]
Available at:
http://www.aeco.com/download/vibration.pdf [Accessed 11 January 2018]

Anaheim Automation, 2017. AC Motor Guide [online]
Available at:
http://www.anaheimautomation.com/manuals/forms/ac-motor-guide.php [Accessed 20 April 2018]

Atzori, L., Lera A., Morabito G., 2010. *The internet of things: A survey In: Computer networks*. Vol 54, PP. 2787-2805.

Berger H., 2013. *Automating with SIMATIC_ Controllers, Software, Programming, Data Communication, Operator Control and Process Monitoring*, 5th Edition, Publicise

Carrow R.S., 2011. *Electrician's technical reference: Variable frequency drives,* Delmar Thomson Learning, PP. 45

Chakraborty K., Roy I. & De P., 2015. Controlling Process of a Bottling Plant using PLC and SCADA, *Indonesian Journal of Electrical Engineering and Informatics (IJEEI),*Vol. 3, No. 1, pp. 39~44

Chang H.F , Chang C.P., Lin W.T & Li Y.G, 2016. “A Study of application of Assessment Modes in Development of Smart Automatic Production Systems in Industry 4.0”, *International Conference onFuzzy Theory and Its Applications (iFuzzy),* PP. 1-8

Cheminod M., Durante L., & Valenzano A., 2013. Review of security issues in industrial networks, *IEEE Transactions on Industrial Informatics.*, vol. 9, no. 1, pp. 277–293
Chen H., Zhou C., & Zhu W., 2009. A hybrid neural-genetic approach for reconfigurable scheduling of networked control system, *Proc. 1st ACM/SIGEVO Summit Genetic and Evolutionary Computation Conference.,* Shanghai, China, pp. 33–38

Choudhary H., 2012. *Interrupts & Programming 8051 Hardware interrupts* [online]
Available at:
https://www.engineersgarage.com/tutorials/interrupts-8051-interrupt-programming [Accessed 17 July 2017]

Comac kegging & bottling plants, 2015.*Water bottling plant.*[Online]
Available at:
https://www.comacgroup.com/comac/focus/water-bottling-plant [Accessed 12 March 2018]

Combest S., 2010. *Siemens S7-1200 Compact Controllers* [online]
Available at :
http://www.totallyintegratedautomation.com/2010/02/siemens-s7-1200-compact-controllers/
[Accessed 12 March 2018]

Copa-Data, 2017. ZENON SCADA Products [online]
Available at:
https://www.copadata.com/en/ [Accessed 22 February 2018]

Copa-Data Automotive, 2017.*Process control system – Automotive* [online]
Available at:
https://www.copadata.com/en/process-control-system/automotive/ [Accessed 22 February 2018]

Copa-Data brewing process, 2014.COPA-DATA introduces the new zenon 7.11 sofware release [online]
Available at :
https://www.copadata.com/en/press-room/press-releases/archive-2014/copa-data-introduces-the-new-zenon-711-software-release-5974/ [Accessed 22 February 2018]

Da Cunha M.J., De Almeria M.B., Fernandes R.F. & Carrijo R.S., 2017. Proposal for an IoT architecture in industrial processes, *12th IEEE International Conference on Industry Applications (INDUSCON),* PP. 1-7

Darvas D., Majzik I.&Viñuela E.B., 2016. Conformance checking for programmable logic controller programs and specifications, *11th IEEE Symposium on Industrial Embedded Systems (SIES),* PP. 1-8

Das S., Dutta S., Sarkar A. & Kar S., 2014. Recognition and Disposal of Faulty Bottles in a Bottle FillingIndustry Using PLC and Producing Human Machine Interface by SCADA, *International Refereed Journal of Engineering and Science (IRJES),* ISSN (Online) 2319-183X, (Print) 2319-1821 Volume 3, Issue 5(May 2014), PP-18-23

Daughtry R., 2014. *A sensor for every application* [online]
Available at:
https://www.crossco.com/blog/sensor-every-application [Accessed 20 May 2018]

Deloitte S.A., 2016. *Industry 4.0 - is Africa ready for digital transformation?* White paper report [Online]
Available at:
https://www2.deloitte.com/content/dam/Deloitte/za/Documents/manufacturing/za-Africa-industry-4.0-report-April14.pdf [Accessed 10 February 2018]

Dhiman J. & Kumar E.R.D, 2014.Hybrid method for automatically filling of the chemical liquid into bottles using PLC & SCADA, *International Journal of Engineering Research and General Science,* Volume 2, Issue 6, ISSN 2091-2730, PP.1000 – 1007

Electric4U, 2011.*Permanent Magnet DC Motor* [online]
Available at:
https://www.electrical4u.com/permanent-magnet-dc-motor-or-pmdc-motor/ [Accessed 20 April 2018]

Electrical Engineering 123, 2016.*Introduction to Induction & AC electric motor* [online]
Available at:
https://www.electricalengineering123.com/introduction-to-induction-ac-electric-motor-working-principle-types/ [Accessed 20 April 2018]

Fargo Controls, Inc., 2016. *Capacitive sensors with programmable output* [online]
Available at:
http://www.fargocontrols.com/sensors/capacitive_nonc.html [Accessed 20 May 2018]

Fran R., 2017. *Bottled water wins as government crackdown on sugary drinks* [online]
Available at:
http://www.fdfworld.com/drink/bottled-water-wins-governments-crackdown-sugary-drinks [Accessed 12 March 2018]

Franson D., Encyclopaedia of occupational health and safety, 2016.*Chapter 65-Beverage Industry* [online]
Available at:
http://www.ilocis.org/documents/chpt65e.htm [Accessed 13 March 2018]

Giannetti C., 2017. “A Framework for Improving Process Robustness with Quantification of Uncertainties in Industry 4.0”, *IEEE International Conference on Innovations in Intelligent Systems and Applications (INISTA)*, PP. 189-194

Gidlund M., Lennvall T. &Åkerberg J., 2017.Will 5G Become Yet Another Wireless Technology for Industrial Automation? *IEEE International Conference on Industrial Technology (ICIT)*, PP. 1319 – 1324

Grilo A.M., 2014. *An integrated WSAN and SCADA system for monitoring a critical infrastructure*, IEEE Transactions on Industrial Informatics, vol. 10, no. 3, pp. 1755_1764

Haider, M., 2008.“Bringen Sie Ihre Abfullanlage auf Vordermann! Schwachstellenanalyse an Mehrweg- ¨ Glas-Anlagen”. Brauindustrie (1): 36–37

Haq M.Z., 2014. *DC motors*, Department of Mechanical Engineering Bangladesh University of Engineering & Technology, PP.1-18 [online]:
Available at:
http://teacher.buet.ac.bd/zahurul/ME475/ME475_dcmotors.pdf [Accessed 22 April 2018]

Harpreet S.D, 2017. *ERP Implementation: Top five challenges for SMEs* [online]
Available at:
https://economictimes.indiatimes.com/small-biz/sme-sector/erp-implementation-top-five-challenges-for-smes/articleshow/57034195.cms [Accessed 09 February 2018]

Hou Z. & Wang Z., 2013. From model-based control to data-driven control: Survey, classification and perspective, *Information Science 235*:PP.23–25

Ifm electronics, 2013.*From process monitoring to vibration analysis, condition monitoring systems* [online]
Available at:
http://www.ifm.com/download/files/ifm-efector-octavis-brochure-GB-2013/$file/ifm-efector-octavis-brochure-GB-2013.pdf : PP. 5-12 [Accessed 18 May 2018]

IMI sensors, 2013.*Motor vibration: Detect Mechanical & Electrical motor faults with vibration monitoring instrumentation* [online]
Available at:
https://www.imi-sensors.com/ContentImages/downloads/IMI-App-Motors_LowRes.pdf [Accessed 22 April 2018]

Jackson B., 2016. *The factory of the future with Elysium's Industry 4.0 solution* [online]
Available at:
https://3dprintingindustry.com/news/factory-future-elysiums-industry-4-0-solution-101815/ (Smart manufacturing) [Accessed 03 March 2018]

Jain S.P. & Dr. Haridas S.L., 2014.Energy Efficient Automatized Bottling Plant Using PLC And SCADA with Speed Variable Conveyor Assembly, *Nagpur IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 1, Ver. II , PP. 09-14

Jaspreetkaur P.,Kulkarni R., &Ranjana M.S.B., 2016. On the automated multiple liquid bottle filling system, *International Conference on Circuit, Power and Computing Technologies*, PP. 1-3.

Jones C.T., 1996. *Programmable Logic Controllers the Complete Guide to the Technology*, Patrick Turner Publishing Company

Kar D. & Sing D. Kr., 2015. Design of Reconfigurable Manufacturing System: A Review, *International Journal of Scientific Engineering and Applied Science (IJSEAS)* - Volume-1, Issue-3, PP. 132 -135

Kaur A.&Bansal D., 2016.Monitoring and controlling of continue furnace line using PLC and SCADA, *5th International Conference on Wireless Networks and Embedded Systems (WECON)*, PP.1-5

Keefer, 2008.*Principles and Practices of Vibrational Analysis*, GEA FES Systems Inc., Vibration Analysis Services, Basics of vibrations, PP. 8-15 [online]
Available at:
https://www.irc.wisc.edu/export.php?ID=158 [Accessed 12 January 2018]

Khemani H., 2009. *Parts of the solenoid valve. How solenoid valve works?* [Online]
Available at:
http://www.brighthubengineering.com/manufacturing-technology/56397-parts-of-the-solenoid-valve-how-solenoid-valve-works/ [Accessed 21 May 2018]

Kim T.H., 2010. SCADA Architecture with Mobile Remote Components, *WSEAS TRANSACTIONS on SYSTEMS and CONTROL*, Issue 8, Volume 5, PP. 611-622

Kohler M., 2016.*Industry 4.0 provides new possibilities for preventive maintenance* [online]
Available at:
https://blog.bosch-si.com/industry40/industry-4-0-provides-new-possibilities-for-preventive-maintenance/ [Accessed 15 January 2018]

Kolberg D. Z. D., 2015. Reference architectures for the internet of things, *IFAC-Papers Online*, vol. 48-3, pp. 1870–1875

Kothari K., Modh N., Prajapati J., & Jakhaniya A., 2015. An autonomous bottle filling machine adding to the accuracy and precision along with rejection mechanism, *International Journal of Advanced Technology in Engineering and Science*, Vol. No3, PP.60-67

Kumar P., 2015. Designing of Home Automation based Switching System in PLC Applications using Phone Accelerometer, *Global Journal of Researches in Engineering: G Industrial Engineering,* Volume 15, Issue 2 Version1.0, Type: Double Blind Peer Reviewed International Research Journal Global Journals Inc. (USA)Online, ISSN: 2249-4596 Print ISSN: 0975-5861, pp.7-17

Langmann R. & Rojas-Pena L.F., 2016. A PLC as an Industry 4.0 component, *13th International Conference on Remote Engineering and Virtual Instrumentation (REV),* PP. 10-15

Lean Manufacturing Times, 2018.*Lean Manufacturing and Industry 4.0* [online]
Available at:
http://www.leanmanufacturingtimes.com/lean-manufacturing-industry-4-0-2/ [Accessed 16 July 2018]

Lee, J., Bagheri, B., Jin, C., 2016. Introduction to cyber manufacturing. *Manufacturing Letters*, 8, PP. 11-15.

Lu, Y., Morris, K. C., Frechette, S., 2016. Current standards landscape for smart manufacturing systems. *National Institute of Standards and Technology*, NISTIR, PP. 8107.

Mainardi E., Bonfe M. &Golfieri M.,2007. Fuzzy Controller for an Air Conveyor Bottling Plant, *IEEE International Conference on Control Applications*, PP. 1504 – 1509

Medida S.,2007. *Industrial Automation for Engineers and Technicians*, IDC Engineering pocket guide, 1st Edition - Pocket Guide on, Rev 1.04, pp. 18-19
Micro Small and Medium Enterprises-Development Institute (MSME), Chemical Division, 2011.*Project Profile on Packaged Drinking Water* [online]
Available at:
www.dcmsme.gov.in/reports/food/packageddrinkingwaterthrissur.pdf [Accessed 17 April 2018]

Mi, M., Zolotov, I., 2016. Comparison between multiclass classifiers and deep learning with focus on industry 4.0. *In 2016 Cybernetics & Informatics (K&I)*, PP. 1-5

Monostori L., 2014.Cyber-physical production systems: Roots, expectations and R&D challenges, *in: Proc. of the 47th CIRP Conference on Manufacturing Systems*, pp. 9–13

Niesen T., Houy C., Fettke P. & Loos P., 2016.Towards an Integrative Big Data Analysis Framework for Data-driven Risk Management in Industry 4.0,*49th Hawaii International Conference on System Sciences*, PP. 5056 – 5074

Niku S., 2011. *Introduction to Robotics: Analysis, Control, Applications*, 2nd edition, John Wiley & Sons, Inc. PP. 280

Ollif H., Liu Y., 2017. Towards Industry 4.0 Utilizing Data-Mining Techniques: a Case Study on Quality Improvement. *Procedia CIRP Volume 63*, PP. 167-172

Park C.Y., Laskey K.B., Salim S., Lee J.Y., 2017. Predictive Situation Awareness Model for Smart Manufacturing, *20th International Conference on Information Fusion*, PP. 1-8.

Payghan S.T., Deshmukh R.H. & Magar P.P., 2016. Automation of Bottle Filling Plant with Industry 4.0, *International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering*, Vol. 5, Issue 3, PP. 1578 – 1583

Pereira R.R, Da Silva V.A.D., Brito J.N. & Nolasco J. D., 2016. On-line Monitoring Induction Motors by Fuzzy Logic - A study for predictive maintenance operators, *12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, PP. 1341 – 1346

Petrasch R. & Hentschke R., 2016. Process Modelling for Industry 4.0 Applications: towards an Industry 4.0 Process Modelling Language and Method, *13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, PP. 1-5

Qureshi W., 2017. *Coca-cola unveils new packaging strategy to double recycle plastic usage* [online]
Available at:
https://www.packagingnews.co.uk/news/materials/flexible-plastics/coca-cola-unveils-new-packaging-strategy-double-recycled-plastic-usage-12-07-2017 [Accessed 12 May 2018]

Roubaud J., 2017. *How predictive maintenance fits into Industry 4.0* [online]
Available at:
https://www.engineering.com/AdvancedManufacturing/ArticleID/15798/How-Predictive-Maintenance-Fits-into-Industry-40.aspx [Accessed 09 February 2018]

Sanders C., 2011.*A Guide to Vibration Analysis and Associated Techniques in Condition Monitoring,* DAK Consulting - Chiltern House, 45 Station Road, Henley on Thames, RG9 1AT [online]
Available at:
http://www.dakacademy.com/newsite/index.php?option=com_k2&Itemid=500&id=94_007cd4b8b347e375bc10dbe5efbccc28&lang=en&task=download&view=item [Accessed 15 January 2018]

Shrouf F., Ordieres J. & Miragliotta G., 2014. Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the internet of Things paradigm, *IEEE International Conference on Industrial Engineering and Engineering Management*, PP.697 – 701

Stankovic J.A. &Rajkumar R., 2004.Real-time operating systems,*Real-Time Systems*, vol. 28, nos. 2–3, pp. 237–253

Technical Editor, 2017. *Block diagram of programmable logic controller (PLC)* [online]
Available at:
http://www.polytechnichub.com/block-diagram-programmable-logic-controller-plc/ [Accessed 18 May 2018]

Tianxia Z., Feng D. & Hao Y., 2012. Application of PLC for Arranging Bottle in Beer Filling Production Line, *24th Chinese Control and Decision Conference (CCDC)*, PP.1215 – 1220

Triada, Engineering and Manufacturing Company, 2013.*Advantages of SIMANTIC S7-1200 controllers SIEMENS* [online]
Available at:
http://triada.com.ua/en/device/179-siemens-simatic_s7-1200_malye_sistemy-advantage.html
[Accessed 19 May 2018]

U.N. Department of Economic and Social Affairs, Population Division, 2018.*World population: Past, Present and Future* [online]
Available at:
http://www.worldometers.info/world-population/ [Accessed 17 March 2018]

Ungurean I., Gaitan N.C. & Gaitan V.G., 2014.An IoT architecture for things from industrial environment, *10th International Conference on Communications (COMM),* PP. 1-4

Upkar S.A., 2014. *Should SMEs automate?*[Online]
Available at:
http://autocomponentsindia.com/should-smes-automate/ [Accessed 13 February 2018]

Upkar S.A., Kumar C.S, Gopal K.P, Parthasathy V.K, & Rathee R.S., 2014. *Should SMEs automate?* [Online]
Available at:
http://autocomponentsindia.com/should-smes-automate/ [Accessed 13 February 2018]

Wan J., Cai H. & Zhou K., 2015. Industry 4.0 Enabling technologies, *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, PP. 135 – 140

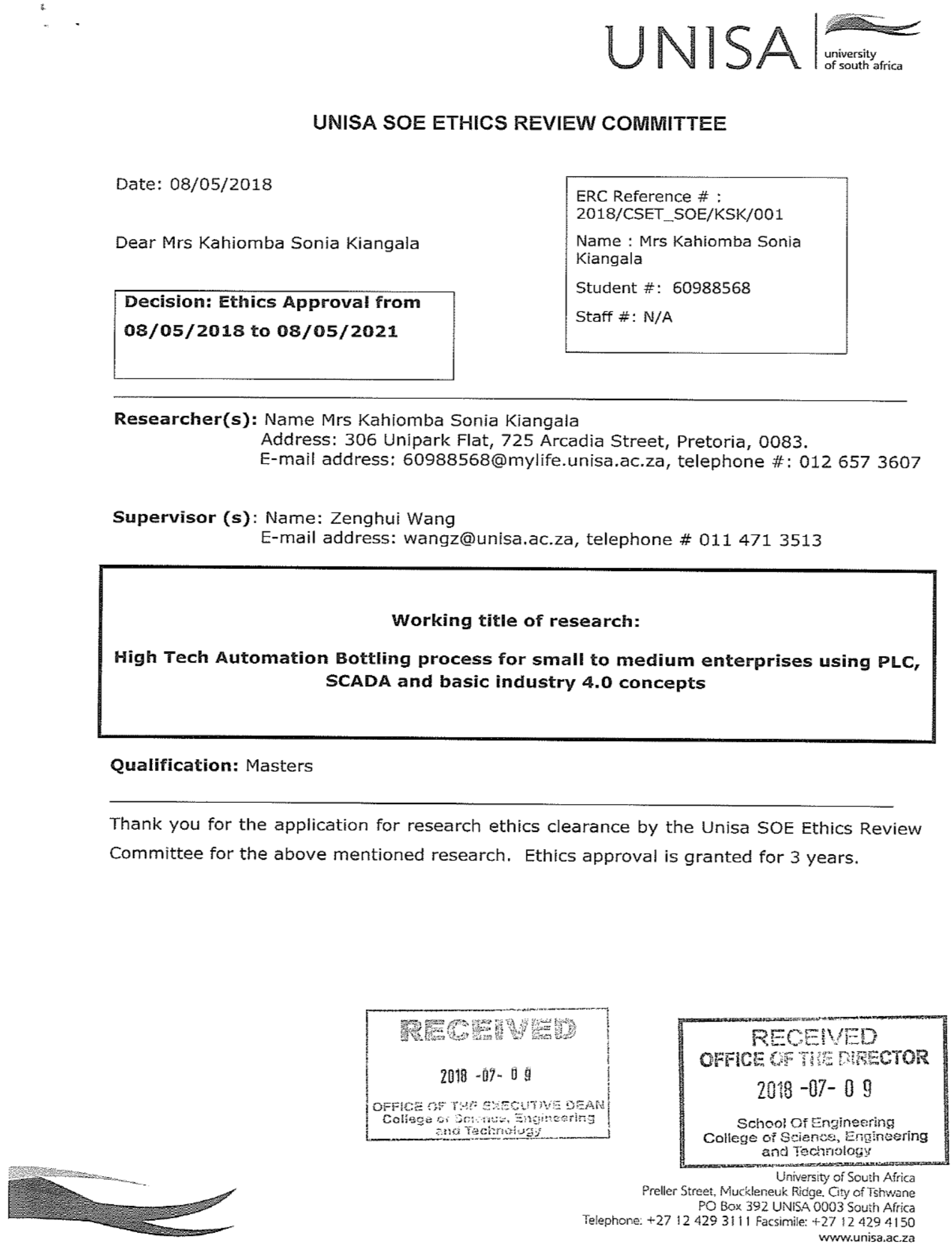
Wan J., Yi M., Li D., Zhang C., Wang S. & Zhou K., 2016. Mobile Services for Customization Manufacturing Systems: An Example of Industry 4.0, *IEEE Journals & Magazines*, Volume 4, PP.8977-8986

Zanella A., Bui N., Castellani A., Vangelista A., & Zorzi L., 2014. Internet of things for smart cities, *IEEE Internet of Things Journal*, PP.22-32

ZVEI, 2015.*Reference architecture model Industrie 4.0 (RAMI4.0)*[Online]
Available at:
https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2016/januar/GMA_Status_Report__Reference_Archtitecture_Model_Industrie_4.0__RAMI_4.0_/GMA-Status-Report-RAMI-40-July-2015.pdf [Accessed 12 January 2018]

9. APPENDICES

9.1 Research ethics approval



UNISA
university of south africa

UNISA SOE ETHICS REVIEW COMMITTEE

Date: 08/05/2018

Dear Mrs Kahiomba Sonia Kiangala

Decision: Ethics Approval from 08/05/2018 to 08/05/2021

ERC Reference # : 2018/CSET_SOE/KSK/001
Name : Mrs Kahiomba Sonia Kiangala
Student #: 60988568
Staff #: N/A

Researcher(s): Name Mrs Kahiomba Sonia Kiangala
Address: 306 Unipark Flat, 725 Arcadia Street, Pretoria, 0083.
E-mail address: 60988568@mylife.unisa.ac.za, telephone #: 012 657 3607

Supervisor (s): Name: Zenghui Wang
E-mail address: wangz@unisa.ac.za, telephone # 011 471 3513

Working title of research:

High Tech Automation Bottling process for small to medium enterprises using PLC, SCADA and basic industry 4.0 concepts

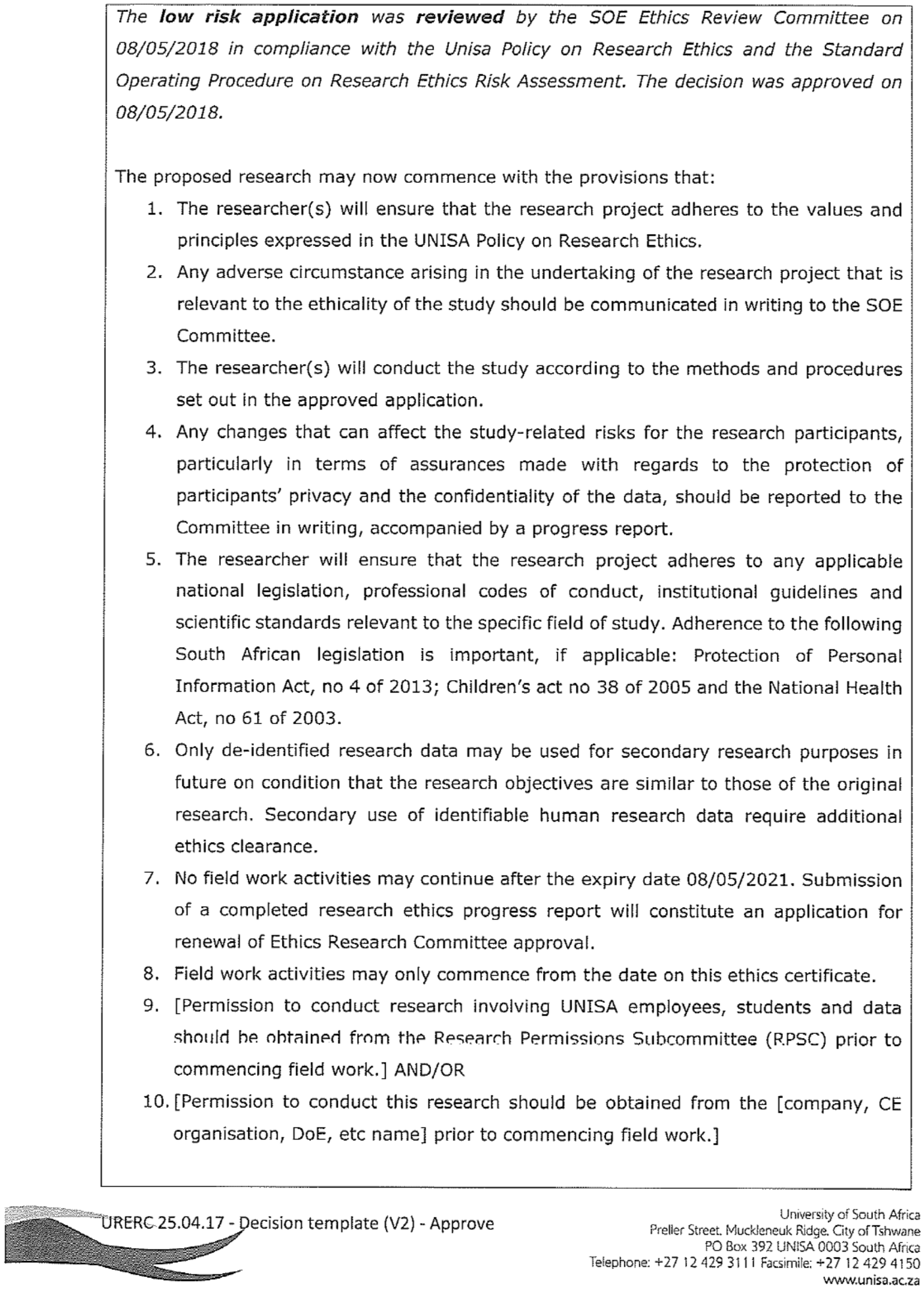
Qualification: Masters

Thank you for the application for research ethics clearance by the Unisa SOE Ethics Review Committee for the above mentioned research. Ethics approval is granted for 3 years.

RECEIVED
2018 -07- 0 9
OFFICE OF THE EXECUTIVE DEAN
College of Science, Engineering and Technology

RECEIVED
OFFICE OF THE DIRECTOR
2018 -07- 0 9
School Of Engineering
College of Science, Engineering and Technology

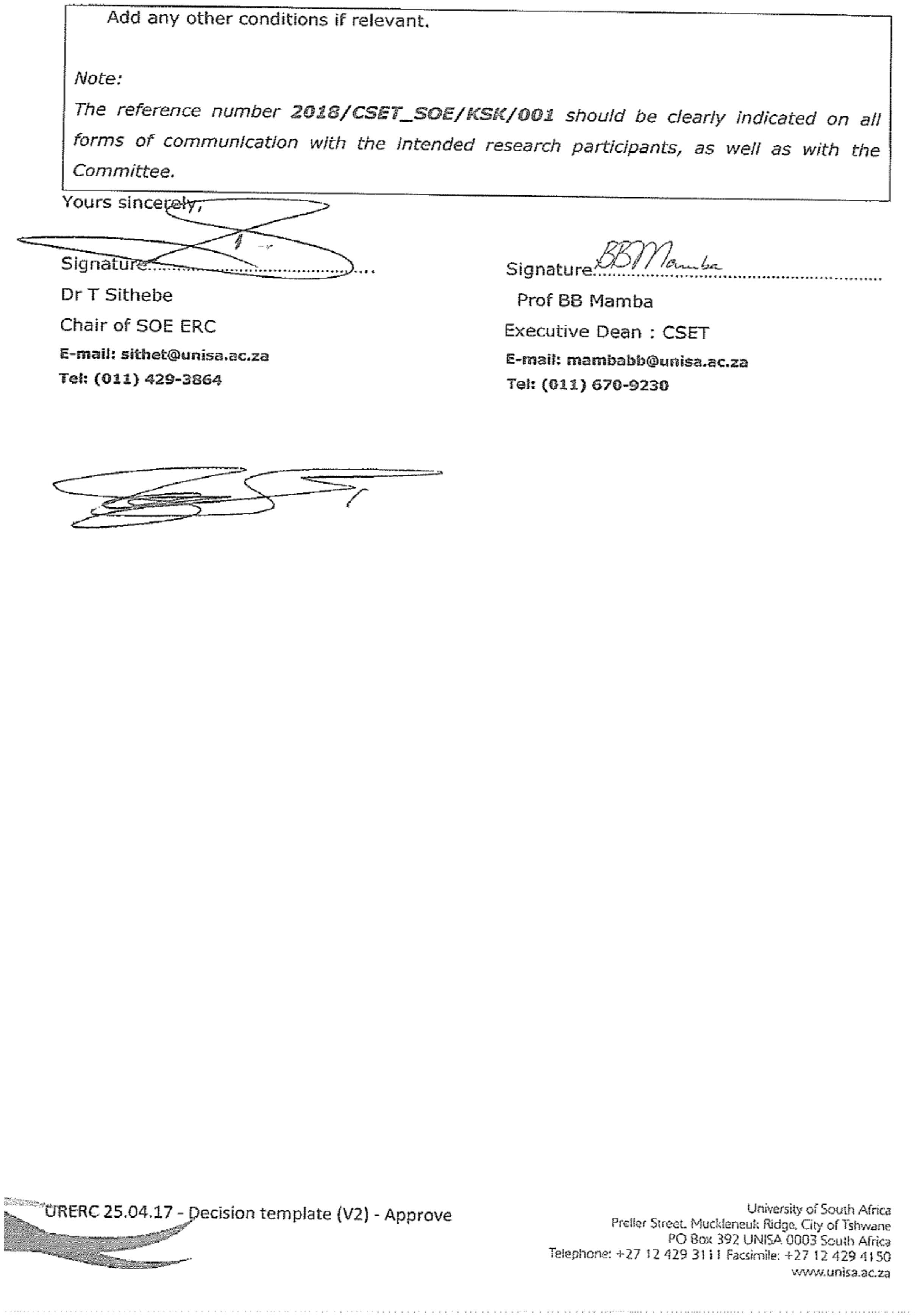
University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za



*The **low risk application** was **reviewed** by the SOE Ethics Review Committee on 08/05/2018 in compliance with the Unisa Policy on Research Ethics and the Standard Operating Procedure on Research Ethics Risk Assessment. The decision was approved on 08/05/2018.*

The proposed research may now commence with the provisions that:

1. The researcher(s) will ensure that the research project adheres to the values and principles expressed in the UNISA Policy on Research Ethics.
2. Any adverse circumstance arising in the undertaking of the research project that is relevant to the ethicality of the study should be communicated in writing to the SOE Committee.
3. The researcher(s) will conduct the study according to the methods and procedures set out in the approved application.
4. Any changes that can affect the study-related risks for the research participants, particularly in terms of assurances made with regards to the protection of participants' privacy and the confidentiality of the data, should be reported to the Committee in writing, accompanied by a progress report.
5. The researcher will ensure that the research project adheres to any applicable national legislation, professional codes of conduct, institutional guidelines and scientific standards relevant to the specific field of study. Adherence to the following South African legislation is important, if applicable: Protection of Personal Information Act, no 4 of 2013; Children's act no 38 of 2005 and the National Health Act, no 61 of 2003.
6. Only de-identified research data may be used for secondary research purposes in future on condition that the research objectives are similar to those of the original research. Secondary use of identifiable human research data require additional ethics clearance.
7. No field work activities may continue after the expiry date 08/05/2021. Submission of a completed research ethics progress report will constitute an application for renewal of Ethics Research Committee approval.
8. Field work activities may only commence from the date on this ethics certificate.
9. [Permission to conduct research involving UNISA employees, students and data should be obtained from the Research Permissions Subcommittee (RPSC) prior to commencing field work.] AND/OR
10. [Permission to conduct this research should be obtained from the [company, CE organisation, DoE, etc name] prior to commencing field work.]



Add any other conditions if relevant.

Note:

*The reference number **2018/CSET_SOE/KSK/001** should be clearly indicated on all forms of communication with the intended research participants, as well as with the Committee.*

Yours sincerely,

Signature...............................	Signature BBMamba...............................
Dr T Sithebe	Prof BB Mamba
Chair of SOE ERC	Executive Dean : CSET
E-mail: sithet@unisa.ac.za	**E-mail: mambabb@unisa.ac.za**
Tel: (011) 429-3864	**Tel: (011) 670-9230**

URERC 25.04.17 - Decision template (V2) - Approve

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

9.2 Weather forecast webpage HTML code

```
<!-- AWP_In_Variable Name='"Weather".Monday' -->
<!-- AWP_In_Variable Name='"Weather".Tuesday' -->
<!-- AWP_In_Variable Name='"Weather".Wednesday' -->
<!-- AWP_In_Variable Name='"Weather".Thursday' -->
<!-- AWP_In_Variable Name='"Weather".Friday' -->
<!-- AWP_In_Variable Name='"Weather".Saturday' -->

<!doctype>
<htmllang=en>
<head>
<metacharset=utf-8>
<title>Weekly Weather Forecast for Beverage Plant </title>
</head>
<body>
<h1>Weekly Weather Forecast for Beverage Plant </h1>
<h3>Weekly weather Forecast are saved here to the beverage PLC:</h3>
<ul>
<em>
<li>Monday :="Weather".Monday: *C
                <formmethod="post">
<input name='"Weather".Monday' type="text" />
<button type="submit">Save</button>
</form>
                </li>
                <li>Tuesday :="Weather".Tuesday: *C
                <formmethod="post">
<input name='"Weather".Tuesday' type="text" />
<button type="submit">Save</button>
</form>
                </li>
                <li>Wednesday :="Weather".Wednesday: *C
                <form method="post">
<input name='"Weather".Wednesday' type="text" />
<button type="submit">Save</button>
</form>
                </li>
                <li>Thursday :="Weather".Thursday: *C
                <form method="post">
<input name='"Weather".Thursday' type="text" />
<button type="submit">Save</button>
</form>
                </li>
                <li>Friday :="Weather".Friday: *C
```

```
                              <form method="post">
<input name="'Weather".Friday' type="text" />
<button type="submit">Save</button>
</form>
                              </li>
                              <li>Saturday :="Weather".Saturday: *C
                              <form method="post">
<input name="'Weather".Saturday' type="text" />
<button type="submit">Save</button>
</form>
                              </li>

</em>
</ul>

<map name="primary">
<area shape="circle" coords="200,250,25" href="another.htm" />
<area shape="default" nohref/>
</map>

</body>

</html>
```

9.3 Equipment order webpage HTML code

```
<!doctype>
<html lang=en>
<head>
<metacharset=utf-8>
<title>Weekly Equipment Order for Experimental Beverage Plant </title>
</head>
<body>
<h1> Weekly Equipment Order for Experimental Beverage Plant </h1>
<h3>Please supply the following equipment for Beverage plant for this week:</h3>
<ul>
<em>
<li>Amount of Bottles :="Equipment".bottles:</li>
                              <li>Amount of Bottles Caps  :="Equipment".caps:</li>
                              <li>Amount of labels :="Equipment".labels: </li>
                              <li>Amount of Ingredients :="Equipment".ingredients: Kg </li>
</em>
</ul>
```

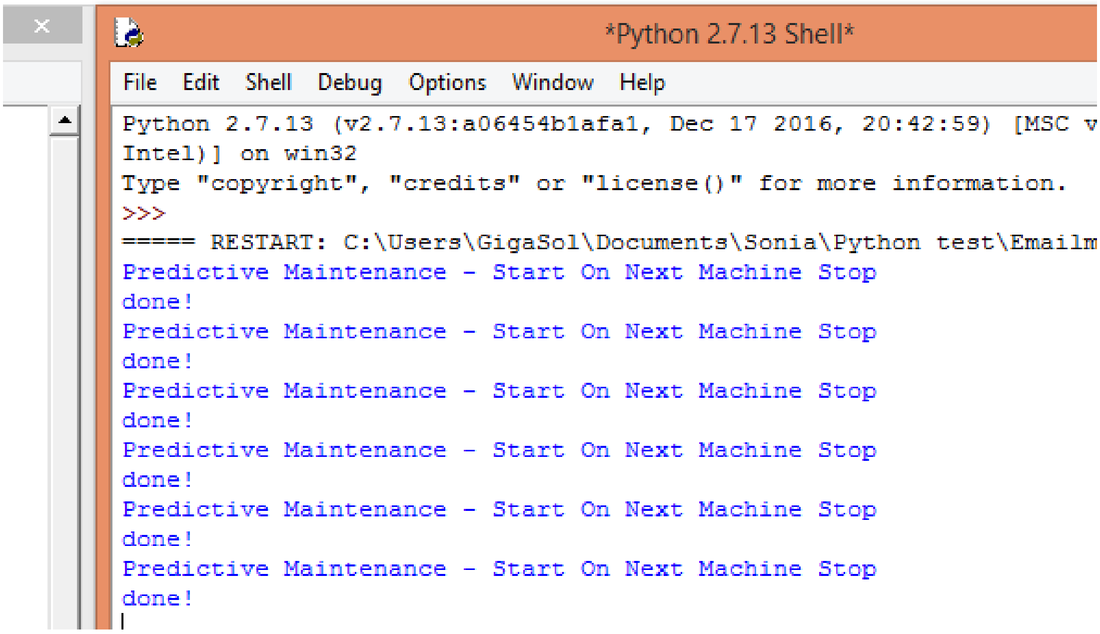
```
<h3>Thanks for your Services!!</h3>

<map name="primary">
<area shape="circle" coords="200,250,25" href="another.htm" />
<area shape="default" nohref/>
</map>

</body>

</html>=
```

9.4 Python code for E-mail notification to Maintenance Supervisor

```
import time
import csv
importsmtplib
fromemail.MIMEMultipart importMIMEMultipart
fromemail.MIMEText import MIMEText
fromemail.MIMEBaseimport MIMEBase
from email import encoders


while(1):
time.sleep(20)
csv_file = open('Test.csv')
csv_reader = csv.reader(csv_file, delimiter= ',')
next (csv_reader)

for row incsv_reader:

if row[3]== "stop":
print ("Reactive Maintenance - Immediate Stop")
fromaddr="bottlingplant001@gmail.com"
toaddr="supervisorbottlingplant001@gmail.com"
msg=MIMEMultipart()
msg['From']=fromaddr
msg['To']=toaddr
msg['Subject']="Bottling plant 001 - Predictive Maintenance Scheduled"

body='''Conveyor Motor K1 - Motor Vibration speed > 7.1mm/s!!!
    Please stop operations as soon as possible for reactive maintenance
ofengine!Urgent!!!'''
msg.attach(MIMEText(body, 'plain'))
```

```
server = smtplib.SMTP('smtp.gmail.com', 587)
server.starttls()
server.login(fromaddr,"SoniA0664#")
text=msg.as_string()
server.sendmail(fromaddr,toaddr,text)
print "done!"
server.quit()

elif row[3]== "start":
print ("Predictive Maintenance - Start On Next Machine Stop")
fromaddr="bottlingplant001@gmail.com"
toaddr="supervisorbottlingplant001@gmail.com"
msg=MIMEMultipart()
msg['From']=fromaddr
msg['To']=toaddr
msg['Subject']="Bottling plant 001 - Predictive Maintenance Scheduled"

body='''Conveyor Motor K1 - Motor Vibration speed is in the satisfactory range!!!
    Please perform maintenance on next machine stop to reduce changes of failure on
engine!Urgent!!!'''
msg.attach(MIMEText(body, 'plain'))


server = smtplib.SMTP('smtp.gmail.com', 587)
server.starttls()
server.login(fromaddr,"SoniA0664#")
text=msg.as_string()
server.sendmail(fromaddr,toaddr,text)
print"done!"
server.quit()

else:
print(" Normal operation ")

csv_file.close()
```

9.5 Python script running on compiler for predictive maintenance



```
Python 2.7.13 (v2.7.13:a06454b1afa1, Dec 17 2016, 20:42:59) [MSC v
Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\GigaSol\Documents\Sonia\Python test\Emailm
Predictive Maintenance - Start On Next Machine Stop
done!
Predictive Maintenance - Start On Next Machine Stop
done!
Predictive Maintenance - Start On Next Machine Stop
done!
Predictive Maintenance - Start On Next Machine Stop
done!
Predictive Maintenance - Start On Next Machine Stop
done!
Predictive Maintenance - Start On Next Machine Stop
done!
```