Experimentation on the Electromechanical Behavior of Automation Safety Buttons Applied to an Industrial PLC.

E. THEOCHARIS
Dept. of Industrial Design and Production Engineering
University of West Attica
Thivon 250 & P. Ralli, Egaleo
GREECE

M. PAPOUTSIDAKIS
Dept. of Industrial Design and Production Engineering
University of West Attica
Thivon 250 & P. Ralli, Egaleo
GREECE

A. SORT
Dept. of Industrial Design and Production Engineering
University of West Attica
Thivon 250 & P. Ralli, Egaleo
GREECE

C. DROSOS
Dept. of Industrial Design and Production Engineering
University of West Attica
Thivon 250 & P. Ralli, Egaleo
GREECE

Abstract: - The growing automation demands of production make the automation systems more complex and vulnerable to failures. For this reason, some instructions have been created (CAT, SIL) that must be followed, in order to insure their safe operation in case of a failure of both the hardware and the software. For a credible operation of a Fail Safety system along with a system to work on SIL2 or SIL3, must have Safety Hardware and Software. This present paper analyses the response of the Hardware of a Basic PLC and the equipment of an automation system. It also presents a descriptive analysis of the experiment, which was conducted to record measurements in order to draw firm conclusions. In addition, the measurements are analysed and evaluated to verify if basic equipment could be used in these systems and insure the Safety function at the same time. The objective is to simply prove that if we manage an already existing Basic PLC equipment differently, it could upgrade the security of automation systems. Therefore, with a low cost in time and money, particularly in existing automation systems, there could be Fail Safety operations.

Key-Words: - Safety Relay, Safety PLC, Safety integrity level (SIL)

Received: September 1, 2020. Revised: December 15, 2020. Accepted: December 19, 2020.
Published: December 23, 2020.

1 Introduction

The reduction of risks in a production process depends on many factors. The main factors are:
a) The separation of the workers from the production machines in natural ways, like doors, bars, etc.
b) The electronic and mechanical equipment to be completely reliable, so that in case of a problem to deflect any accidents.[1,4]
In the first automation systems where Safety operation was needed, due to the process, Safety Relays were used. Safety Relays are specially designed Relays formally certified to constantly be armed. Thus, when wiring all the Normally Closed contacts of the protection elements in order (e.x. Stop Emergency) for the armament of the Relay, it automatically shuts down if any of these contacts turn on. For the operation of the automation system one Normally Open contact of Safety Relay is used, as a consequence by deactivating the Safety Relay it automatically pauses the operation of automation.
Safety Relays are still used today in simple automation systems providing additional capabilities, such as two or more control channels, time delays, communication abilities etc. In figure 1 below a typical Safety Relay use is being described.

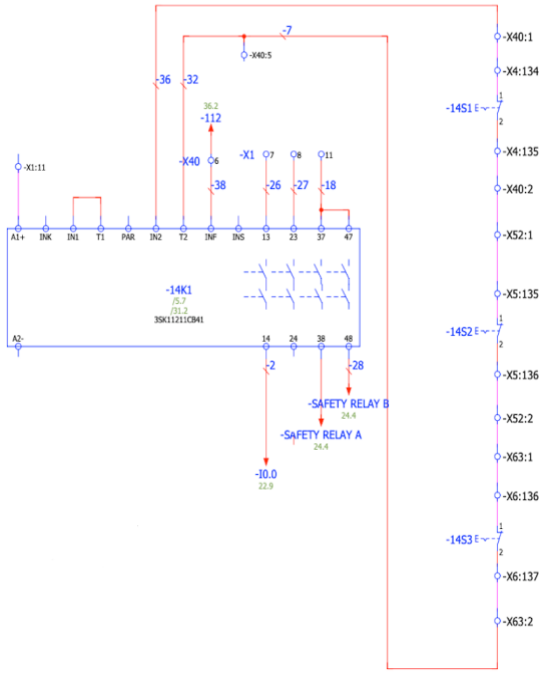


Fig. 1: Typical use of a Safety Relay

As shown in figure 1, there are three «Stop Emergency» buttons. The first button takes in current from the Safety Relay and it is wired up in series with the other two. So, if one of the three buttons is pressed the current turns on and there is no more voltage return in the Safety Relay. As a result of losing the voltage, the Safety Relay opens the contacts Safety Relay A and B and the automation system operation is then paused.

If the process requirements for Safety automation operations are large and complex then it is not covered only by Safety Relays, so Safety PLC is then used. For instance, there is not just a Stop emergency in such a process but there is also a Laser Curtain used. That means, if a worker walks through an area where machines are placed, then their operation must be stopped and start again once the worker has exited that area. In fact, the operation should resume after a certain time and after a Reset is performed.

Now, most PLC manufacturers produce Safety PLC with additional functions, both in Hardware and Software rather than in Basic PLC. There are PLCs that are only used for Safety operations and PLCs that perform both automatic and Safety operations at the same time.

In very critical facilities like refineries, airports, nuclear power plants, we use double Safety PLC, one is the Redundancy of the other. In a redundant system we have two CPUs which run the same program at the same time and are synchronized (usually with optical fibers) so that they are on the same processing step. If a CPU defects then the other CPU undertakes in order to resume controlling the system. The control of proper functioning and control transmission from one CPU to another is not programmed by the respective automation mechanic, but it has been implanted by the PLC manufacturer. That is to say, he has gone under extensive chek-ups and certifications, thus offering greater reliability. If both CPUs are available, only one of them has the control of the automation system, and so it is stated as Master. Primary and Backup are the two CPUs, while the ET200SP and ET200MP are signal cards for the automation system checkup and each time are being monitored from the Master CPU. The reticulation between them is in Ring topology, which means that if the cable is cut anywhere, the system will continue to operate normally. In most redundant systems, the channel switching time from one CPU to the other is less than 100 msec. This results in creating no malfunctions in the control system transitioning from one CPU to the other. In Figure 2 below, we can see this kind of system [7].

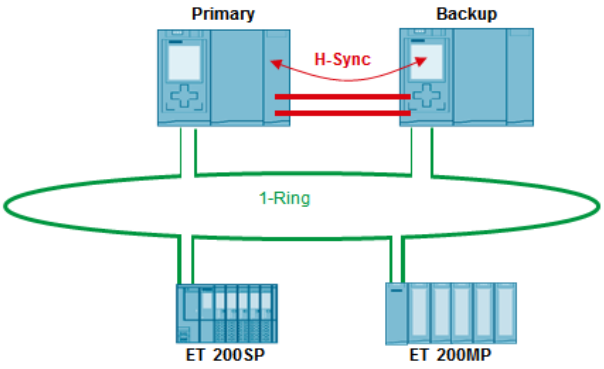


Fig. 2: Redundant PLC System

The basic Safety PLC properties are:

…in Hardware:

A Safety PLC is a specially designed controller for security use (there are PLCs that can operate both as security controllers and as automation controllers). The Safety PLC owns self-testing functions. It has high reliability and it meets the required security standards such as SIL3 / IEC61508 and Category 4 / ISO 13849. [2,5]

The Safety PLC interacts with the Automation PLC through Input/Output network or signals, it manages the emergency system and when the conditions are followed it then allows the Automation PLC to operate properly. In Safety PLC all the security sensors are connected with double wiring and double-checking contacts. Finally, by having double commands, the actuators are activated.

...in Software:
The security standards require strict restrictions of the programming languages of a Safety PLC. For this reason, most Safety PLC manufacturers provide special programming languages which are formally certified to cover these restrictions. In the operation of a Safety PLC not only the signal check-up routines are operated, but also the routines to ensure proper operation as a CPU. Only a certified mechanical engineer can remotely modify the operation of a Safety PLC by default. The modification can occur through the security level settings by automatically recording the alterations (signature). [1].
The following units describe the use of Safety Input - Output PLC cards. The experiment checks how the Inputs of a Basic PLC respond so that they can be used to upgrade the security of automation systems

2 Signal Implementation of Safety PLC

The signals (Input - Output) of a Safety PLC are being described below. It will then be ascertained whether or not this connection and the additional internal security features of a Safety PLC signal card could be implemented using a Basic PLC.

2.1 Sensor wiring on safety PLC input cards

The connection of the sensors to the input cards can occur with a single channel or with a double channel. The Safety Input cards have two independently galvanic ally isolated channels. The sensor wiring capabilities are being given below. [2,4]

2.1.1 Sensor wiring with a single channel (1oo1)

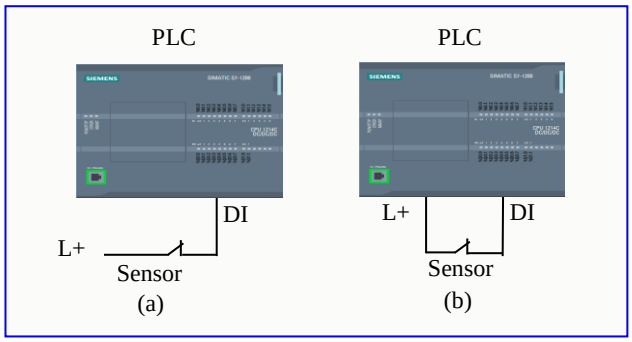


Fig. 3: Sensor wiring with a single channel (1oo1)

In figure 3 above, there is a contact wiring (Normally Closed for safety reasons) of a sensor in a PLC input. The sensor can be powered by an external power supply (a) or by the PLC (b). Although, if this wiring is connected to a Fail Safe input card in a PLC, it can provide security Cat.2/ PLc/SIL1.

2.1.2 Sensor wiring with double channels (1oo2)

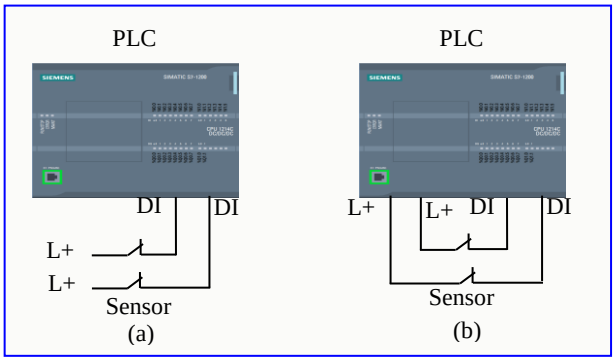


Fig. 4: Sensor wiring with double channels (1oo2)

In figure 4 above, there is wiring of two contacts (Normally Closed for safety reasons) of a sensor in two PLC inputs or of two independent sensor wirings from one contact (Normally Closed for safety reasons) in two PLC inputs. The Fail Safe input cards of the PLC check both inputs and they transfer the information in the CPU as a single input whether they are activated or not or if there is a malfunction (e.x. having a signal at one input but not having at the other). The power supply of a sensor can occur by an external power supply (a) thus, having security up to Cat. 3/PLd/SIL2 or from the (b) thus, having security up to Cat. 4/PLe/SIL3.
The above link assemblies refer to Input Fail Safe cards, where they have extra integrated features, like cut cable control, non-synchronization of inputs, etc. For this reason, these systems could be formally certified by the manufacturers even up to SIL3.

2.2 Wiring actuators in safety PLC output cards

The wiring capabilities of the sensors with one Relay per output and with two Relays per Output are given below.

2.2.1 Relay wiring per output

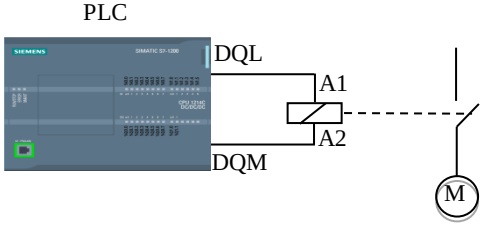


Fig. 5: Relay wiring per output

Figure 5 above displays the wiring of a Relay with one output from a Fail Safe card. In Fail Safe output cards, each output does not only give e.g. 24VDC for the activation of the Relay but also 0VDC. That is to say, when there is no output command from the PLC to the Relay, the circuit in both A1 and in A2 is open. The Fail Safe output cards have additional diagnostic routines for extra check-ups e.g. cut cabling, command capability check when it is required etc.

2.2.2 Two relay wiring per output

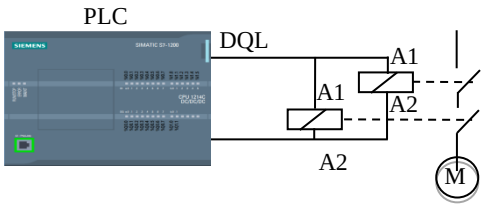


Fig. 6: Two relay wiring per output

Figure 6 above displays the wiring of two Relays with one output from a Fail Safe card. With this linkage we could have security up to Cat.4/PLe/SIL3. One of the two Relays could have a constant A1 voltage and load when through the Q of a PLC shuts down the circuit from the A2. It is usually recommended that both Relays are wired to the PLC card. In this function, a digital output is being commanded through the program. This information is then transferred to the output card and afterwards both relays are loaded. By using two relays, it ensures that the ordered component (e.g. an engine) will not start operating if for some reason only one relay is loaded or is not deactivated.

3 Double Contact Response of a Basic PLC Experiment

In the following experiment, the double contact response of a component is checked (e.g. button). Through the results of the experiment, we will draw conclusions whether it is possible to have not only SIL 1 function but also SIL 2, using concessional cards in a PLC.

With this experiment, the response of double contacts of a component (like that of a button) will be examined, both in the beginning of its operation and when it has been used a few thousand times. Knowing the expected response of the component, with the functions that support the Basic PLC, and not the Safety ones, the component can be examined and a Safety function can be produced. With this implementation as well as with many potential assumptions required, leads to the conclusion that the reliability of the already existing automation systems could be increased without any additional cost-effective equipment but also without any specialized installing and programming procedures. In this case of course, the Safety operation of the component (e.g. the button) will be achieved and not the PLC Safety operation of a CPU.

3.1 Overview of the Experiment

The implementation of this experiment required industrial equipment, which is used to monitor automation systems and is described in detail below.

3.1.1 Piston – Push button

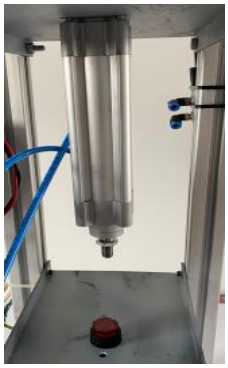


Fig. 7: Piston – Push Button

Ain air piston was installed to simulate the push of the button. When the piston takes an order from the PLC, it goes down and presses the button. The PLC command (i.e. a digital output) gives 24VDC to a valve which then channels air into the piston, causing it to go downward. By deactivating the command, the valve alters the air supply to the piston and this way the piston goes upward. This procedure checks the piston and therefore the push of the button. By adjusting the pressure of the air which channels into the piston, it is more possible to monitor the speed at which the button is pressed (slow or fast), something that will lead to additional information for the contact response of the button. To the button, a 24VDC lamp and two Normally Closed contacts are connected. When the button is pressed then the lamp switches on and the contacts

open up. The buttons were used from two different manufacturers, and this was done in order to monitor the behavior of each button and not which company button is better. Also, the buttons of the experiment have automatic reset so that they do not buckle and then need to be pulled or rotation to unbuckle. So, with the automatic reset there is a possibility of direct push of the button. In this way, many automatic presses are achieved in minimal time.

3.1.2 PLC

A Siemens series S7-1500 PLC was used to command the piston and also to read the contacts of the button.

Fig. 8: PLC

More specifically, a CPU 511 was used. That is one of the smallest in CPU capabilities in the mid-range PLC series which are owned by SIEMENS. It has input and output signals. This way, one digital output was used to command the piston and two digital inputs to read the contacts of the button.
That specific CPU (like most of the CPUs) has the ability to read Outputs in time less than 0,05m/s and Interrupt routines for a more immediate reading and processing of the Inputs.

3.1.3 Scada

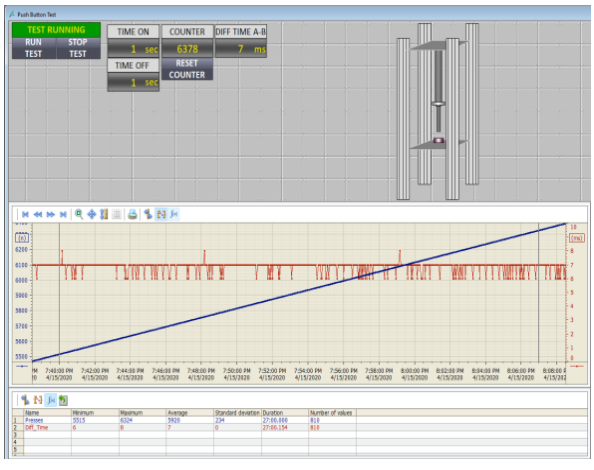


Fig. 9: Scada

The SCADA of the experiment was developed in WinCC V7.5 of SIEMENS. The WinCC V7.5 has the communication interface with the PLC (Profinet is used for the experiment), Visualization abilities, Control and Tag logging which are required for the conduct of the experiment.

3.2 PLC- Piston control

The automation check of the experiment was done through a TIA V16 program. An FB1 routine was developed (Function Block) in SCL language to control the Piston. With FB1 routine, the operation of the piston is controlled via SCADA, ie when and for how long it will be activated to push the button. [10]

```
REGION Pulse Generator
    #TIME_OFF := "CLOCK_DATA".TIME_OFF * 1000;
    #TIME_ON := "CLOCK_DATA".TIME_ON * 1000;
    #IEC_Timer_0_Instance(IN := NOT "Tag_1",
                          PT := #TIME_OFF,
                          Q => "PULSE_ON_OFF");
    #IEC_Timer_0_Instance_1(IN := "PULSE_ON_OFF",
                            PT := #TIME_ON,
                            Q => "Tag_1");
END_REGION

REGION Piston Control
    "OUT_EMBOLO" := "ENABLE" & "PULSE_ON_OFF";
END_REGION

REGION Piston Position
    #IEC_Timer_0_Instance_3(IN:="OUT_EMBOLO",
                            PT:=T#6S,
                            ET=>#TIME_OUTPUT);
    "PISTON_POSITION" := #TIME_OUTPUT / 14;
    #IEC_Timer_0_Instance_4(IN:=NOT "OUT_EMBOLO",
                            PT:=T#6S,
                            ET=>#TIME_OUTPUT_1);
    IF ("OUT_EMBOLO" = 0)  THEN
        "PISTON_POSITION" := (1000-#TIME_OUTPUT_1)/14;
    END_IF;
    IF ("CONTACT_A"=0)OR("CONTACT_B"=0) THEN
    "PISTON_POSITION" := 80;
END_IF;
IF ("PISTON_POSITION" < 0) THEN
    "PISTON_POSITION" :=0;
END_IF;

END_REGION

REGION Lamp ON-OFF
    IF ("CONTACT_A" = 0) OR ("CONTACT_B" = 0) THEN
        "LAMP_ON" := 1;
    ELSE
        "LAMP_ON" := 0;
    END_IF;
END_REGION
```

Fig. 10: FB1

Pulse Generator: A pulse generator was designed by using two IEC TON Timers. The ON / OFF interchange duration is regulated by the SCADA.
Piston Control: When SCADA gives the order 'ENABLE' and the pulse generator 'ON', the piston is ordered to go downward. As a result, when there

is ‘ENABLE’ signal from SCADA, the piston implements a back-and-forth motion.
Piston Position: Depending on how long the piston command is activated, the corresponding value in the ‘PISTON POSITION’ variable is generated in order to show the position of the piston in SCADA.
Lamp ON-OFF: If one of the input contacts of the button is lost, then it activates the ‘LAMP ON’ variable in order to show how the button was pressed in SCADA.

3.3 PLC - Inputs Control

The following procedures were implemented to monitor the inputs in PLCs.

3.3.1 Deactivation of delay filter

To avoid interference of installing to an automation system, the inputs in PLC have the ability to initiate an activation delay for one channel or for a pair of channels.
The interference pulses, of whose pulse is less than the specified input delay (ms), are ignored and thus are not visible in PII of the PLC.
For the purposes of this experiment, that delay was completely deactivated in order to achieve an immediate response as the one of the PII in PLC.
To be noted, that the experiment was carried out in the laboratory, where there is no electromagnetic noise as there is in an industrial environment.

Fig. 11: Input Parameters

When the Input Delay is deactivated, it is very possible to read the response of an input in time less than 1m/s.

3.3.2 Activation of interrupt procedure

Because the response of an input in the CPU of a PLC needs to be read as immediately as possible, the Hardware Interrupt procedure is then activated. With the Hardware Interrupt procedure the input in a PLC routine is read immediately when the input state changes. In this way, it was avoided to be carried through the Main routine of its cycle, because that would mean additional delay even for unstable time intervals. [3]

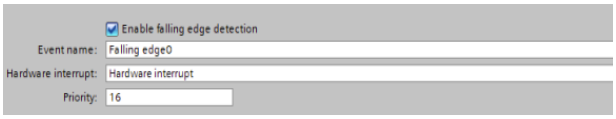


Fig. 12: Hardware Interrupts

The figure above shows the activation of the Hardware Interrupt at Falling Edge, in other words a transition to the Interrupt routine the moment the Input transmission goes from 1 to 0. The contacts used to the button are Normally Closed. An alternative Interrupt routine is performed for each Input. The following code runs in each Interrupt routine:

```
1
2 REGION Read CPU Clock
3     "CLOCK_DATA".CLOCK_TIME_OB_A := #DATE_TIME;
4 END_REGION
5
6 REGION Time A
7     "CLOCK_DATA".CLOCK_TIME_A := #DATE_TIME;
8 END_REGION
9
```

Fig. 13: Read PLC Clock

Read CPU Clock: The Real Time Clock of a CPU is read.
Time_A: The value is assigned by the Real Time Clock of the CPU to the variable: “CLOCK_DATA”. CLOCK_TIME_A. That command is performed when the Interrupt routine is activated, in other words when the Input is deactivated. With this method, the Real Time Clock of the CPU is stored in the variable above.

3.4 PLC - Difference record

```
1
2 REGION Calculate Difference
3   "R_TRIG_DB"(CLK:=("CONTACT_A"=0)&("CONTACT_B"=0),
4               Q=>"Tag_6");
5     IF "Tag_6" THEN
6       "CLOCK_DATA".CLOCK_TIME_DIFF := T_DIFF(IN1 := "CLOCK_DATA".CLOCK_TIME_B, IN2 := "CLOCK_DATA".CLOCK_TIME_A);
7     END_IF;
8
9 END_REGION
```

To calculate and record the deactivation time difference of two Inputs the following procedure was implemented in the pattern FB2 (Function Block):

Fig. 14: FB2

Calculate Difference: When in both of the Inputs there is 0 state, the time difference of the response between the two Inputs is calculated (in ms). The accuracy of the calculation, with the specific

Configuration in the experiment, can be per 1 ms. This accuracy, as will be ascertained below, is more than enough for the needs of the experiment.

3.5 PLC - Push button count

Button response does not remain the same, not only over time, but also after extreme usage, so that must be acknowledged by the user for more effective results.

To check the response of the button contacts whether it is similar or different after a few hundred clicks, the pattern FB3 was developed (Function Block). With an FB3 pattern, the pushes of the

```
REGION Push Button Count
   "IEC_Counter_0_DB".CTU(CU:=("CONTACT_A" = 0) OR ("CONTACT_B" = 0),
                          R:="RESET_COUNT",
                          PV:=1000000,
                          Q=>#COUNT_TEMP,
                          CV=>"CLOCK_DATA".COUNTER_PUSH_BUTTON);
    IF ("RESET_COUNT") THEN
        "RESET_COUNT":= 0;
    END_IF;
END_REGION
```

button are counted.

Fig. 15: FB3

Push Button Count: When one or both of the Inputs are zero (0) the value of the counter increases. If the "RESET_COUNT" variable has substance 1, then the counter is eliminated - this variable is then activated from the SCADA, when for example a new button is placed and the experiment starts again.

3.6 Scada - Configuration

The implementation of this experiment requires a reliable user interface, along with the equipment, as well as an automated recording procedure of the results. For this reason, WinCC was used, as it is one of the most reliable SCADA being used within the industrial environment. To monitor the experiment through WinCC, the following were (briefly) implemented:

- Creation of WinCC-PLC communication and definition of the necessary variables (Tag).

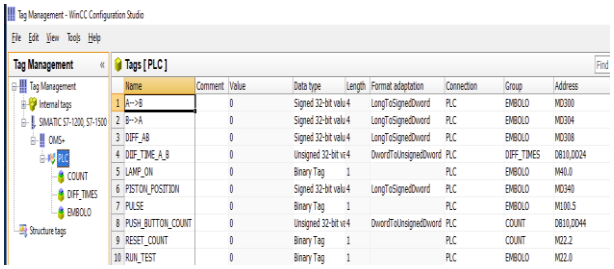


Fig. 16: Tag Management

- Graphical interface development data to display and operate the system.

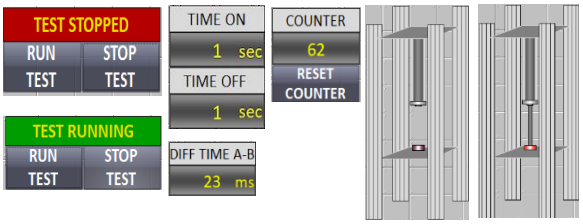


Fig. 17: Screens

- Creation of records in SQL Development to the graphical interface data in order to display and operate the system.

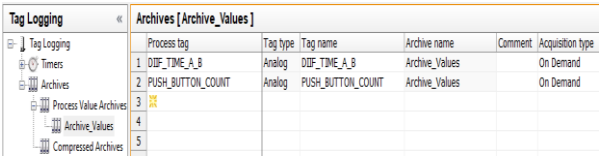


Fig. 18: Tag Logging

The recording of the time difference among the two Inputs is not implemented at a certain time but at the moment when both Inputs are lost from the PLC.

3.7 Input response experiment

For more accurate conclusions, many different trials were tested with this experiment having different factors each time. Each trial tested is analyzed below.

Trial 1

For this trial:

- A new button with two Normally Closed contacts was applied.
- The pressing speed of the piston was adjusted to 10cm/sec.
- The piston is positioned so that it does not push towards the center of the button but towards the edge.
- 3.800 samples were recorded.

The graph below illustrates the contact response of the button.

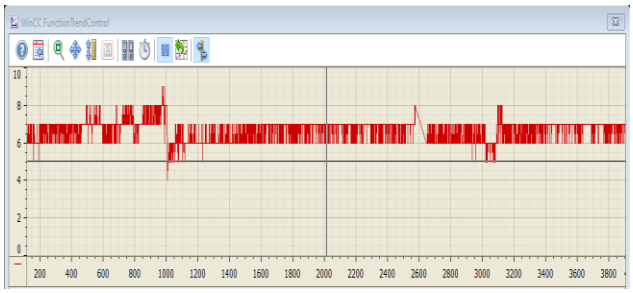
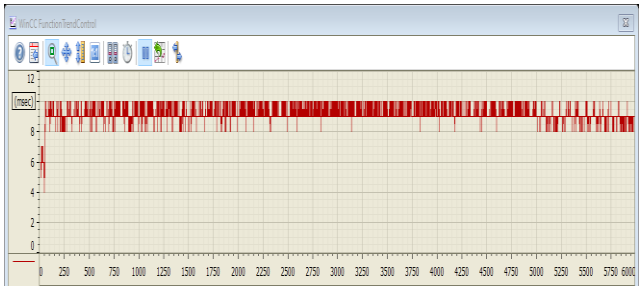


Fig. 19: Trial 1 Report



By studying the graph above, it is observed that the response time is 4-9 msec, with an average value of approximately 6,8msec.

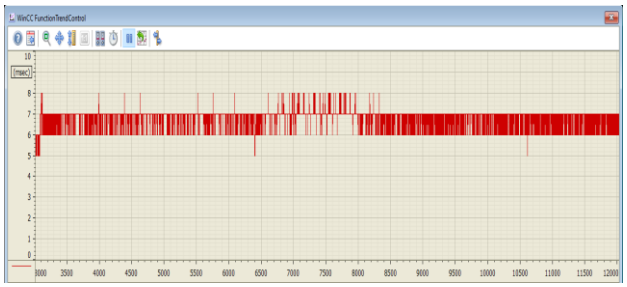
Trial 2

For this trial:

- The same button was used (two Normally Closed contacts).
- The pressing speed of the piston remained stable (10cm/sec).
- The piston was positioned in a way so that it pressed the middle of the button.
- 9.000 samples were recorded.

The graph below illustrates which was the contact response of the button.

Fig. 20: Trial 2 Report



From the graph above, it is observed that the response time is 5-8 msec, with an average value of approximately 6,5msec.

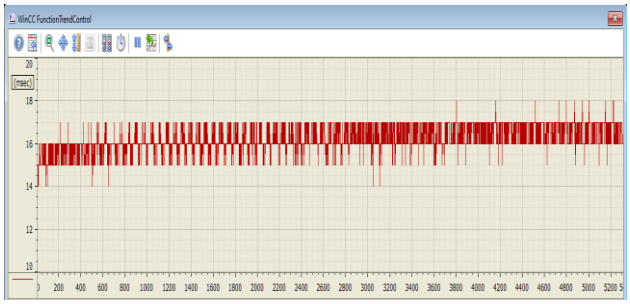
Trial 3

For the third trial:

- A new button was applied with two Normally Closed contacts.
- The pressing speed of the piston was adjusted to 5cm/sec.
- The piston was positioned in a way so that it pressed the center of the button.
- 6.000 samples were recorded.

The graph below illustrates what was the contact response of the button.

Fig. 21: Trial 3 Report



From the graph above, it is observed that the response time is 8-10 msec (except from the first almost 100 samples when the push of the piston was quick), with an average value of approximately 9,2msec.

Trial 4

For the fourth trial:

- A new button was placed with two Normally Closed contacts.
- The pressing speed of the piston was adjusted to 3cm/sec.
- The piston was positioned in a way so that it pressed the center of the button.
- 5.500 samples were recorded.

The graph below illustrates what was the contact response of the button.

Fig. 22: Trial 4 Report

By studying the graph above, it is observed that the response time is 14-18 msec with an average value of approximately 16,3msec.

The results of the trials above are the following:

- The difference in response time of the contacts on a button with a quick press, does not exceed the 10msec and with a slow press it does not exceed 20 msec.
- Even after 13.000 pushes, the contacts have the same bearing response.
- The response time of the contacts does not differ much when the push occurs in the canter or at the edge of the button.
- The response time of the contacts does not differ much when the push occurs for a short period of time e.g. per 2 sec or per 1 min.

From all the various tests which were performed, it is observed that the difference in response time of the contacts is generally stable, even after several thousand presses. It should be noted that by using these elements, like the Stop Emergency, are elements which are not often activated during the producing process but only when we have an issue. It has been observed that even days could pass and these elements would not have been used at all. This means, if a machine has approximately a 10-15 year old life cycle, a Stop Emergency is very likely to be

activated even less than 5.000 times. Of course, for the reliability of the item (e.g. Stop Emergency) its construction materials play a very important role. An item made with good materials endures time, high temperatures, humidity, vibrations etc, which is something that most manufacturers now consider.

4 Conclusion

After the tests were carried out, it is observed that the response of two different contacts of a button can be detected through a Basic PLC, without any additional, special hardware. The response time difference is low, with an average value of 10 msec. If this difference is greatly increased e.g. at 300 msec, it automatically means there is a problem in the contacts of the button. This problem is certainly easy to be detected, so the production can be stopped safely. In most Safety cards the response time difference between the two contacts is adjusted at 500 msec and these contacts monitor this difference with their own machinery in order to remove any possible problems. There are many old automation systems with slow processes that are still operating and due to their structure or their production rate are practically prohibited to upgrade their Safety operation. Today in most cases, those systems are completely replaced by new ones (with a high cost). According to everything mentioned above, it is concluded that by using specific CPU routines of a PLC in automation systems (especially existing ones) increases the reliability of the operation as well as the security, without any further costs, either in money or in implementation time. The final conclusion which emerges from the experiment is that by using Basic PLCs and particular routines, automation function reliability can be increased, without any additional equipment. This means, without extreme costs either in money or in implementation time, that even existing automation systems can have a more reliable and safe operation without being replaced by new systems, as PLC manufacturers recommend.

ACKNOWLEDGMENTS

All authors would like to thank the University of West Attica for the financial support provided to them to undertake this research project.

References:

[1] EN ISO 12100 (Safety of machinery - General principles for design - Risk assessment and Risk reduction).

[2] IEC 61508-3 (Functional Safety of Electrical/ Electronic/Programmable Electronic Safety – Related Systems).

[3] https://www.tuv-sud.co.uk/uploads/images/ 13972201802365442503 95/sil-or-pl.pdf.

[4] IEC 61511 (Safety instrumented systems for the process industry sector).

[5] EN ISO 13849-1(Safety of machinery - Safety-related parts of control systems).

[6] IEC 62061 (Safety of machinery).

[7] S7-1500R/H redundant system, System Manual, 11/2019, A5E41814787-AB

[8] https://www.industry.siemens.com/topics/global/en/safety-integrated/machine-safety/safety-evaluation-tool/Pages/Default.aspx.

[9] https://www.dguv.de/ifa/praxishilfen/practical-solutions-machine-safety/software-sistema/index.jsp

[10] Programming Guideline for S7-1200/1500, ID: 81318674, V1.6, 12/2018.

[11] Working with WinCC, System Manual, A5E45518672-AA, 09/2018.

[12] STEP 7 and WinCC Engineering V16, System Manual, 11/2019.

[13] Tham M.T., Warwick K. “Fail-Safe Control Systems” ISBN 9789401066778.

[14] D. Smith, K. Simpson, "Safety Critical Systems Handbook – A Straightforward Guide to Functional Safety, IEC 61508 (2010 Edition) and Related Standards", 3rd Edition, ISBN 9780080967813.

[15] M. Punch, "Functional Safety for the Mining Industry – An Integrated Approach Using AS(IEC)61508, AS(IEC)62061 and AS4024.1." (1st Edition, ISBN 9780980766004.

[16] H. Hartmann, H. Thomas, E. Scharpf, "Practical SIL Target Selection - Risk Analysis per the IEC 61511

[17] Safety Lifecycle", ISBN 9781934977033.

[18] https://download.beckhoff.com/download/document/automation/twinsafe/applicationguidetwinsafeen.pdf

[19] https://plcopen.org/system/files/downloads/plcopen_safety_part_1_version_2.01.pdf

[20] M. Medoff, R. Faller, "Functional Safety - An IEC 61508 SIL 3 Compliant Development Process, (Third Edition)" ISBN 9781934977088.

[21] Dave Macdonald “Practical Industrial Safety, Risk Assessment and Shutdown Systems”, ISBN 99780750658041.
[22] https://www.leuze.com/media/assets/archive/UM_MSI-T_en_700948.pdf
[23] https://support.industry.siemens.com/cs/attachments/109444336/manual_safety_relays_3SK2_en-US.pdf?download=true
[24] https://download.schneider-electric.com/files?p_enDocType=User+guide&p_File_Name=33003879_K01_000_07.pdf&p_Doc_Ref=33003879K01000.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

E. Theocharis has construct the demo unit and implemented the PLC - SCADA code.
M. Papoutsidakis carried out the simulation and the optimization of the experiments.
A. Sort has organized and executed the experiments.
C. Drosos was responsible for the Reports and the requirement for their repetition.

Sources of funding for research presented in a scientific article or scientific article itself

All authors would like to thank the University of West Attica for the financial support provided to them to undertake this research project.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)