Jussi-Pekka Nikko

Safety PLCs – Competitor analysis of software usability

Usability investigation of safety PLC related software

Thesis
Fall 2019
School of Technology
Automation Engineering

SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Thesis abstract

Faculty: School of Technology

Degree Programme: Automation Engineering

Specialization: Electrical Automation

Author: Jussi-Pekka Nikko

Title of thesis: Safety PLCs – Competitor analysis of software usability

Supervisor: Jarkko Pakkanen

Year: 2019 Number of pages: 50

To achieve better results in a specific market, having an understanding of current and possible new competitors is valuable information for companies predicting the possible changes in the market and creating successful sales strategies. Analyzing the market and performing tests on competitor's products is a common practice in any industry. In the thesis, three competitors of Schneider Electric were analyzed.

Market analysis was introduced, focusing on the Porter's five forces -analysis and usability testing. The current machine safety -related standards were explained, and a glance at Schneider Electric's portfolio was taken.

The competitor analysis was conducted using a comparative analysis form created in an earlier thesis for Schneider Electric. One competitor was analyzed on an overall level by ordering physical equipment from the competitor, and by programming a safety function task. Two other competitors were analyzed only on a software level, meaning that the software was ordered, installed, and its usability was analyzed to a point which was plausible without the physical equipment. As the result of these analyzes, documents were created for the use of Schneider Electric's research & development department.

Keywords: competitor, analysis, safety

SEINÄJOEN AMMATTIKORKEAKOULU

Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikan yksikkö

Tutkinto-ohjelma: Automaatiotekniikka

Suuntautumisvaihtoehto: Sähköautomaatio

Tekijä: Jussi-Pekka Nikko

Työn nimi: Ohjelmoitavat turvalogiikat – Kilpailijoiden ohjelmistojen käytettävyyden analysointi

Ohjaaja: Jarkko Pakkanen

Vuosi: 2019 Sivumäärä: 50

Markkina-analyysi on yksi yritysten paremman markkina-aseman tavoittelussa käytetyistä työkaluista. Kilpailijoiden tuotteiden tutkiminen ja testaaminen antavat yritykselle tietoa heidän kilpailijoidensa nykyisestä kilpailukyvystä. Tässä opinnäytetyössä analysoitiin kolmea Schneider Electricin kilpailijaa.

Opinnäytetyö esitteli markkinatutkimuksen työkaluina Porterin Viisi kilpailuvoimaa mallin ja käytettävyystestauksen sekä koneturvallisuutta koskevia standardeja. Lisäksi työssä perehdyttiin Schneider Electricin ohjelmoitavaan turvalogiikaan ja sen ohjelmoinnissa käytettävään ohjelmistoon.

Kilpailijoita analysoitiin tuotevertailuarviointiin perustuvalla arviointipohjalla. Yhdeltä kilpailijalta tilattiin tarvittavat komponentit ja ohjelmistot, jonka jälkeen luotiin yksinkertainen turvaohjelma. Kahdelta muulta kilpailijalta tilattiin ainoastaan ohjelmistot, jotka analysoitiin niin, että fyysisiä komponentteja ei tarvittu. Näiden analyysien pohjalta luotiin dokumentit kilpailijoista Schneider Electricin tuotekehityksen käyttöön.

Asiasanat: kilpailija-analyysi, koneturvallisuus, turvaohjelmointi

TABLE OF CONTENTS

Tables, Figures and Pictures

Terms and Abbreviations

IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
SFS	Finnish Standards Association
PLC	Programmable Logic Controller
SLC	Safety Logic Controller
SS1	Safe Stop 1
ST	Structured Text
SCP	Structure-Conduct-Performance -paradigm
PEST	Politics, Economics, Social culture, and Technology Framework
R&D	Research and development
SIL	Safety Integrity Level
I/O	Input / Output
IL	Instruction List
SFC	Sequential Function Chart
LD	Ladder Diagram
FBD	Function Block Diagram

1 INTRODUCTION

As factories are becoming bigger and machines getting more complex, the number of risks and hazards is increasing. The possibility of causing injuries to an operator, bystanders, or machinery is to be minimized. One way of reducing risk is to design the machine to have fewer moving parts. Most of the time, having fewer moving parts cannot resolve all safety issues and still maintain the functionality of the machine. Thus a different approach to risk reduction is required. Safety logic controllers are up to the task, and there is a competitive market around them (Markets and markets 2018.) Industrial automation companies employ competitor analyses and usability testing methods to help to create products that stand out because of their functionality and serve the customer demands better.

The thesis introduces the Porter’s Five forces -competitor analysis and the theory about usability testing. In addition to the theory about analyzing competitors, the safety logic controllers, and the programming software are introduced along with the regarding standards.

To gain knowledge of the current state of the competitors’ devices and software, the related components were ordered from one competitor for further analysis. As for the two other competitors, the programming software was evaluated on a level, which could be achieved without having the physical equipment. The application created in the analysis template was a basic safety software example, which could be produced with every competitor’s devices.

Schneider Electric Group is an international company, which employs over 137000 employees on all continents and has a revenue of 25.7 billion euros. Asia Pacific, North America, and Western Europe are the biggest markets, which take up 84% of Schneider’s global footprint. Overall, Schneider Electric offers its customers industrial automation and energy management solutions. (Schneider Electric. 2019.)

Schneider Electric Automation GmbH in Marktheidenfeld in Germany is the international headquarters for the Machine solutions and System Consistency divisions (Schneider Electric A. [Referred 04.04.2019].) The office in Marktheidenfeld has almost 500 employees from 26 different nations and multiple different fields of knowledge, completing software and hardware projects, training, and sales (Schneider Electric B. [Referred 04.04.2019].) The thesis is created for the use of Schneider Electric’s research & development.

2 MARKET AND COMPETITOR ANALYSES

The most widely used business strategy development tools are strengths, weaknesses, opportunities and threats analysis, in short SWOT, the industry structure analysis called Porter’s five forces, value chain analyses and generic and group strategy analyses. These methods were developed by economists and then simplified to an easier form to be used by managers of companies. (Prahalad 1999.)

Most of the methods were created during the 1980s, while the competitive conditions evolved, and the market was well-understood. Japanese manufacturing and production market started flourishing and shook the markets in the United States and Europe. (Prahalad 1999.)

2.1 Macro- and microeconomics

Macroeconomic analysis or the PEST analysis of a market gives a company insight into political, economic, socio-cultural, and technological fields and factors regarding the external environment. Pressure and legislation from the political field can influence the profitability of the industry. Environmental legislation, tax policies, laws, and trade restrictions are vital factors when analyzing political factors of a company. (Trigeorgis 2011, 57.)

Economics always plays a part in how an industry is formed and what is its growth potential. Factors related to the global- or country-level economy are comprised of the level of unemployment, differences in foreign versus domestic currencies, inflation, and economic growth. Financial crises and otherwise poor economy reflect the weak buying power of customers. An enterprise can expand its operation to foreign markets and take advantage of the local legislation to reduce the economic impacts of a fluctuating market. As an example, the foreign market can use a different currency, protecting the company from changes in currency’s value. (Trigeorgis 2011, 57.)

Consumer needs and buying behaviors play significant roles in socio-cultural factors on a macroeconomic scale. A company should carefully follow how the trends change during different fluctuations in the market. Multinational companies face different cultures in each different location upon expanding to foreign countries. Technological changes press companies to change with the market and take advantage of emerging innovations and technologies. If a company does not change with the market, it may face difficulties when compared to its competitors. (Trigeorgis 2011, 57.)

A structure-conduct-performance paradigm or SCP is an industry analysis method that assumes that the market structure determines a company's conduct and therefore determines industry and company performance in that market. Market structures vary in the amount of competition. A purely competitive market would be possible if all the companies in the market produced identical products and if there were no entry or exit barriers. Companies would base the price of their product on the price of resources and take a fixed profit. A monopolized market, on the other hand, offers no competition and the company in the monopoly position can use its market power to set the price significantly higher, gaining more profit. (Trigeorgis 2011, 58.)

A company's conduct includes such things as product or service pricing, merger and acquisition activity, and innovation. Companies can use these techniques to increase their market power and to gain a better position in the market. A company's decisions on the subjects mentioned above are reflected by the performance in the industry. Financial performance, the strength of the brand, and company growth are some of the metrics that can be analyzed, and used to create a picture of the market. (Trigeorgis 2011, 59.)

2.2 Porter's five forces

Porter's competitor analysis focuses on forces that shape and modify any given industry. As shown in figure 1, it consists of two major threats, two major bargaining powers and the rivalry between the existing companies in the industry. The threat of substitute goods and new entrants to the market pose a threat to the income of companies. The bargaining power of customers and suppliers can drive the prices down, reducing profit. Rivalry in the market driven by competition causes prices to drop; therefore, companies earn average profit instead of higher profit. (Marburger 2012, 60.)

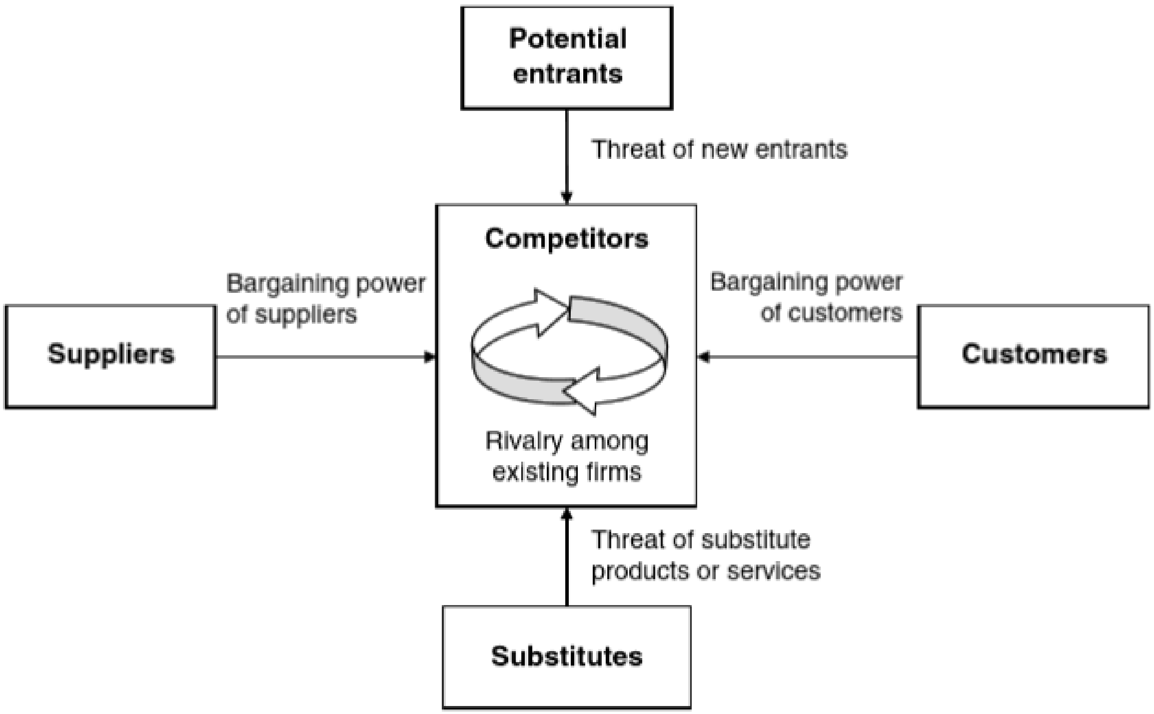


Figure 1. Porter's five forces (Trigeorgis 2011, 60).

2.2.1 The threat of substitute goods

Substitute goods are present in any industry and pose a threat to all companies in it. A rival product might arise from a different industry altogether, which makes the process of detecting potential competitors harder. Substitute goods place a limit on potential profits with the threat of customers switching away from a product. A product is considered a substitute when it performs the same function for the customer as the existing products. Deterring substitute products can be achieved by advertising the existing products. A collective advertisement by all industry participants strengthens their position in the market. (Porter 1998, 23-24.)

2.2.2 The threat of new entrants and market barriers

Companies which enter the industry as new competitors press the existing companies to give up market share and lower prices. Entering a new market can be hindered by the presence of market entry barriers. Market entry barriers are obstacles which can deter potential competitors from entering the market. Most impactful market barriers are

- supply-side economies of scale
- demand-side benefits of scale
- customer switching costs
- capital requirements
- advantages of established companies independent of scale
- access to distribution channels
- governmental policies.(Porter 2008.)

Supply-side economies of scale is a term used for a market barrier where the price of starting a company is high, due to the existing companies in the market. The existing companies may harness better deals from suppliers and have more efficient manufacturing processes which allow the company to divide fixed costs to higher number of products , allowing cheaper manufacturing costs per product, thus lowering the end price of a product. Newcomers to the industry are forced to invest heavily in production to lower their manufacturing costs or to suffer negative economic profits. (Porter 2008.)

Demand-side benefits of scale, on the other hand, act as a market barrier when an industry has networked. Networked industries grow more beneficial to the existing customers with each new customer. Entering a networked industry requires new companies to offer lower prices in order to attract customers. (Porter 2008.)

Customer switching costs apply pressure on the supplier side through the buyer side of an industry. The existing companies on the supplier side may have developed their products and systems in a unique way, forcing the buyer side to continue using the same products. Switching costs to the buyer side can be economic. Such as employee training, acquisition of new equipment, and qualifying a new source of raw material or even the psychological cost of losing a profitable business relationship can be mentioned as examples. (Porter 1998, 10.)

Capital requirements faced when entering an industry form up from the fixed costs of starting a company and unrecoverable costs such as R&D and pre-advertising. Finding the capital can be difficult depending on the industry. Companies in high-value industries have a better chance of finding investors because of the potential revenue and high resale value of the equipment if a new company was to fail. (Porter 1998, 2008.)

Already established companies in an industry may have locked up the most favorable access to raw materials, patents to their name and advantageous locations to bring their products to the consumers, which gives a competitive edge when compared to the entering rivals. Such advantages are called advantages of established companies independent of scale. (Porter 1998, 11-12.)

Distribution channels may already be filled with existing products, which forms a market barrier for new companies. Shelf space and wholesale transportation rights may be hard to acquire. Entrants to the market can offer promotions and strong selling efforts to secure a piece of the mentioned channels. (Porter 1998, 10-11.)

Governmental policies can either strengthen or weaken market entry barriers. Regulations and licensing requirements raise the difficulty of entering, whereas government-funded research provides every company with valuable information. (Porter 1998, 2008.)

Since market growth is slow, new companies can mainly gain market share by taking it from existing companies. By evaluating these entry barriers, entering companies determine if a market share is possible to attain. The established companies in the market may retaliate against entry attempts by lowering prices, making the profit margin for entering companies smaller, over saturating the market with their own products, and by negotiating better deals with distribution channels. (Porter 2008.)

2.2.3 Bargaining power of suppliers

Pressure from suppliers reduces the profit margin for companies operating in the industry. Leverage can be obtained by market integrity, market diversification, strategic locations, product differentiation, and forward integration. While the suppliers' market is less fragmented than the buyers' market, the suppliers have more buyers, which in return, allows them to set prices higher. A supplier with a broad market range can gain revenue from several different markets. Switching costs may be caused to the buyers by location and product differentiation. If a buyer wants to change the supplier, it must invest in new equipment or training. The suppliers can threaten to enter the market themselves if the current companies make too much money relative to the suppliers' profit. (Porter 1998, 2008.)

2.2.4 Bargaining power of customers

In relation to the bargaining power of suppliers, the buyers have their own advantage in the market. Buying companies which operate in a more consistent field than their sellers give them leverage in driving prices down. Companies that buy large volumes in a particular market can make the sellers compete against each other in hopes of better profit. Standardized products drop down the switching costs if the company can get the same product form a different manufacturer. As with the suppliers entering the market, the buyers may enter the supplier market by producing the needed parts themselves. (Porter 1998, 2008.)

Price sensitivity is a characteristic of a buyer, which determines the importance of cost in purchase decision making. A buyer with high price sensitivity implies that the bought item or material is a significant part of production costs, the profit margin is low, the bought product does not affect profoundly to the buyer's product quality or the bought commodity has little effect on the other costs of a company. Being less dependent on sellers' products and having a higher-value customer base lower the buyer's price sensitivity. (Porter 2008.)

2.2.5 The rivalry of existing competitors

The rivalry in a market originates from the lack of a market leader, slowness of growth, exit barriers, and business commitment. While the companies in an industry are several or equal in size, convincing customers to switch is inevitable, causing tension in all companies. If the industry growth is slow, new market share is not created fast enough. Just as entry barriers deter entry to a market, exit barriers prevent negative profit companies from leaving the market. Exit barriers are caused by:

- very specialized products which are needed in another industry
- managements’ psychological need to stay in the industry.

Companies that cannot leave the industry stay lingering in it, claiming market share from more successful companies. Business commitment may be caused by individuals’ ego or political reasons. As an example, state or country-owned enterprises may have a political agenda for employment or regional economic stability. (Porter 2008.)

The most common cause of rivalry is price competition, as it is a quick reaction to fluctuations in the market. Price cuts are harmful to the companies in the industry due to moving profit directly to customers instead of keeping it between the companies. If the sold product is a commodity or very generic, the customers face no switching costs and encourage the competitors to cut prices in order to poach customers. Companies try selling perishable products away with lower prices to avoid economic losses that can be avoided. (Porter 2018.)

2.3 Usability testing

A competitor analysis, such as Porter’s five forces is a comprehensive market analysis consisting of multiple different parts. One aspect of having a more significant market share is distinguishing oneself from the competitors by having better usability. By definition, usability can be described by the following terms:

- usefulness
- efficiency

- effectiveness
- satisfaction
- accessibility. (Rubin & Chisnell 2008, 21.)

Usefulness and efficiency concern the user’s ability to reach a set goal. The easier it is for the user to create what was intended, the more effective the product overall is. Learnability is incorporated into effectiveness and describes how easily the competence to use the product properly is achieved. Satisfaction gives insight into the user’s feelings towards the product and the process of using it, and finally, accessibility gives the product the scope of being able to accumulate knowledge. (Rubin & Chisnell 2008, 21.)

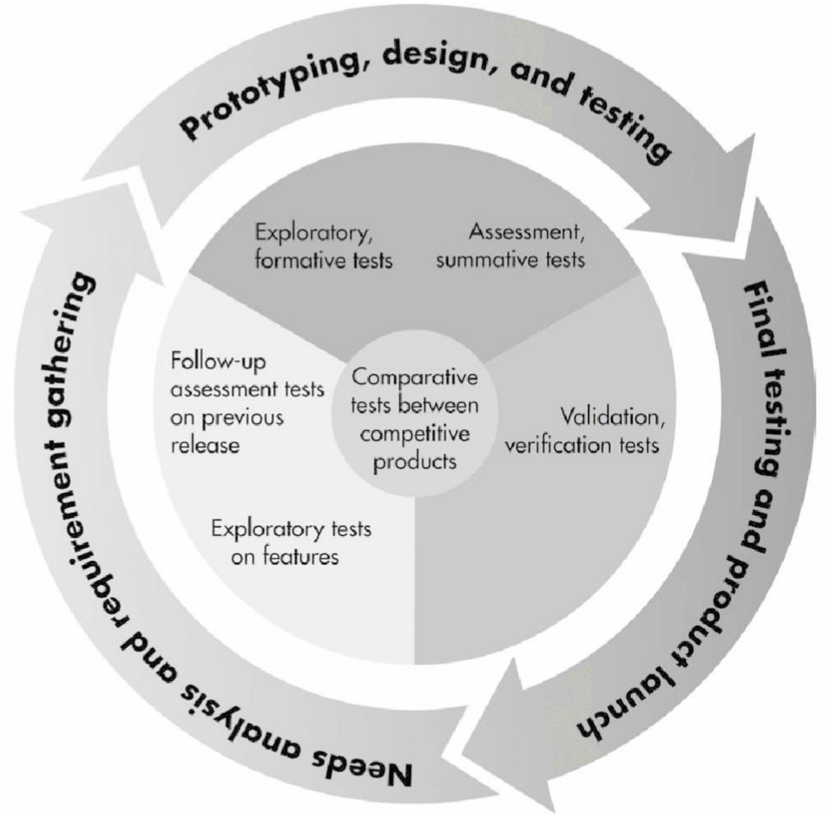


Figure 2. Usability testing in the product lifecycle (Rubin & Chisnell 2008, 28).

Usability testing offers a broad scope to the testing of a product, as shown in figure 2. Whether it would be during the development of a product or after launching and comparing it to the competitor’s products. Exploratory assessment and validation tests during the different phases of a product’s life cycle can be conducted to gain advantage and to accumulate insight to improve the product. A comparison test can be run at any point of the life cycle independently or as a part of product life cycle tests. (Rubin & Chisnell 2008, 27.)

2.3.1 Exploratory test

Exploratory usability tests are to be conducted during the prototyping and designing phases when the user and usage profiles are defined, and possibly some prototypes have already been produced. The objective of these early usability tests is to receive valuable information early in the development process to avoid developing complicated products or software that could prove to be challenging to modify. This type of test is usually run within the company producing the product, and the counterparts are the other versions of the same product. (Rubin & Chisnell 2008, 29-30.)

2.3.2 Assessment and Validation tests

As the production of a product advances from the prototype phase, an assessment test can be run to test the product on a higher level of detail. More features are implemented and tested than before. The assessment test is a complementary test to the exploratory, testing if the implemented features work as intended. Validation tests are performed close to the launch of a product to check if the issues found earlier have been solved. The product is usually tested against an internal or a competitor's standard for the product type. (Rubin & Chisnell 2008, 34-36.)

2.3.3 Comparison test

A comparison can be made at any stage of the product lifecycle. As the product is being developed, it can be compared to different versions of itself. During the development, before the launch, the different versions of the product are to be compared again but on a deeper level. Decisions on using visual components as opposed to textual can be evaluated. When the product is nearing its launch or even after the product has been published, it can be compared to the competitor's product. Traditionally the comparison test is run with a control group performing a task, and another group performing a similar task with different equipment. If statistically valid information is wanted, the task should vary only in one aspect at a time. (Rubin & Chisnell 2008, 37-38.)

3 SAFETY

Creating a safe working environment is one of the most crucial parts of having a productive and enjoyable workplace. Safety increases the overall productivity of a machine by reducing the downtime between errors. When a machine stops operation in a safe manner, the risk of damaging crucial parts is lower. (VTT 2004). Within this chapter, a look at the current safety standards regarding the safety-related devices and software is taken.

3.1 Functional safety

Safety of an automated machine or production line depends more on control systems as they get faster, comprehensive, and complex. Errors and faults in these control systems can lead to loss of safety functions and unexpected start-ups, thus causing danger and harm to the operator. Since humans rely more on these systems to keep them safe from harm, which lowers the personal focus on safety, the systems must be designed in a way they do not cause harm in an emergency. (VTT 2004.)

According to the IEC 61508 standard safety is defined as follows:

> Freedom from unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly as a result of damage to property or the environment. (IEC/TR 61508-0 2012, 9.)

Functional safety is having active components as a part of a system to protect it and people around from hazards. An active component is usually an electronic sensor which keeps track of hazards. As an example, a thermal sensor detecting overheating and cutting the power to the device before that overheating occurs is functional safety. Passive methods such as insulating or preventing the same hazard are not considered functional, rather as safety through design. Safety function and integrity requirements are necessary to achieve proper functional safety, and they are derived from hazard analysis and risk assessment of the system. (IEC/TR 61508-0 2012.)

A machine with hazardous parts or areas protected by physical barriers that require entry through a door or a hatch usually has a safety circuit built into them. Ensuring the functionality and safety of the safety circuit must be proven by risk assessment and hazard analysis. The hazard analysis is used to identify hazardous scenarios such as the machine not stopping fast enough. Performing a risk assessment to the system gives the requirements for the safety function where minimal risk is exposed to the user and environment. Factors affecting the safety integrity are risks emerging from a machine, which range from bruises to loss of limb to fatal accident, and frequency of accesses to the hazardous area. (IEC/TR 61508-0 2012, 17.)

Safety-related methods can be divided into three categories: prevention, automation, and design. By preventing access to dangerous areas of a machine or production line, the operator can be secured from harm. Traditionally prevention is done using gates and locks while the machine is working. (VTT 2004.)

Using automation in machine safety increases the response time to events and allows the whole system to communicate with different parts. This communication improves the overall safety of the system. The responsibility is thus moved from humans to the automated system. Safety through design is the third option when designing safe machines. It takes responsibility away from the automated control systems by adding common sense instead of more hardware. As an example, a pressurized pipe can have its wall thickness reduced to create a reliable pressure control system. (VTT 2004.)

3.2 Safety standards

Safety standards are divided into three different types; type A, B, and C. These types aim to make standardization balanced and straight forward. Type A standards are defined as general standards that can be applied to all machines. These standards are applied if more specific standards are not available and define a general level of safety which other standards must follow. EN ISO 12100:2010 Safety of machinery standard, and EN ISO 14121:2013 Safety of machinery – Risk assessment are type A standards. (Siirilä 2008, 59.)

Type B can be divided into two different branches, B1 and B2. Type B1 standards provide a framework for designing safe machines regarding particular safety aspects such as safe distances, surface temperatures, and the noise levels of a machine. Whereas B1 defines the machine, devices such as pressure sensors, two-hand controllers, and interlocking devices on board the machine are defined in type B2 standards. (Schneider Electric 2012.)

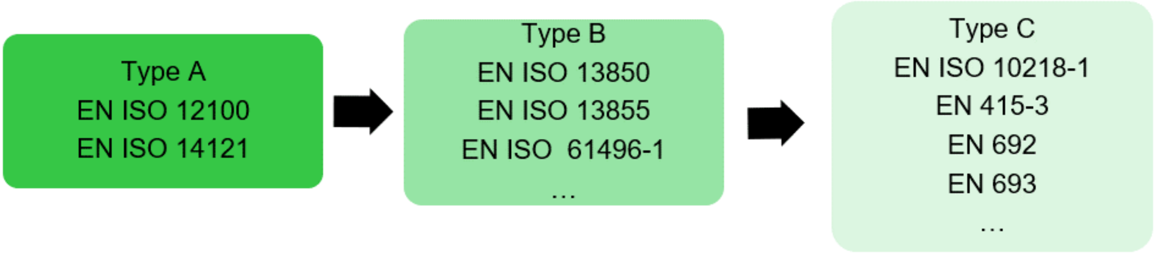


Figure 3. Safety standard types (applying Schneider Electric 2012).

Type C standards are specifying standards that are used with type A and B standards for specific machines or machine groups such as conveyors, lathes, and robot cells. Standards do not repeat the contents in lower levels therefor types A and B must be used to complement type C standards. (Siirilä 2008, 61.)

3.3 Performance level

When designing new machines and their required safety measures, performance levels are used to determine the total safety level. First, a risk assessment is made after which the required performance level or PL_r for every safety function in the control system must be determined. PL_r has three main questions which are asked and answered by the machine designers:

- the severity of injury,
- frequency and exposure of dangerous events
- the probability of avoiding them.

All variables are split into two different levels, as shown in figure 4. (SFS-EN ISO 13849-1 2015, 53.)

The severity of injury is split between reversible (S1) and irreversible (S2) injuries. Reversible injuries include bruising and small cuts. Irreversible injuries would be loss of limbs or loss of life. When determining between S1 and S2, the consequences and healing processes of injuries are to be noted. (SFS-EN ISO 13849-1 2015, 53.)

Exposure and frequency are added to the equation as seldom (F1) and frequent (F2). It makes no difference in terms of the frequency if the person subjected to the hazard is the same or not. Every time the hazard occurs in the machine, it is counted towards the frequency factor. Exposure to the hazard should be evaluated by the average exposure time relative to the life span of the machine. (SFS-EN ISO 13849-1 2015, 54.)

Avoiding hazards and the probability of their occurrence is evaluated in the parameters P1 and P2. Most of the machines are rated as P2 because of their unexpected behavior in hazardous situations. Only if the hazard can be very easily recognized or its risk substantially reduced, the machine may be rated as P1. (SFS-EN ISO 13849-1 2015, 54.)

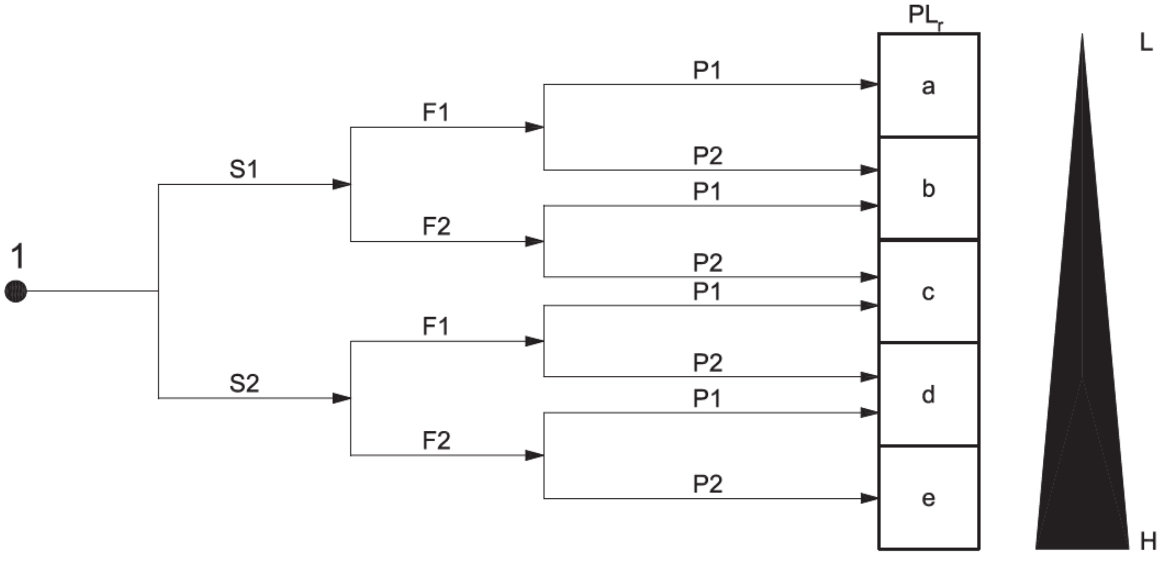


Figure 4. Performance Level requirement graph (SFS-EN ISO 13849-1 2015, 55).

The performance level (PL) of the safety system is determined by architectural categories (categories), Mean Time to Dangerous Failure ($MTTF_D$), Diagnostic Coverage (DC) and Common Cause Failure (CCF). Calculations of these values and the percentages for the proper performance level can be made manually, but mainly an automated software such as Sistema is used. The architectural categories are defined in ISO 13849-1:2015 standard and are used to set requirements to systems in an understandable way. The categories range from 1 to 4 and also include category B. The categories get more specific and safer as the number grows. (Nix 2017.)

Terms *well-tried components* and *well-tried safety principles* are used in categories 1 to 4, and they are introduced in ISO 13849-2 as well-tried safety principles. Screws, springs, cams, and break-pins are all fundamental parts of machines with a long and successful history of usage, and they are made and proofed by using methods which show their suitability and reliability for safety applications. As with components, well-tried safety principles have been tested and used in several different machines and are proven to be functional. As an example, increased off-force is a method where the safe mechanical state is obtained through higher off than on force. (Nix 2010.)

Category B is the lowest single-channel category containing a requirement that used circuits are rated to a correct voltage, and other specifications are followed. Category 1 must meet the requirements of the category B and also must use well-tried safety principles. Category 2 is the highest single-channel category and it must meet the requirements of the Category 1, and include the testing of the safety function by the machine control system. Category 3 requires dual channels for more secure safety testing. It must fulfill the Category 2 requirements and be able to withstand single faults without losing the safety function. Category 4 is a dual-channel category and it is the highest of the categories fulfilling the Category 3 requirements, and in addition, it must detect faults at or before the next demand of the safety function. If detecting faults is not possible within the duty cycle, a load of undetected faults may not cause the loss of a safety function. These categories are shown in an illustrative way in figure 6. (Nix 2017.)

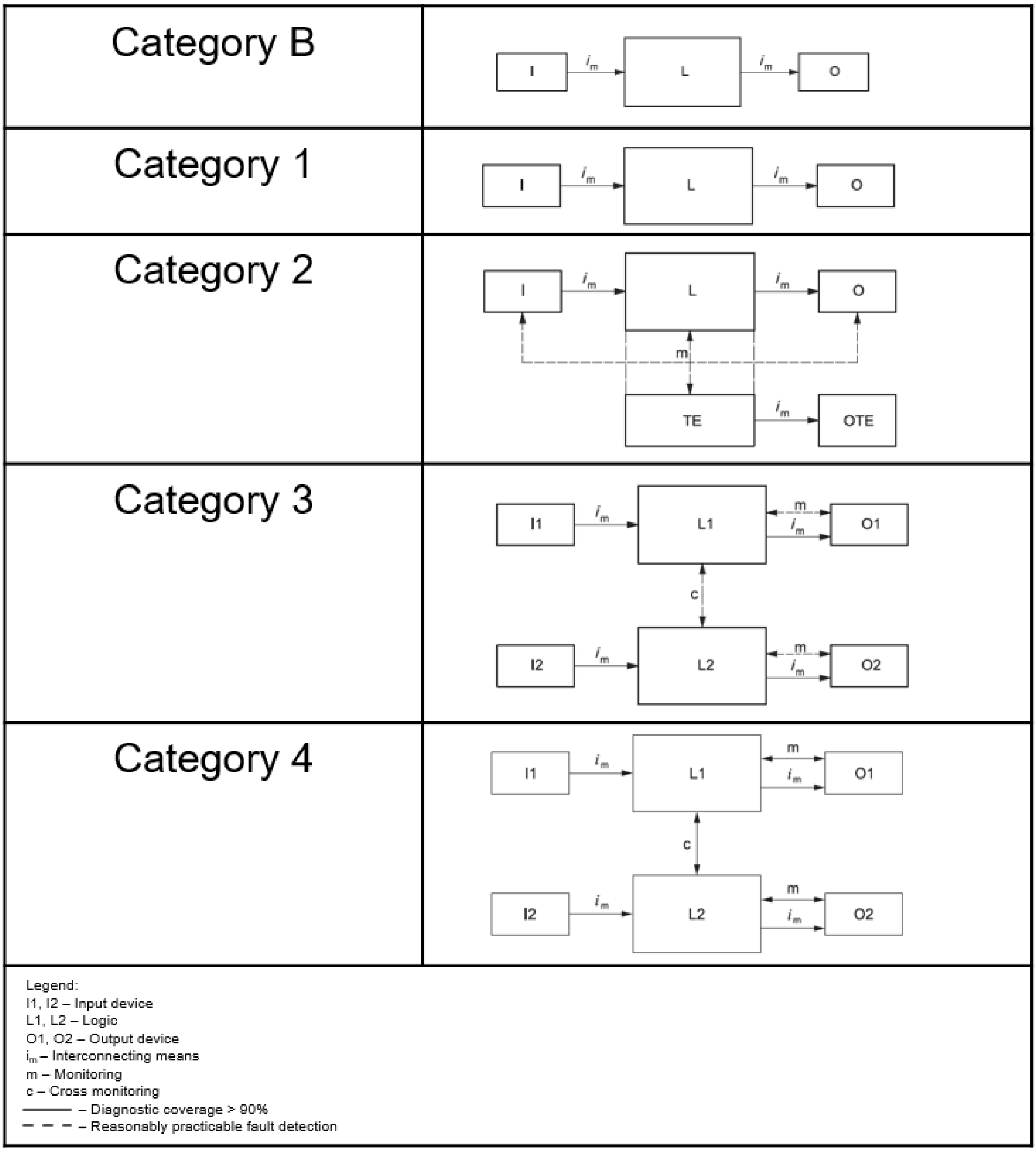


Figure 5. Category representation (applying SFS-EN ISO 13849-1 2015, 138-143).

MTTF$_D$ is a value given by the manufacturer of a device. This value is derived from MTTF$_D$ values of individual components in the system, determined by the standard SN 29500 series database. To reach a certain performance level, the category rating must be higher, thus requiring the diagnostics mentioned in categories 2, 3, and 4. DC is a percentage that shows the ratio of detected dangerous faults to all dangerous faults. (Nix 2017.)

CCF is a numerical value in the standard ISO 13849-1:2015 representing the chance of two different parts of a system failing at the same time and the failures are not consequences of each other. The fault may be caused by an external source or by a another part of the machine. The system is scored by the resistance to failures on such aspects as separation, diversity, design, analysis, competence, and environmental factors. The higher the score, the lower the chance of CCF occurring. (Nix 2017.)

When all the factors mentioned above are taken into account, the PL can be determined using the standard ISO 13849-1:2015. The standard IEC/TR 62061-1:2010 introduces another possible way of giving performance levels, which is called safety integrity level or SIL. The relation between PL and SIL is bridged by PFH_D, as shown in table 1. PFH_D is simplified by PL and SIL to a more understandable format, as the PFH_D values are very small. (Nix 2017.)

Table 1. Correspondence between PL and SIL by PFH_D (applying SFS-EN ISO 13849-1 2015, 19).

Performance Level (PL)	**Average probability of dangerous failure per hour (PFH_D)**	Safety integrity level (SIL)
A	$\geq 10^{-5}\ to\ < 10^{-4}$	No correspondence
B	$\geq 3 * 10^{-6}\ to\ < 10^{-5}$	1
C	$\geq 10^{-6}\ to\ < 3 * 10^{-6}$	1
D	$\geq 10^{-7}\ to\ < 10^{-6}$	2
E	$\geq 10^{-8}\ to\ < 10^{-7}$	3

3.4 Safety logic controller

Safety logic controller is a substitute for the current methods of increasing machine safety. The current methods include using safety modules and controllers, which require extensive wiring or are too simple to be used in complex or demanding applications. For example, a safety function fulfilled by safety modules would be stiff, heavily wired and require vast amounts of components, whereas the same system created with SLC is flexible due to several different factors such as:

- different programming capabilities
- less components
- possibility of wireless communication
- space-efficient design.

Each part of a system is given its own PL from which the total PL of the system is calculated. Having fewer parts in a system eases the process by eliminating less reliable parts and reducing the random fault chance. (Hogan 2015.)

Using an SLC gives a production line the flexibility a working area where humans and robots may work at the same time. The computing power inside the SLC, which safety modules lack, allows usage of more diverse safety devices. The robots can be programmed to decrease speed while the operator is inside the dangerous area. When the operator gets too close to the possible danger, the machine stops, and a restart is required. (Hogan 2015.)

SLCs usually consist of two microprocessors performing the same task simultaneously. The processors compare their program flow to the other processor's program flow, and at the end of each program cycle, outputs are written if no faults are present. SLCs use timed checks or “watchdogs” to monitor if the program runs as intended. If a particular function is not performed in a specific time limit, the watchdog results in a fault state and stops the program. (Bolton 2009, 218.)

In addition to having two microprocessors running the same program, SLCs include built-in testing of inputs and outputs. Inputs are tested in each program cycle for integrity. Outputs are assigned to their processors, and all of them must have the same set state. If this state is not reached, the SLC will order the machine to a known state, bringing it to safe shutdown. (Rockwell Automation 2002.)

Schneider Electric’s products

As there are several different PL_r, the safety-related devices should respond to different requirements. The higher the PL_r, the higher the cost of a device. Some applications do not require a high performance, which spawns the need for a less expensive device. The first level of safety devices is the safety modules, which offer the possibility of safely stopping a motor after a fault has occurred. They are intended for use in smaller machines with low requirement of safety functions. As the machine gets more complex, and the performance demand is higher, a modular safety controller can be used to fulfill the needs. Multi-function modular safety controller can monitor the speed of motors, receive input data from other devices, and set its outputs to control valves and contactors. The highest level of safety device performance is reached with distributed multi-function safety. The Modicon TM5 SLC and XPSMCM modular safety controllers in combination with embedded safety motors bring the machine up to PL_e. Using the mentioned devices, it is possible to achieve safety-related functions such as Safe Stops, Safe Limited Speeds, and others. These devices are illustrated in figure 7**.** (Schneider Electric 2018a.)

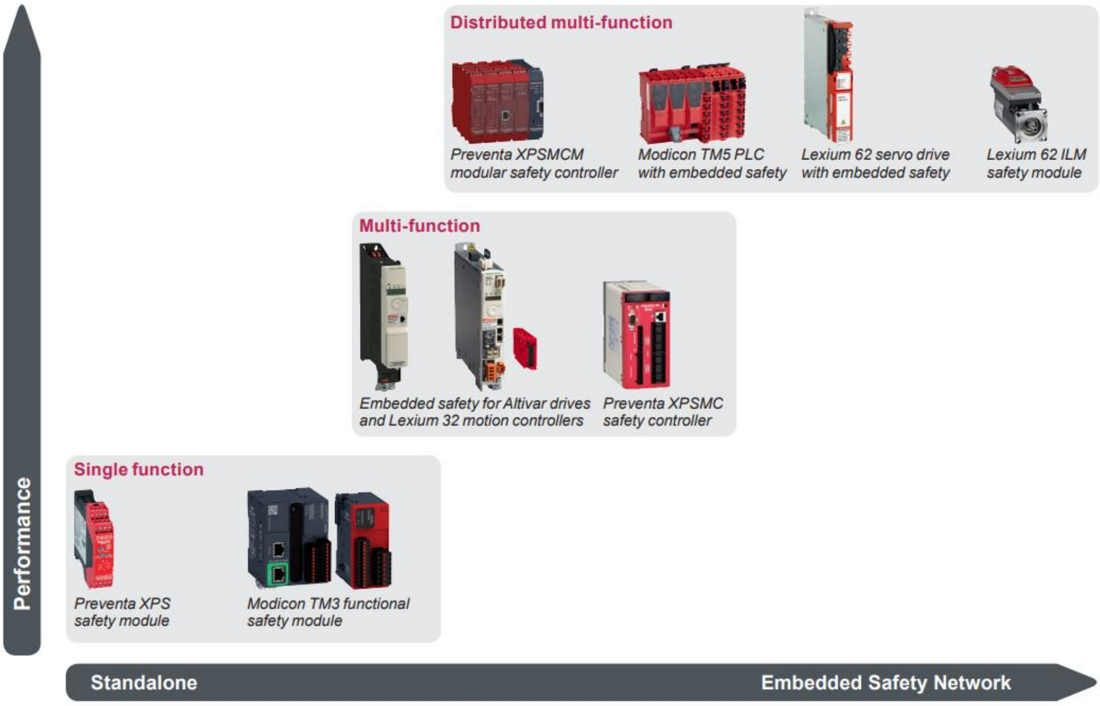


Figure 6. Levels of safety (Schneider Electric 2018a).

The SLC is a SIL 3 device that can manage up to 100 different safety devices at the same time, including inputs and outputs and safety servo drives. It provides the safety-related application the functionality to ensure a consistent safety configuration and verify the different devices and their firmware's during startup and operation, proofing parameter consistency and verification of safety application parameters, and the ability to run safety-related applications for higher safety rating. The safety application and its runtime memory are stored on an external memory key, which allows the user to change the safety program to a different device if needed. (Schneider Electric 2018a, Schneider Electric 2018d.)

Picture 1. TM5 SLC and memory key (Schneider Electric 2018b).

The SLC is certified by TÜV, which is a German agency that provides certifications to vehicles, electronic devices, and factories. In Schneider Electric's case, the SLC is not integrated into the PLC, but it is a separate device. This simplifies the retrofitting of safety devices to older systems, removing the need of buying a completely new system. (Schneider Electric 2018a.)

3.5 Safety programming

Standard IEC 61131-3 defines programming languages to be used in PLC programming, and these apply to safety programming. Structured text (ST), Instruction list (IL) and Sequential function chart (SFC) are text-based programming languages which are included in the standard. Ladder diagram (LD) and Function block diagram (FBD) use symbols instead of text, to create a more graphical method of programming. (ABB [Referred 26.06.2019].)

When developing a safety program to safety devices, the embedded safety software should be verified during and validated after the development process. As verifications are performed during the development, the risk of failures in the program is reduced, and they can be corrected more reliably. Methods in the development of the safety software vary depending on the PL of the safety device. The basis for the development is the V-model illustrated in figure 8. The V-model incorporates verification to every step of the development. The end goal of safety programming and mid-development verification is to produce readable, clear, testable, and maintainable software. (SFS-EN ISO 13849-1 2015, 21.)

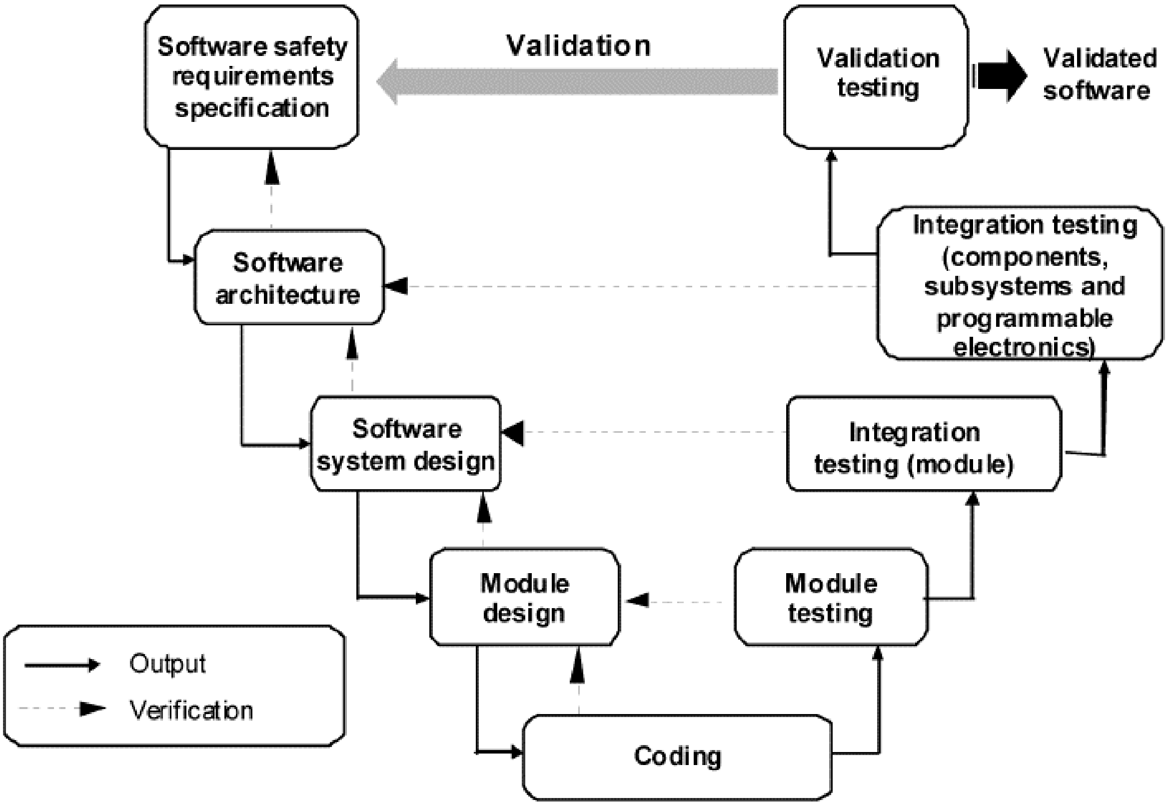


Figure 7. V-model. Safety software development lifecycle (SFS-EN 61508-3 2010, 31).

When developing safety-related embedded software to PL$_{a..d}$ safety devices, the process of development must include basic steps such as

- the V-model
- specification and design documentation
- structural and modular programming methods
- systematic failure
- fault injection

- functional testing with a method of “black-box” testing
- proofing of the safety software.

The usage of mentioned methods ensures that the programmed code is easily readable and fails predictably. “Black box” testing is used for proofing the code without analyzing its contents. (SFS-EN ISO 13849-1 2015, 20-22.)

In addition to methods mentioned earlier, software for PL_c and PL_d devices is also subjected to further validation including usage of a project management system, documentation of all related steps in the whole lifecycle of development, usage of programming languages defined in IEC 61131-3 and software modification impact analysis (SFS-EN ISO 13849-1 2015, 127). Impact analysis composes of different parts of function affecting each other after a change (Guru99 [Referred 13.06.2019]).

PL_e can be reached by $PL_{c,d}$ program when the program is coded to use two channels of the safety system. Otherwise, the program must fulfill the standard IEC 61508-3 software requirements (SFS-EN ISO 13849-1 2015, 128). These requirements cover simplicity, readability, and isolation from the main PLC program. When the program contains both non-safety and safety functions, the entire software is to be considered as safety-related. (SFS-EN 61508-3 2010, 48.)

3.5.1 PLCopen

PLCopen is an independent organization which aims to reduce costs through faster application development, reduced commissioning time, and life-cycle costs (PLCopen A [Referred 18.06.2019]). IEC 61131-3 standard sets the requirements for PLC programming methods and languages. PLCopen brings the requirements to motion control, safety functionality, and communication in a more normalized fashion. For motion control and safety functionality, PLCopen has developed and certified function blocks to be used by the members of the community. The function blocks are offered for integration in languages defined IEC 61131-3 standard. Having function blocks that are already certified by a third party reduces the development time and costs to individual designers. Modular programming methods and certified function blocks reduce human errors in the coding process, significantly decreasing the needed time for troubleshooting. (PLCopen B [Referred 18.06.2019].)

3.5.2 Schneider Electric’s software

Machine Expert – Safety is an add-on to the primary software Machine Expert – Logic Builder. The software is used to program and control the SLC and all other safety devices such as the inputs and outputs and integrated safety devices in servo drives. It offers three standard ISO 61131-3 certified programming languages; FBD, Ladder, and ST, which allow the user to create versatile safety applications. The PLCopen motion control and safety libraries are implemented by default for faster development. Comparing safety programs, either online or offline, can be done for error solving. The safety-related program is protected by two different passwords, one for commissioning and one for development. Using the development password grants full access to programming. All the changes made to the program are saved in separate files, and they can be reviewed at any time. The application created in the safety programming software is downloaded to the device through the Sercos network or by attaching an ethernet cable directly to the SLC. (Schneider Electric 2018c, Schneider Electric 2018a.)

3.6 Stop categories and prevention of unexpected start-up

Every machine must have a control method for complete stopping and safe stopping. General rule for machine safety and stop categories is, that the power to the moving parts is cut when all the moving parts have stopped. According to the SFS-EN 60204-1 standard, this method of stopping a machine fulfills the requirements of category 0 and category 1 stop. A complete and safe stop is defined as a process where the machine stops fast enough and is in a stable state after stopping. Parts of the machine must not move because of gravity or any other reason. Cutting the electrical supply does not always guarantee a safe stop. In need of proper standstill, brakes, and latches must be used. (Siirilä 2008, 175-178.)

In category 0 stop the power from motors is removed, and the movement is brought to a standstill through freewheeling or mechanical braking. Category 1 stop uses the power available to slow down the motor and stop it. When the motor stops completely, the power is removed, as shown in figure 9. (IEC 60204-1 2018, 95.)

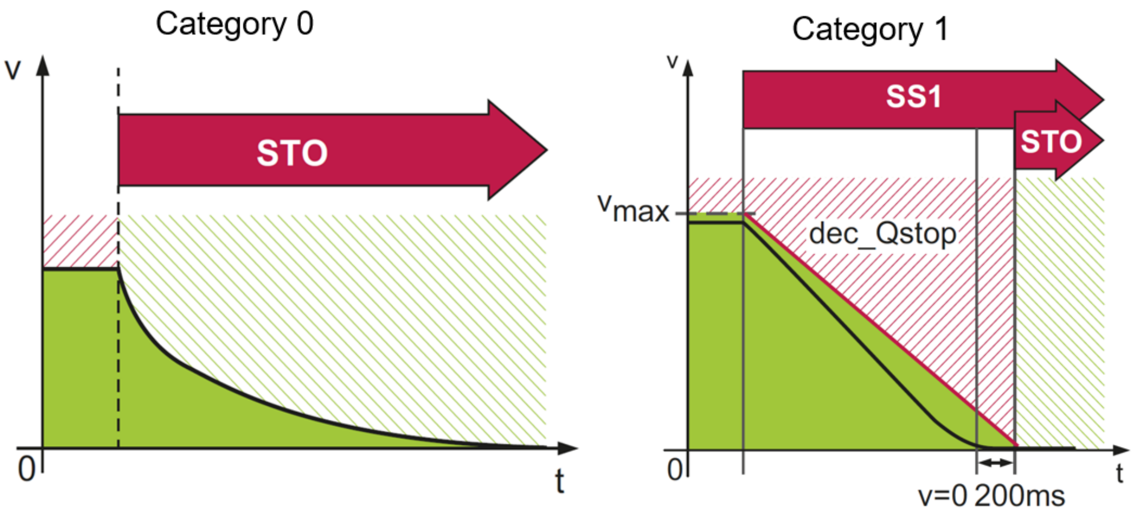


Figure 8. Stop Category 0 and Stop Category 1 (Schneider Electric 2018b).

Power to the moving parts may be maintained if proper monitoring is implemented to the system. This is a category 2 stop, illustrated in figure 10, where power is not cut off, allowing the motors to have full torque but are monitored to remain at stand still. In the event of an unexpected movement by motors or unauthorized entry to a dangerous area, the motors will drop to category 0 stop causing complete standstill and removing power. (Siirilä 2008, 179.)

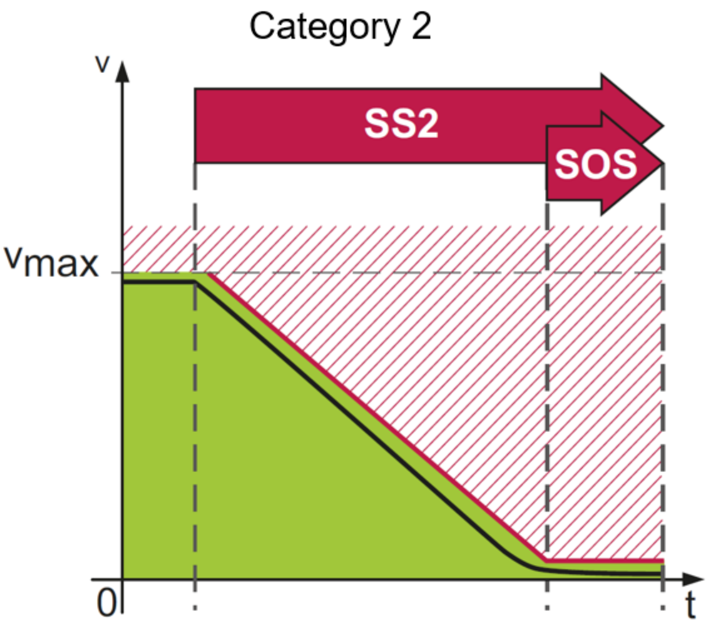


Figure 9. Stop Category 2 (Schneider Electric 2018b).

If a person can enter the danger zone of the machine in a way that all safety devices can be reset, the system must be designed in a way that it cannot be started while safety is compromised. Entrance to the danger zone must be lockable, and it must be designed in a way it cannot be closed accidentally. When the danger zone, and all safety devices are cleared, the stop state must stay active until it is reset by a reset button. (Siirilä 2008, 179.)

4 COMPETITOR ANALYSIS

As introduced in chapter 2.3.3, a comparison test was conducted to the competitor. The analysis was done with a template created in an earlier thesis for Schneider Electric (Nivajoki 2019), which has the characteristics of a comparison test. In the template, the aim was to create a successful Safe Stop 1 from scratch and evaluate different parts of the process, starting from downloading the program to tracing the SS1 action. Even though the devices and programs are different, the task remained the same. The actions are evaluated by the evaluator based on their complexity.

4.1 Evaluation form

As a first step, the evaluator’s prior experience with PLC programming and the software under investigation is asked. This gives the reader of the evaluation an idea of the evaluator’s skills, which helps to distinguish experienced users from inexperienced users. Next, the evaluator must search for training material on the internet to assess how easy it to do the required steps. Material created and provided by the analyzed company is differentiated from the material created by the community or users. The evaluator should look for company-provided programming manuals, commissioning manuals, tutorial videos, and possible example programs. Community made material is also explored, because of it being a good representation of the company's market presence. The more there is material made for users by users, the more the company's equipment is used.

Because of PLC and SLC programming software are occasionally separate, the PLC and SLC programming software manuals are linked to the template for further investigation by the reader of the evaluation. Available coding languages in both programs is asked from the evaluator as these are usually found from the manual and are part of criteria. Usage of PLCopen libraries is asked to check if the company under investigation is part of the PLCopen community. Further questions include safety motion function block count, downloading and installing the software, and the programming task. The evaluator is asked to specify which actions had to be taken to complete the task.

Evaluating the actual usability is asked in the final section of the template with general questions about the feeling of the programming. These questions go more in-depth to the previous questions and require the evaluator to get to know the program even more to find all the required functions of the software.

4.2 Reference evaluation

Each competitor was referenced to a reference evaluation. To get an understanding of competitors pluses and minuses, the reference evaluation was done for Schneider Electric's software Machine Expert - Logic Builder and Machine Expert – Safety using a virtual machine.

First, the installer for Machine Expert installer was downloaded and installed. Through the installer, the software itself and all of its components were downloaded and installed. A license was acquired from a colleague for both Logic Builder and the Safety addon. The licenses were added to the current virtual machine account and registered, even though it was not mandatory, on Schneider's website. Next up, the software was started for the first time. As a secondary task, the physical equipment for the reference evaluation was put together in the architecture configuration shown in figure 12.

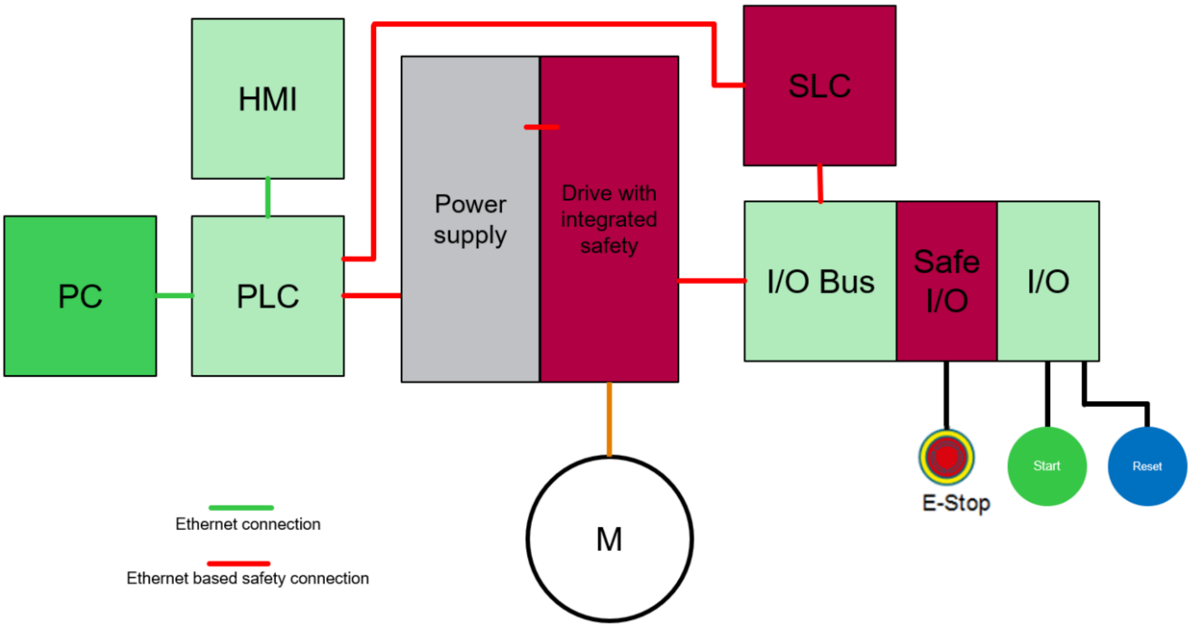


Figure 10. Reference evaluation architecture.

As the software started to run, some changes to the network adapter settings of the virtual machine were required. To connect to the PLC, the network adapter had to be configured to a static IP-address. An IP-address was checked from the PLC’s screen to avoid changing too many settings on the PLC. The adapter settings were changed in the Windows control panel to match the IP-address of PLC, with the differentiation of the last three digits. Matching the IP-address caused the device to be in the same network, but with a different identity. When the connection was made possible to be attained, a new project was created in Logic Builder.

Schneider Electric offered ready-made templates and programs built into the software. The reference evaluation was made using a “Template full” project. The template had devices already added to it and had most of the configuration already done. Having this premade template helped the user to create a project but required some training which was not available to the public. When the project was created, the connection to the PLC had to be established, and afterward, the firmware had to be updated. The update process was done using a separate program which was installed with Logic Builder.

After having connected to the PLC and ensuring that the newest firmware version was installed, the device list in Logic Builder was updated to match the current setup. With one Sercos scan of the system, the devices were identified and added to the project. The devices had to be defined as either real or virtual in the Sercos scan. The project involved using a virtual master axis according to which the real axes were run. When the devices were added to the project, they had to be parameterized, the inputs and outputs had to be mapped, and cam-profiles had to be made. Parameterizing the power supply module to 230v 50hz grid and setting the possible loads and gearing to the motor were done in the settings. Emergency stop, start, and reset buttons were mapped to the correct modules corresponding to the physical connections. The cam-profiles were done for the main axis and the slave axis. The main axis cam-profile was modeled to run continuously at set a speed defined in the program further in the project. Slave axis cam-profile was set as the same in this project as it does not affect the case of SS1 in any way.

To make controlling the SLC easier, an SLC remote controller was imported to the project. The SLC remote controller created by Schneider Electric was used to remotely control the SLC, as otherwise, it had to be controlled by turning the rotary switch with a screwdriver. The remote controller was copied and pasted from a new project to the existing project and added to the tasks to run with the main program.

Next, the necessary changes and programs were made in the Safety program. Global variables were added to the program, and they were linked to the inputs created earlier in Logic Builder by dragging and dropping. From the PLCopen library, an emergency stop function block was added to the program, and inputs of the emergency stop button were linked with it. The integrated safety drive function block created by Schneider Electric was also added to the program. This function block sent the requests to the SLC in a certain event, in this case, an emergency stop button activating, to start monitoring the motor. Shown in figure 13.

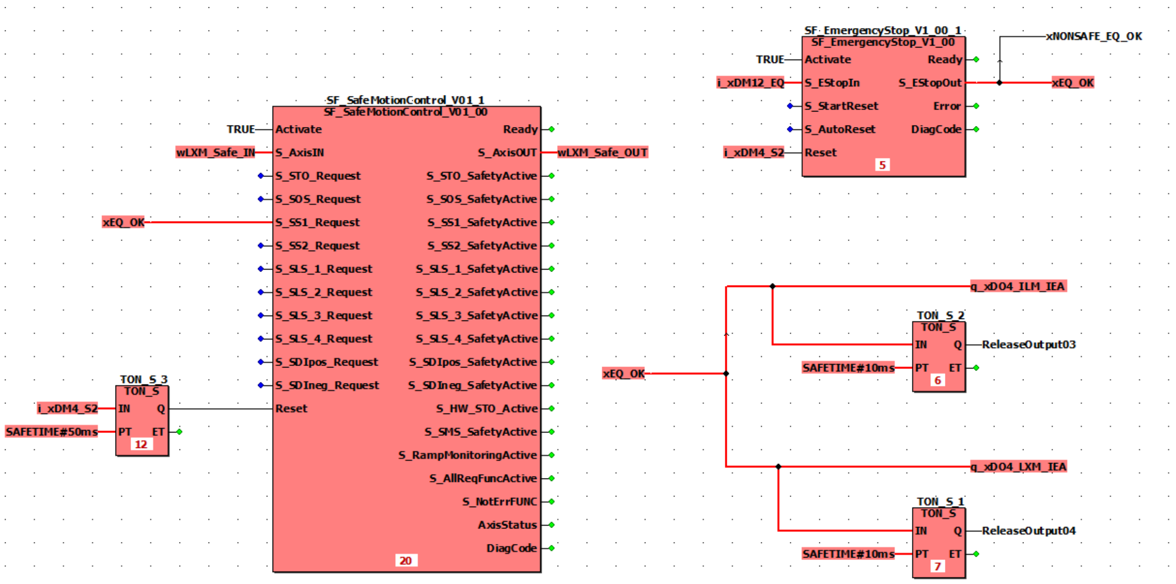
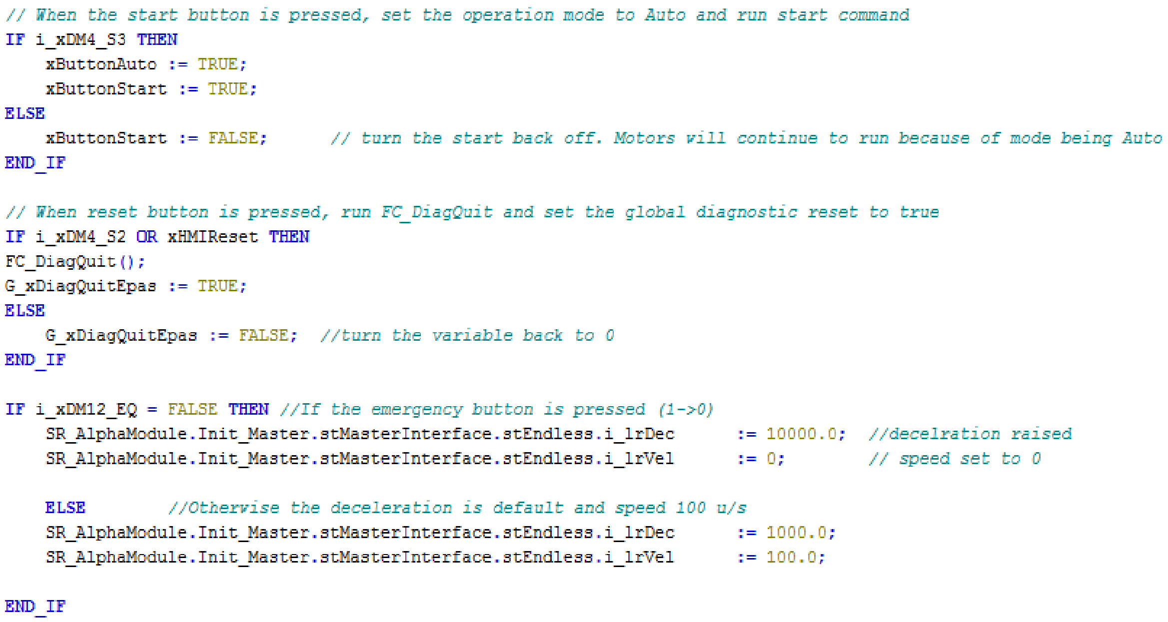


Figure 11. Safety program.

After creating the safety program, the safety devices were parameterized in the Machine Expert – Safety. From SLC, the cycle time, maximum and minimum data transport time were modified to fit the application, and the permission for the SLC remote controller was given. The safety drive was parameterized with the correct values of the requested SS1. The values were derived from the current application. At this point, the safety program could be downloaded to the SLC through the Safety software.

The PLC program was created next. As the SLC in Schneider electric's case is not controlling anything, the deceleration and stopping of the motor had to be done by the PLC. In the template, was already implemented start and mode select functions for each motor, but just as a precaution, extra variables were added to the function blocks in order to control the software with physical buttons. As the template could be programmed with FBD, Ladder, and ST, some of the parts were done in FBD and some in ST. Input action program was made in FBD, which made it clearer and more approachable, so adding just one extra input to the start function was needed. The logic part was done in ST, as illustrated in figure 14, which involved programming a start button to set the variables created earlier and reset button to reset global errors and alarms. Finally, deceleration of the motors and stopping of the program were programmed.



```
// When the start button is pressed, set the operation mode to Auto and run start command
IF i_xDM4_S3 THEN
    xButtonAuto := TRUE;
    xButtonStart := TRUE;
ELSE
    xButtonStart := FALSE;      // turn the start back off. Motors will continue to run because of mode being Auto
END_IF

// When reset button is pressed, run FC_DiagQuit and set the global diagnostic reset to true
IF i_xDM4_S2 OR xHMIReset THEN
FC_DiagQuit();
G_xDiagQuitEpas := TRUE;
ELSE
    G_xDiagQuitEpas := FALSE;  //turn the variable back to 0
END_IF

IF i_xDM12_EQ = FALSE THEN //If the emergency button is pressed (1->0)
    SR_AlphaModule.Init_Master.stMasterInterface.stEndless.i_lrDec      := 10000.0;  //decelration raised
    SR_AlphaModule.Init_Master.stMasterInterface.stEndless.i_lrVel      := 0;        // speed set to 0

    ELSE        //Otherwise the deceleration is default and speed 100 u/s
    SR_AlphaModule.Init_Master.stMasterInterface.stEndless.i_lrDec      := 1000.0;
    SR_AlphaModule.Init_Master.stMasterInterface.stEndless.i_lrVel      := 100.0;

END_IF
```

Figure 12. ST code to perform SS1.

When the logic part was done, the program could be downloaded to the PLC using the Logic Builder. After download, the Logic Builder asked if the PLC would be switched to run-mode. As the PLC was in run-mode, the SLC had to be started using the remote controller. The remote control had a clear user interface, and instructions were given through the whole process.

Finally, the functionality of the system was tested. The testing was done using the tracing option of Logic Builder. A new trace was created, and variables for the velocity of the motor, emergency stop, safety function requested, ramp monitoring, and inverter status were added. Emergency stop, ramp monitoring, and safety function requested variables were added to trace that the response time of the SLC was sufficient. Motor velocity and inverter status were added to see that the motor performed SS1. The trace was downloaded to the PLC and the cycle was started along with the tracing. After the motor got up to speed, the emergency stop was activated, and the motor executed SS1.

4.3 Competitor 1

The 1st competitor's programming software was compared to the reference evaluation with the help of physical equipment. The equipment was ordered directly from the competitor along with the license for the programming software. When the physical equipment arrived, it was assembled to a test rig in the architecture, as shown in figure 15, to ease the process of testing and analyzing.

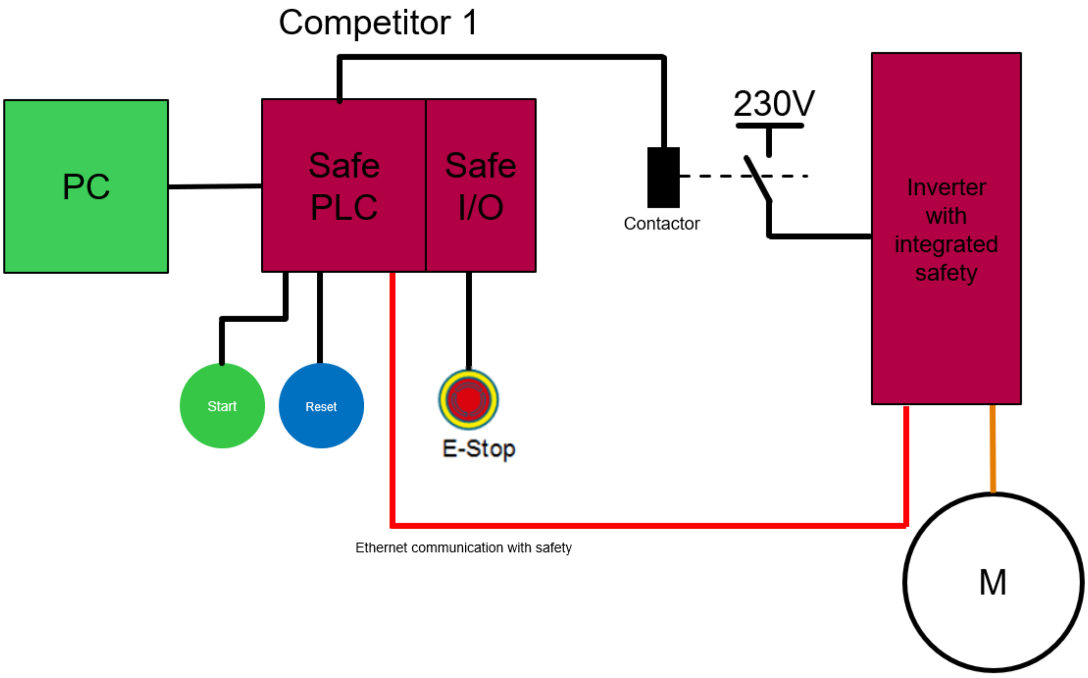


Figure 13. Competitor 1 architecture.

All the ordered equipment was mounted securely on the rig, and proper electrical safety was followed to avoid hazard to the user or the devices themselves. In the case of the competitor 1, the SLC was integrated into the PLC, which made the wiring and mounting clearer and faster. The SLC and the safe I/O extension were interconnected through a bus-system running inside the devices. Start and reset buttons were wired to the SLC’s digital inputs, and the emergency stop button was wired to the safe I/O extension’s digital inputs. One of the SLC’s digital outputs was used to switch the state of the contactor, connecting the inverter unit to the 230V grid. The inverter unit and the motor were attached a safe distance away from the SLC to avoid signal interference. The motor used in the project was a standard three-phase 230V 50Hz induction motor connected to the inverter unit with a regular electrical cable. An ethernet-based communication network was used to connect the devices and to allow the SLC to send commands to the inverter unit. Finally, the test rig was connected to the PC using an ethernet connection.

As in the reference evaluation, the process of creating the example program was started by downloading and installing the necessary software. The software was installed on a virtual machine to ease the process of moving the software between different users. After the programs had been installed, the license was activated, and the programming began. The correct versions of the SLC and the inverter unit were added to the project and parameterized correctly. The safety functions in both devices were turned on, and a password was set to prevent unauthorized access. The non-safe and safe data transfer protocols were added to the project, and the initial downloading to the devices was performed. The variables and the needed program libraries were added to the project, and the programming itself was completed. The inverter unit was commissioned, and the motor data was entered to the inverter unit for proper motor controlling. The safety functions in the inverter unit were turned on and set to perform the SS1 successfully.

Finally, a trace was created to observe the occurrence of the SS1. The traced variables were added to the list, and a trigger event was set. As the last step, all the programs were downloaded to the SLC and the inverter unit, and the test was performed.

Competitor 1 analysis

The competitor 1 performed better in the helpfile section of the analysis. The files found were more comprehensive, and information was easy to find using a search engine. A document giving detailed instructions on how to commission and program the equipment was found, and the project was performed according to the instructions. The complexity of Logic Builder and the 1st competitor's software was similar. Both programs had unique methods of programming different parts of the required task and included things that just had to be known by the programmer. These are illustrated in figure 16.

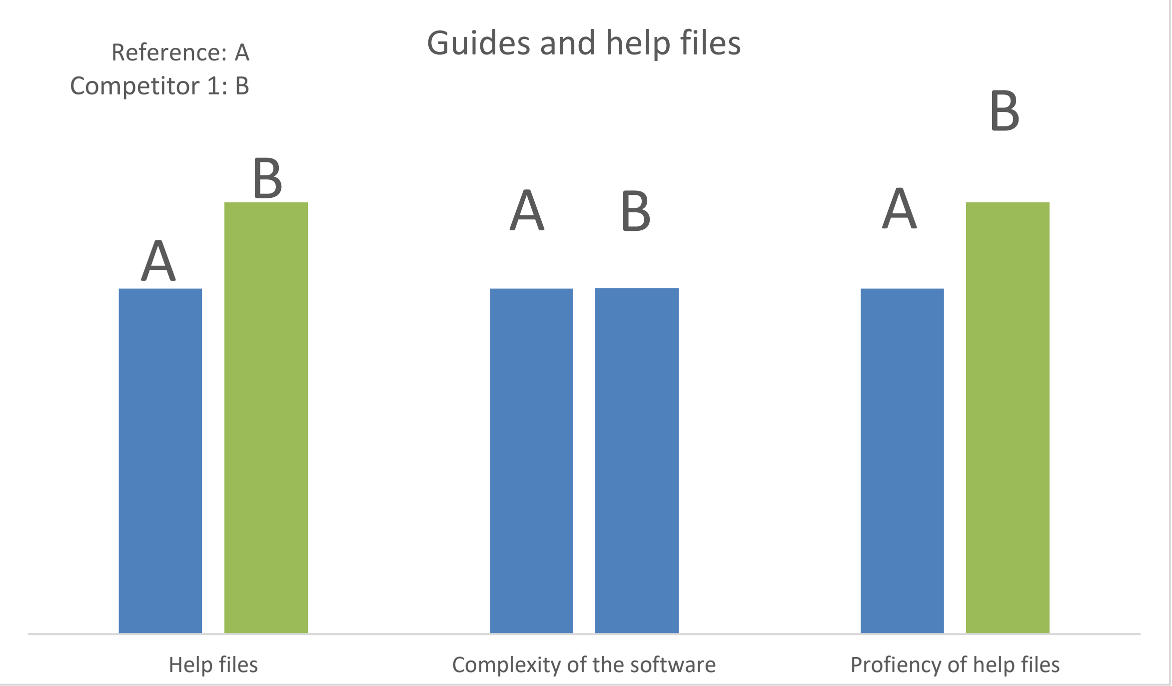


Figure 14. Guide and help file comparison.

As figure 17 presents, the installation and licensing process were similar for both programs. The time it took to download and install the programs was on a similar level. Both the competitor 1 and Logic Builder supported projects created with an older version of the software. Keyboard shortcuts for both programs were identical due to the rebinding of shortcuts.

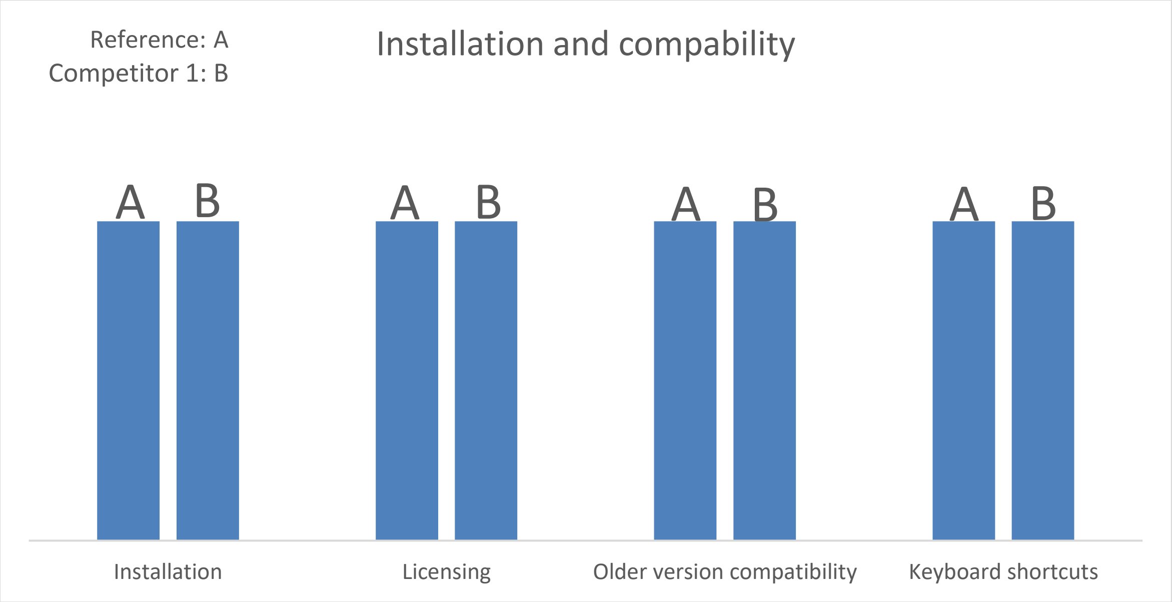


Figure 15. Installation and compatibility comparison.

As shown in figure 18, the competitor 1 performed better in almost every aspect of programming and the project structure. Most notably, the SLC programming and variable linking between the logic and safety programs were made easier. The integration of safety software into the main software was the main reason for simplicity. Parameterization and assisted value entering made the process of achieving motor motion feel safer. Motor data and inverter options were added through separate windows and information was provided through the whole process, whereas in Logic Builder the parameterization was made directly to the motor data parameters. Logic Builder performed better in the way the safety program communicated with the inverter unit. The competitor 1 used two function blocks to control the inverter unit, one for controlling and one for the status of the device. Logic Builder, on the other hand, had combined these two function blocks into one function block, which made the programming clearer.

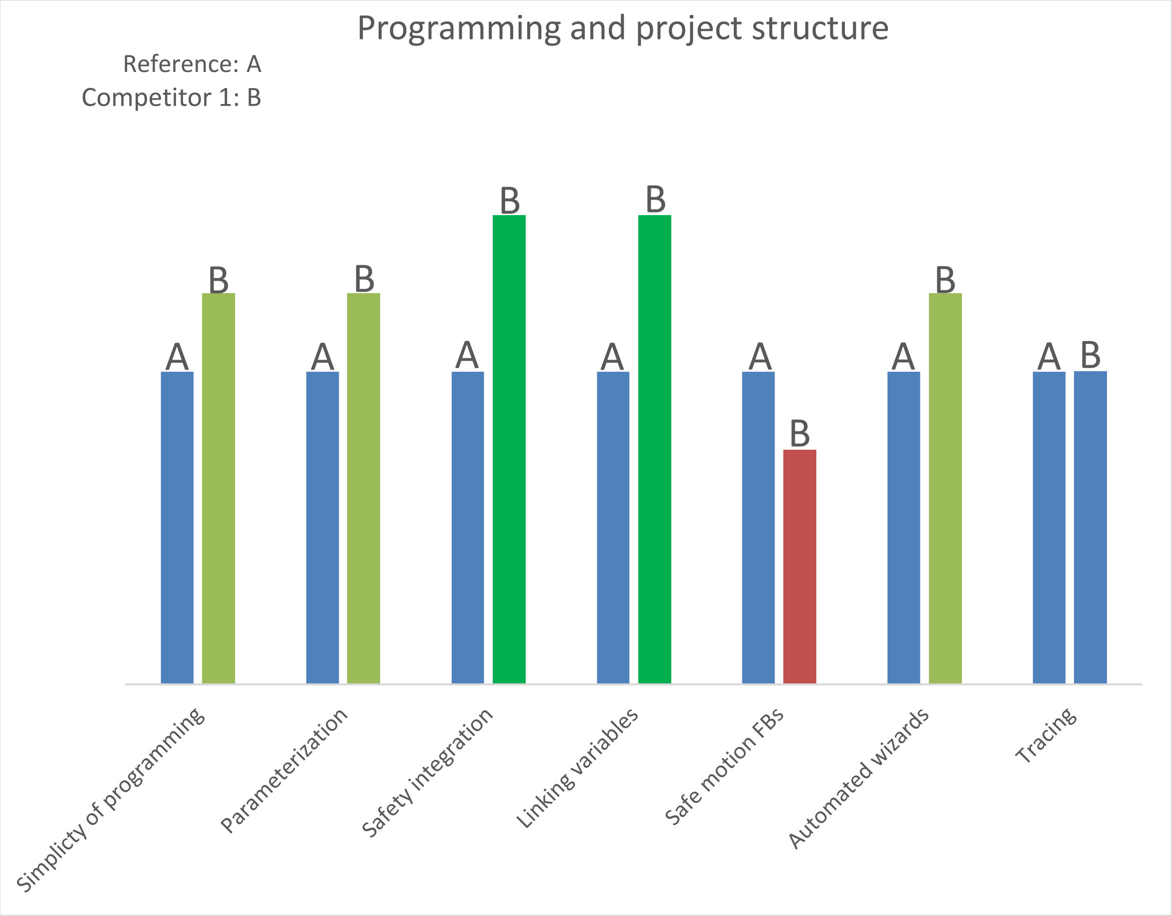


Figure 16. Programming and project structure comparison.

The complete analysis of the competitor 1 was kept confidential and handed over only to the use of Schneider Electric's research & development.

4.4 Competitor 2

Unlike the 1st competitor, the 2nd competitor was analyzed purely on a software level. No physical equipment was used during the analysis. This meant that the execution of the SS1 could not be done, but valuable information about the general feeling of the program was accumulated. The programming software was compared with the reference evaluation to the extent that was possible without the physical equipment.

Information about the software was gathered from the documents provided by the manufacturer and by testing the software. Relevant user guides on how to program were found, and some example programs were discovered, but nothing related to creating a working SS1 application. Community made videos were found on a video uploading platform. They were created mostly by automation solution providers and not by the competitor itself. The software was installed and activated according to the competitor's installation and activation guides.

Competitor 2 performed on a very similar level as Schneider Electric with the help files found on the internet, and the software installation and activation. The difference between the companies’ programs was found in the way of incorporating the safety software into the main software. As the 1st competitor, also the competitor 2 had incorporated the safety software into the main software, removing one step from programming.

4.5 Competitor 3

As the competitor 2, also competitor 3 was analyzed without any physical equipment. The SS1 was not produced using the software, which leaves the usability part untouched. The competitor 3 was analyzed based on the documents and example programs found online.

Information about the software of the competitor 3 was accumulated by analyzing example projects and trying different functions of the software. These methods gave some understanding of the software and enabled the analysis of the software. The competitor 3 had provided a lot of tutorial videos on a video uploading platform. The videos ranged from connecting the PLC with the PC, to a programming guide. However, a guide on how to create the SS1 application was not found. Community generated tutorial or video material was scarcely available.

Compared to Schneider Electric, the competitor 3 performed mostly on a similar level, performing better only in the way the safety program was incorporated into the main software. Schneider Electric, however, performed better in the activation part of the program. The software of the competitor 3 did not use activation codes, but a credit-based system where each function was priced depending on the complexity. These credits were needed when downloading the software to the PLC.

5 CONCLUSION

As the safety-related device and software market is growing, also competition is increasing. To stand out from competitors, the company must provide high-quality products. This thesis gave Schneider Electric insight into their competitors' devices and software, which will help in the development of new versions.

In the theory section, market- and product -analysis methods were taken under observation, Porter's five forces and usability testing being the focus points. Functional safety involving performance levels, stop categories, safety logic controllers, and safety programming was presented.

The practical part consisted of the explanation of the analysis form used in this analysis. The questions and the reasons behind them were explained to clarify the purpose. The reference evaluation of Schneider Electrics software's Machine Expert – Logic Builder and Machine Expert – Safety was made public, but the competitor analyses remained confidential.

As the method of study was qualitative, the results are biased by the thoughts and feelings of the person conducting the analysis. In the analysis, all the competitors had used the same method of incorporating the SLC programming software to the main software, which was a different method compared to Schneider Electric's Logic Builder. The advantages and disadvantages of the integrated or separate program were based on personal preferences, which did not give a definite answer to the question which one was better.

The thesis could be expanded to include more physical testing with the competitors' devices. Work on the analysis template could continue to expand its area of influence from only analyzing software to including also hardware usability.

BIBLIOGRAPHY

ABB. No date available. Overview of the IEC 61131 standard. [www-document] ABB [Referred 26.06.2019]. Available: https://library.e.abb.com/public/81478a314e1386d1c1257b1a005b0fc0/2101127.pdf

Bolton, W. 2009. Programmable logic controllers. Fifth edition. Oxford: Newnes.

Chevalier-Roignant, B. and Trigeorgis, L. 2011. Competitive Strategy: Options and Games. London: MIT Press.

Goble, W. 2003. Learn to Trust Safety PLCs. [Web page]. Control design [Referred 09.05.2019]. Available: https://www.controldesign.com/articles/2003/249/

Goble, W. 2013. Reliable Safety PLC software. [Web page]. Control design [Referred 09.05.2019]. Available: https://www.controldesign.com/articles/2003/257/

Guru99. No date available. What is impact analysis? [Web page]. Guru99. [Referred 13.06.2019]. Available: https://www.guru99.com/impact-analysis-software-testing.html

Hebert, D. 2013. Safety relay or safety-rated PLC? [Web page]. Control design [Referred 09.05.2019]. Available: https://www.controldesign.com/articles/2013/safety-relay-or-safety-rated-plc/

Hogan, H. 2015. Programmable safety cuts costs and adds capabilities. [Web page]. Control design [Referred 22.05.2019]. Available: https://www.controldesign.com/articles/2015/programmable-safety-cuts-costs-and-adds-capabilities/

IEC 60204-1. 2018 Safety of machinery – Electrical equipment of machines. International Electrotechnical Commission. Helsinki: Finnish Standards Association.

IEC/TR 61508-0. 2012. Functional safety of electrical/electronic/programmable electronic safety-related systems. Part 0: Functional safety and IEC 61508. Helsinki: Finnish Standards Association.

IEC/TR 61508-1. 2012. Functional safety of electrical/electronic/programmable electronic safety-related systems. Part 1: General requirements. Helsinki: Finnish Standards Association.

Marburger, D. 2012. How strong is your firm's competitive advantage? New York: Business Expert Press.

Markets and markets. 2018. Machine safety market. [Web page]. Markets and markets. [Referred 11.07.2019]. Available: https://www.marketsandmarkets.com/Market-Reports/machine-safety-market-1188.html

Nivajoki, J. 2019. Investigation of embedded safety software in the safety market: Accessibility, learnability and usability of each software. Turku University of applied sciences. Degree programme in mechanical and production engineering. Bachelor’s thesis. Published.

Nix, D. 2010. Interlock Architectures – Pt. 2: Category 1. “Well-tried components” and “Well-tried safety principles”. [Web page]. Machinery Safety 101 [Referred 15.05.2019]. Available: https://machinerysafety101.com/2010/07/28/interlock-architectures-pt-2-category-1/

Nix, D. 2017. How to do a 13849-1 analysis. Defining required performance level according to ISO 13849-1 standard. [Web page]. Machinery Safety 101 [Referred 14.05.2019]. Available: https://machinerysafety101.com/series/how-to-do-a-13849-1-analysis/

PLCopen A. No date available. What is PLC open. [Web page]. PLCopen. [Referred 18.06.2019]. Available: https://www.plcopen.org/what-plcopen

PLCopen B. No date available. Benefits of PLCopen membership. [Online publication]. PLCopen. [Referred 18.06.2019]. Available: https://www.plcopen.org/sites/default/files/downloads/benefits_of_plcopen_membership.pdf

Porter, M. E. 1998. Techniques for analyzing industries and competitors. 1st edition. New York: Free Press.

Porter, M. E. 2008. The five competitive Forces That Shape Strategy. [Online publication]. Harvard business review. [Referred 12.04.2019]. Available: http://www.business.uwm.edu/gdrive/Goranova_M/Readings_712/Porter%205%20forces.pdf

Prahalad, C. K. 1999. Changes in the Competitive Battle field. Financial Times.

Rockwell Automation. 2002. PLC vs. safety PLC – Fundamental and significant differences. [Online publication]. Rockwell Automation [Referred 22.05.2019].Available: https://literature.rockwellautomation.com/idc/groups/literature/documents/wp/safety-wp002_-en-e.pdf

Rubin, J & Chisnell, D. 2008. Handbook of usability testing: How to plan, design, and conduct effective test. 2nd edition. Indianapolis: Wiley Publishing, Inc.

Schneider Electric A. No date available. Schneider Electric machine solutions, Marktheidenfeld. [www-document]. Schneider Electric. [Referred 04.04.2019].

Available: https://www.se.com/de/de/about-us/careers/vocational-training/marktheidenfeld/

Schneider Electric B. No date available. Schneider Electric Marktheidenfeld company profile. [www-document]. Schneider Electric. [Referred 04.04.2019]. Available: https://www.se.com/de/de/about-us/company-profile/standorte/marktheidenfeld.jsp

Schneider Electric. 2012. Safe Machinery Handbook. [Online publication]. Schneider Electric [Referred 06.05.2019]. Available: https://www.schneider-electric.co.uk/en/download/document/SE7847/

Schneider Electric. 2018a. Safety Modicon TM5 & TM7: safety logic controllers and safety I/O modules. [Online publication]. Schneider Electric [Referred 02.04.2019]. Available: https://www.se.com/fi/fi/download/document/DIA3ED2160309EN/

Schneider Electric. 2018b. 01 Machine_Safety_June_2018. [PowerPoint-presentation]. Schneider Electric [Referred 02.05.2019]. Available: Only for internal use.

Schneider Electric. 2018c. Software introduction. Introduction to Machine expert – Safety. [PowerPoint-presentation]. Schneider Electric [Referred 26.06.2019]. Available: Only for internal use.

Schneider Electric. 2018d. Modicon TM5 – Safety logic controller SLC100/200 FS – Hardware guide – 09/2018. [Online publication]. Schneider Electric [Referred 02.07.2019]. Available: https://www.schneider-electric.com/en/product/TM5CSLC200FS/safe-logic-controller---cpu-plus---sercos-iii-interface---80-nodes/

Schneider Electric. 2019. Schneider Electric in numbers. [www-document]. Schneider Electric [Referred 04.04.2019]. Available: https://www.schneider-electric.com/en/about-us/investor-relations/investment/overview.jsp

SFS-EN 61508-3. 2010. Functional safety of electrical/electronic/programmable electronic safety-related systems. Part 3: Software requirements. Helsinki: Finnish Standards Association.

SFS-EN ISO 13849-1. 2015. Safety of machinery. Safety-related parts and control systems. Part 1: General Principles for design. Helsinki: Finnish Standard Association.

SFS-EN ISO 13849-2. 2015. Safety of machinery. Safety-related parts and control systems. Helsinki: Finnish Standard Association.

Siirilä, T. 2008. Koneturvallisuus, EU-määräysten mukainen koneiden turvallisuus 2. Renewed edition. Helsinki: Fimtekno oy.

VTT. 2004. Turvallisuuteen liittyvät ohjausjärjestelmät konesovelluksissa. Safety related control systems in machine solutions. [Online Publication]. VTT [Referred 03.04.2019]. Available: https://www.vtt.fi/inf/pdf/tiedotteet/2004/T2264.pdf