A Probabilistic Approach to PLC Crash Problem

(Full text in English)

Fatima BENNANI ZOHRA[1], Sekhri LARBI[1], Hafid HAFFAF[1]

1Industrial Computing and Networking Laboratory, Computer Science Department, University of Oran 1, Ahmed Benbella, BP 1524 Oran, Algeria

Abstract

We propose in this work, a new approach based on a probability study to choose the standby Programmable Logic Controller (PLC). Indeed, the solution of standby PLC was adopted by the majority of the industrial companies faced to automated production management and scheduling. This solution proved its efficiency by guaranteeing the continuity of functioning of the equipment of production managed by PLC; the latter can be automatically replaced by another in the case of failure. Unfortunately, this solution has shown its limits. Several anomalies have been identified by the automation specialists. The most known are the not starting up of the standby PLC for various causes. The new developed approach allows not only incrementing the number of standby PLC but also choosing the best one in the replacement task. This choice is based on a new probability approach. Simulations are performed through T.I.A Portal platform.

Keywords: Programmer Logic Controller, Supervision, TIA Portal, PLCSim, Protool

Received: September, 01, 2016

To cite this article:
BENNANI ZOHRA Fatima, SEKHRI Larbi, HAFFAF Hafid, “A Probabilistic Approach to PLC Crash Problem”, in *Electrotehnica, Electronica, Automatica (EEA)*, 2017, vol. 65 (1), pp. 45-53, ISSN 1582-5175.

1. Introduction

Efficiency and productivity are decisive success factors in manufacturing and automated management industrial process. The industrial Programmable Logic Controller (PLCs) are the most widespread controllers in automatism that are used in various areas of industry, energy production, agriculture, etc. [1] [2]. Indeed, its success witnessed many application researches in industry which was considerably developed since. Some important required properties are well known in industrial process and have to be verified before any operating mode. Among these properties, we can cite reliability, availability, maintainability, safety and security [7].

By the increasing complexity of the automated systems, and specially in Fault Tolerant Control (FTC) domain, PLC became source of many practical problems; the most frequent is its stop functioning. In order to ensure the continuity of service even in the presence of failure, this problem leads the researchers in automatic control to improve the operating reliability of the PLC. We consider in this work, the system operating safety from the failure point of view domain [20]. The Bayesian methods are used in well-known cause-effect decision methods taking into account statistical data provided by the system [10]

The obvious solution in monitoring is material redundancy of some critical components [3], which consists to realize the same function by different ways. With the same principle, the redundancy of PLC (which merely duplicate the important automaton) is used and the PLC is called standby PLC [2]. Another kind of redundancy is analytical redundancy [20], which uses the information and knowledge provided by the model. In this case, we suppose that mathematical relationships describing components’ behaviour are available.

However, despite the enormous investments in the field of automation research, this solution has shown its limits because the starting up of the PLC is executed automatically when the principal PLC breaks down. In other words, one stand by replacement PLC is associated for each working on-site PLC.

In order to prevent this kind of problems, we propose in this work a new architecture to improve the replacement task. Our proposed solution is based on two main aspects: using probability concepts to encompass the good PLC functioning, and incorporating these non-deterministic data over time to evolve a strategy in choosing the better operating safety mode from the proposed PLC crash mode situations. This methodology proposes a monitoring system, sometimes enhanced by simulation software allowing choosing the most reliable standby PLC to ensure the replacement task.

The study of operating safety of a system is a part of major challenge in the last decade due to its importance in industrial plants, and constitutes performance criteria of the diagnosis system [5]. The evaluation of the system operating safety consists in analysing and estimating the consequences on system performances after subsystems and/or components failures. This study contains three attributes [15]:

Reliability: is the ability of the component or system to perform its required function in a period of time. It is a measure used to assess engineering systems and industrial plants. Formally, we can define the reliability R(t) of a component in [0, t] as the probability of its good functioning noted as: R(t) = Prob {Not failing system on [0, t]}

Maintainability: is the capacity of an entity to be maintained or to be restored in a previous acceptable state in which it can ensure a required function.

The study of operating safety is generally based on [18]:

Functional Analysis: that allows to decompose the system into sample elements and to define their functions in terms of input-output operations. Generally, the functional analysis is defined by external aspect illustrating the relations and the activities of the system with its environment and internal aspect for analysing system activities.

The most used analysis methods are: SADT (Structured Analysis and Design Technique) [23], FAPT (Function Analysis Program Technique) and UML (Unified Modelling Language) used for complex industrial systems [17].

The system dys-functioning analysis consists in identifying the conditions leading to failures and thus, planning their consequences for each failure mode. Some used methods in dysfunctional analysis are fault tree-based, consequence tree-based and the most used is FMECA (Failure Modes, Effects and Criticality Analysis) [19]. FMECA method is well suited for the PLC dysfunctional analysis.

Several proposals were given in literature to compute the operating system safety [14]. Unfortunately, most works were not validated in practice. By the contrary, some simulation software developed in research laboratories has concretized their approaches. Among the proposed software in this domain, we note [19, 20, 21, 24]:

RAM Commander proposed by PHIMECA Engineering is the world's leading software in the area of reliability and maintainability prediction with the fully integrated FMECA and design/process «*Failure Modes, Effects and Analysis*» FMEA.

Open FTA is an advanced tool for Fault Tree Analysis developed by *Auvation advanced software* [13]. It has an intuitive front-end which allows the user to construct, modify or analyses fault trees. This software has been designed to wide international acceptance for fault tree analysis, particularly in the aerospace, nuclear, medical equipment and defence fields.

Protool simulator proposed by Siemens allows to simulate the various failures of the automated systems and to plan the reaction of the PLC to correct them. Based on the history of the occurred failures, the software [19] offers to the automation specialists the possibility of studying the reliability of the system.

To achieve our objective, we developed new Siemens software with new functionalities in simulation.

This approach relies on operating safety concepts of systems to calculate reliability, availability and security. The rest of the paper is organized as follows: In section 2, some critics are addressed to the existing methods. In section 3, we give the general proposed approach through a system design. We can see how the replacement task management has led us to propose a new redundancy based architecture. The last section is devoted to simulation in order to validate our approach. The tests and simulations were performed through a new Siemens tool named TIA PORTAL " Totally Integrated Automation Portal", before ending by some concluding remarks.

2. Limit of the existing solutions

The solution of standby PLC was proposed by the Siemens research laboratory in automatism to remedy the problem of PLC crash, which could engenders the immediate stop of all the equipment involved in the production chain, and consequently leads to economic loss [1, 25].

The research laboratory suggested configuring for each PLC functioning on-site a Standby PLC, which will replace the principal automatism in the case of "stop" state [10].

Two kind of redundancy were given in this context. *Active redundancy* where both "principal PLC and Standby PLC" ensure their automatism tasks, and *passive redundancy* where only one PLC works on-site at a given moment [15, 25].

We also have the redundancy of order 'n' such as 'n' is the number of redundant PLCs that are able to replace the main PLC. These parameters depend on the desired level of security that may be different for different kind of equipment. Unfortunately, in spite of all invested efforts in this domain, and for different causes, crash problem persist in several companies, and affect the capacity of Standby PLC to ensure the replacement task.

This preoccupation motivated us to propose a new probabilistic management approach ensuring the continuity of service even if crash situation arises.

3. Proposed approach

In the new replacement strategy, we assume the replacement of the principal PLC is not programmed in advance. This situation means that the triggering of the standby PLC is not executed automatically when the principal stops. Indeed, we suggest choosing the standby PLC among all PLCs having the biggest probability of good functioning in an interval of time 'T'.

This solution offers two main benefits:

- Increase the number of standby PLC: In fact, we changed the architecture of standard redundancy adopted by the majority of the companies, which defines for each principal PLC, only one standby PLC.
- Make sure that the strategy chooses the best standby PLC among the functioning ones.

To do this, we propose a probability method, which allows computing the probability of good functioning. The standby PLC having the biggest value of this probability is the most indicated to the replace task in crash mode situation.

Let's now, just recall some probability theory concepts necessary to our approach.

4. Methods of probabilistic calculus

Several probabilistic based calculi exist in literature [11, 15]. However, there are two most used tools to model the shelf- life of a system:

Weibull law: Characterize the system performance in its three life phases (youth, constant and ageing) according to the value of parameter B such as [12]:

B <1: period of youth
B =1: constant period
B >1: period of ageing

Three parameters define this law, we have:

Scale parameter n, Location parameter γ and Shape parameter B.

Probability density:

$$f(t) = \frac{\beta}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{\beta-1} e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \quad (1.1)$$

Reliability:

$$R(t) = e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \quad (1.2)$$

Default rate:

$$\lambda(t) = \frac{\beta}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{\beta-1} \quad (1.3)$$

Mean time to failure MTTF:

$$\mathrm{MTTF} = \int_0^{+\infty} R(t)dt \quad (1.4)$$

Exponential law is used often in reliability when failure is constant. This law with parameter λ (λ>0), computes probability of density on IR [16]:

$$f(t) = \lambda \exp\{-\lambda t\}$$

The distribution function is defined as:

$$F(t) = 1 - \exp\{-\lambda t\}$$

The reliability function is defined by:

$$R(t) = \exp\{-\lambda t\}$$

Meantime to failure is defined by:

$$\mathrm{MTTF} = \mathbb{E}[X] = \int_0^{+\infty} R(x)\,dx = \int_0^{+\infty} \exp(-\lambda x)\,dx = \frac{1}{\lambda}$$

Default rate:

$$h(x) = \frac{f(x)}{R(x)} = \frac{\lambda \exp(-\lambda x)}{\exp(-\lambda x)} = \lambda$$

The first law of probability calculus was adopted in our approach for the following reason:

Exponential law is characterized by its 'without memory' principle [11].

Exponential law considers the default rate constant over time [12], which is not the case for the PLCs. In fact, we consider in this work, that default rate of these systems depends on the failure of their critical components, and as they are different from each other, this parameter is thus considered as variant.

5. Standby PLC management

The proposed approach is split into two parts:

- Software parts: The system manages all Standby PLCs.
- Hardware parts propose a new material configuration of all PLCs. Concerning the first part, we designed a system ensuring two main functions: control functioning and the choice of the standby PLC.

5.1. Control of PLC functioning

To ensure the control of each PLC functioning, the system controls the state of functioning of each critical component, i.e., whose failures lead to crash of the principal PLC [25, 28]. Indeed, we distinguish in this work between two types of PLC components:

Critical components associated to the immediate stop of the PLC. These last are considered like non-repairable and are subjected to one failure. They can be repaired without stopping the whole system.

The criterion of criticality appearing in the FMECA analysis [19] [22] is used to determine the list of critical components. Indeed, the principle of this method is based on system decomposition (Figure 1).

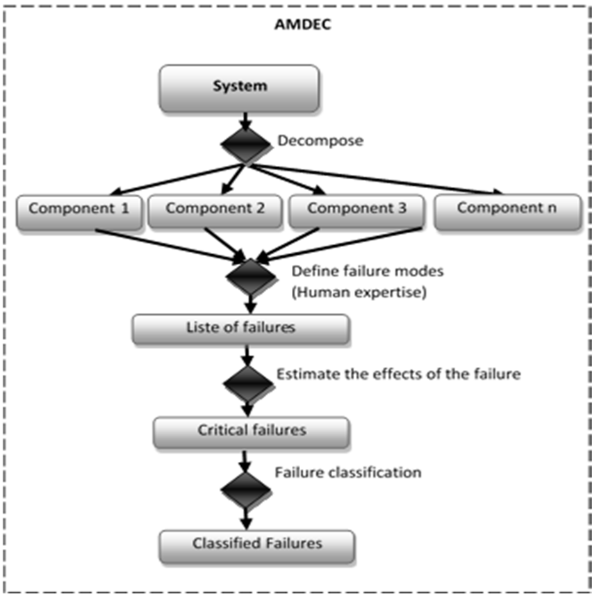


Figure 1. FMECA principles

For each element, we look for malfunctioning or for the possible failure modes. Their effects or consequences of this dys-functionning are analysed through their criticality level. The main purpose is to determine the importance of each failure mode by listing the involved critical components of the system under study.

We calculate for these components, and by statistical measurements, the time before the transition to failure state [8,9, 21]. We use the following formula:

$$MTTF=MTBF \quad (2.1)$$

MTBF = Mean time between failures MTTR: Mean Time to repair

The formula used for calculation of reliability is defined as follows:

$$MTTF=MTBF+MTTR \quad (2.2)$$

So, MTTR is negligible and considered as null: MTTR=0

We have:

TBF=number of hours of good functioning/Number of failures (2.3)

Since these components are irreparable and undergo a one failure, so the number of failure is equal to 1, which amount to proving that MTTF=MTBF= Number of hours of good functioning.

The second functionality supported by the proposed system is the choice of the standby PLC.

5.2. Choice of standby PLC

The system chooses the standby PLC having the biggest value of good functioning probability [11]. It means that the replacing function is not any more executed automatically by the standby PLC when the principal PLC broke down. Indeed, the computing of the probability of good functioning of the PLC boils down to calculating its reliability function [10].

In our case, as argued before, we assume the system follows Weibull distribution law. This hypothesis is well accepted in engineering systems in lifetime measurement of the components.

In reality, the calculus of reliability of a complex system depends of the reliability of its components [10, 21]. In this step, no-critical components (whose failures decrease the reliability of the PLC) are not considered. However, the good functioning probability of the PLC during the failure of its critical components is null because it leads directly to the shutdown of the PLC.

If we adopt the formula of calculation of reliability's Weibull law [12], we have:

$$R(t)=e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \quad (2.4)$$

The calculi of the reliability R (t) from the formula (2.4) does not give exact values because the shape and scale parameters are estimated by experts, and then this may distort the choice of the standby PLC. To avoid this constraint, we calculate the reliability from the more realistic observations. These observations are realized during all standby PLC' life concerned by the replacement operation in period T. We consider T as the time of the principal PLC stopping [10, 14].

We calculate for each standby PLC for a period [t1, t2] / t1, t2 <T time of functioning before failure of the principal PLC. The following measured parameters are presented below:

Table 1. Calculus of reliability

Functioning time T	t1	t2	T
Number of the components which underwent a failure	X1	X2	X3
Number of the components functioning at the beginning of period	Y1	Y2	Y3
Reliability R(t)	X1/ Y1	X2/ Y2	X3/ Y3

The system realizes afterwards, a comparative study of the various values of the reliability for the existing standby PLC (3 PLCs in our application). Indeed, the chosen emergency controller PLC is characterized by an increasing value of reliability meaning that the number of its damaged components decreases.

6. Proposed material configuration

The architecture of redundant automatons belongs generally to two categories: active and passive redundancy as defined in section 2. The switch towards the standby PLC will be automatically done during the breakdown of the principal PLC.

In this work, we consider the first category. The latter is the most used strategy in industrial plants; each on-site functioning PLC is also configured to work as standby PLC, i.e. is a potential future principal PLC.

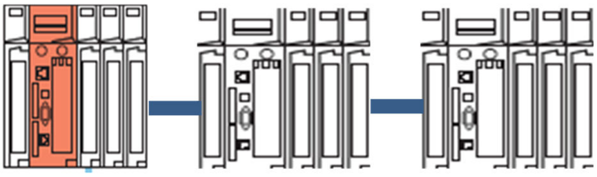
- The main functions of the standby PLC are:
- Acquiring input in real application cluster (RAC) of the principal PLC in its input memory. B)
- Registering output in RAC of the principal PLC in its output memory.

Getting back cyclically the information on the state of the principal PLC functioning.

The proposed approach in this paper will replace a classical serial architecture by a star architecture, exactly like star network architecture:

- Connection between principals PLC and standbys:
  - The configuration of the architecture of standby PLC adopted by the majority of companies defined a connection between a principal PLC and its first replacement PLC in the list (Figure 2, *infra*) [4].

    This last is connected to the second replacement controller and so on in such a way that the PLCs are beforehand classified into an emergency order. The number of the connected standby PLC depends on the required security level, and the sequence of the machines is already established.



Principal PLC Standby PLC1 Standby PLC2

Figure 2. Architecture of standard standby PLC

We suggest here modifying this architecture by connecting the principal PLC to all standbys PLC (Figure 3).

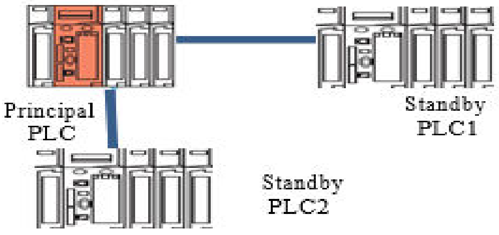


Figure 3. Star architecture of PLCs

This proposal allows increasing the number of the standbys PLC [5], and is better in case of a principal crash because, in each step, an "***elitist***" strategy is ready for the best PLC choice. This method is inspired from meta-heuristic strategies.

We also suggest creating a new connection between the designed system and all PLCs, allowing data exchange on the PLCs functioning state (Figure 4).

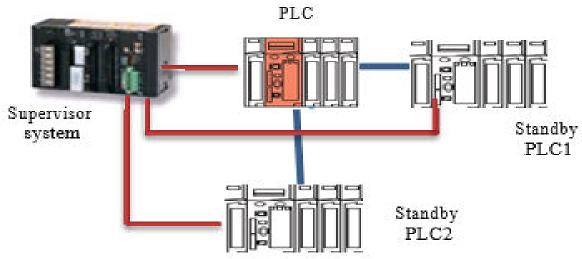


Figure 4. New proposed architecture

In this architecture, the supervisor is connected to all PLC including principal one. Its role is to collect information from the operating PLCs modes in order to evaluate continuously PLCs reliability and order emergency PLC to replace the principal one in case of failure.

7. Case study

This study was realized in Sonatrach Company, GP2Z complex (Arzew- Algeria). The complex ensures the treatment of the GPL raw product forwarded by the deposits of the South through pipelines for the commercial production of the propane and butane. Once treated, products are stored in storage tank.

In this study, we are interested by the architecture of standby PLC adopted by this company. Each PLC functioning on-site controls and ensures the good functioning of equipment of production [20]. For the compressor system, the company possesses three PLC controllers (Figure 5).

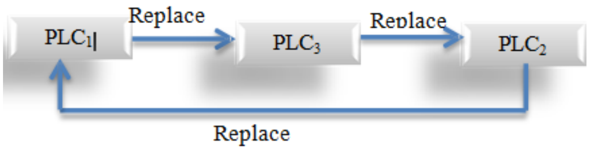


Figure 5. Architecture of standby PLC in GP2Z site System design

The redundancy adopted in this site is an active type: each of the three PLCs has simultaneously the role of principals PLCs and the role of Standby PLC.

System design

The first objective of system design is to propose a global and practical control of functioning of all PLCs. The developed program is divided into two main parts:

PLC Control: allows controlling the functioning of the PLC from the information provided by the state variables of operating of each of critical components. The function of control is based on the "set point" parameter in order to compare the real state of the component (Information given by the PLC) with its in normal behaviour. The list of the critical components, which we defined by applying the principle of the method of FMECA, contains five components. These components are considered as irreparable meaning that their failures will immediate stop the PLC.

Power block: the value of the normal state of energy consumption is 24 V. We declared in the program a variable Bl=24 V. This variable will be compared with the real value of the block sent to the supervisor. If this value is different to 24, the system declares the failure of the power block and starts the execution of the second part of the choice program of the standby PLC. **Processor**: there is no set point value, i.e. we cannot detect this failure until the shutdown of PLC occurs.

Input/output cards: this component is used to send the orders emitted by the PLC to the equipment of production or to receive information from it. Depending on the type of the input/output card, we defined a range of values could be transmitted in the Interval [min, max]. The presence of a fault is detected at monitoring level that determines whether the process is in normal operation or not.

Program: The state of functioning of the program is defined by its cycle of execution. If it exceeds its Nominal value, the program is declared in failing mode. The supervisor system plans the stopping of the PLC, if it detects one of the errors cited above, and then jump to starting the execution of the chosen PLC procedure.

PLC Choice: this part of the program allows calculating the reliability of PLC for the period T. This period has been evaluated during months; we have considered also the date of the beginning of functioning five months before the first appearance of failure.

The calculation of the reliability of PLC boils down to calculation of the shelf-life of its no critical PLC components and which can be easily corrected.

To validate our approach, we have chosen in this paper the case study with three PLCs of the company. We have then noticed the observations of failures as follows:

- In date of May 22, 2016 at 23:32, PLC2 is put in stop after a failure detected at the Power block: the value of energy distribution decreases to 22 V, which is less than 24V threshold.

- At 23:33 of the same day, PLC3 Tried to read the data of the sensors of PLC2 to ensure the task of standby.
- At 23:34, an error detected on the panel of PLC3 indicating the percentage of occupation of its memory at 98 % rate.
- At 23:35 the equipment of production managed by PLC3 is stopped.

We can conclude that PLC3 is not able to ensure the task of standby. To remedy this limit, the principle of the proposed approach consists in increasing the number of the automatons which can ensure the standby task including even PLC1.

The occupation of the memory of PLC3 at 98 % decrease its reliability what returns us to choose PLC1 instead of PLC3 to replaces PLC2.

To validate this choice, we assumed the failure of the components of PLC1 and PLC3 during five months before the appearance of the first PLC2 failure.

Table 2. Calculation of reliability of PLC1

Time of functioning	Month **5**	Month **6**	Month **7**	Month **8**	Month **9**	
Number of the components which underwent a failure	7	2	5	2	15	
Number of the components functioning since the beginning	15		14	11	9	7
reliability R(t)	7 /15= 0.46	2/14= 0.14	5/11= 0.45	2/9= 0.22	15/7= 2.14	

The number of the components, which underwent a failure, differs from one moth to another.

The total number of the components of the PLC in this application is 23 Components.

We can thus plot the reliability curve of PLC1 during this period as follows:

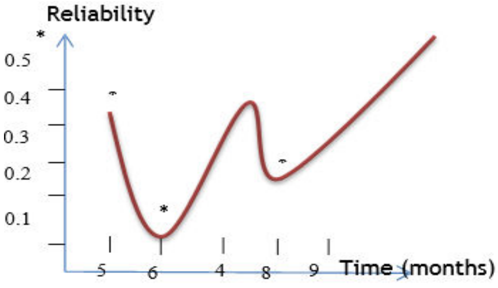


Figure 6. Calculation of reliability of PLC1

As the reliability of PLC1 is the valuable sum of reliabilities of its components, we have:

R(t)PLC1 = R(t) component

R(t)API1 = 0.46+0.14+0.45+0.22+2.14=3.41

In the same manner, we calculate the reliability of the PLC2 for the same period. We obtain the following table:

Table 3. Calculation of reliability of PLC2

Time of functioning	Month 5	Month 6	Month 7	Month 8	Month 9
Number of the components which underwent a failure	9	3	6	5	2
Number of the components functioning since the beginning of period	14	11	9	8	5
Reliability	9/14= 0.64	3/11= 0.27	6/9= 0.66	5/8= 0.26	2/5= 0.4

We can thus plot the reliability curve of PLC2 during this period as follows:

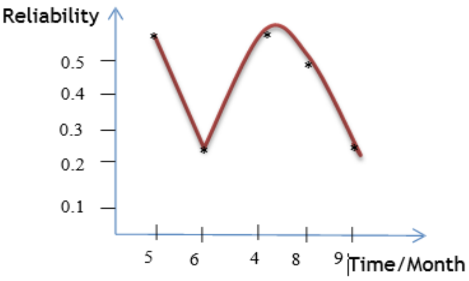


Figure 7. Calculation of reliability of PLC2

As the reliability of PLC2 is the valuable sum of reliabilities of its components, we have:

R(t)PLC2 =0.64+0.27+0.66+0.62+0.4=2.59

The principle of "PLC Choice" program consists to take the maximum reliability value of the existing PLCs. Indeed, we remark from the example that during the period of five months before the appearance of the failure at PLC2, the PLC3 has suffered from many failures compared to PLC1 that explains the degradation of its reliability value.

Figure 8 and Table 4 show a comparative study between the architecture adopted by the company and that we proposed.

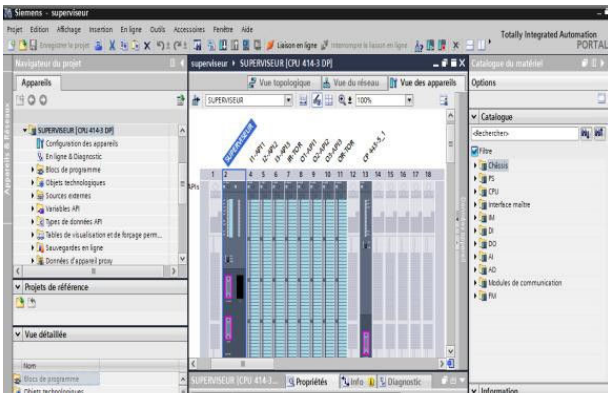


Figure 8. Material configuration of the supervisor

Table 4. Comparison of the architectures

	Architecture of the company	Proposed architecture
Chosen PLC	PLC2	PLC1
Failure risk of the standby PLC	Yes	No
Breakdown risk of functioning of the production equipment	Yes	No

Test and simulation

The system is deployed on a station Console of Programming and Simulation “CPS” [9, 27] (Figure 9).

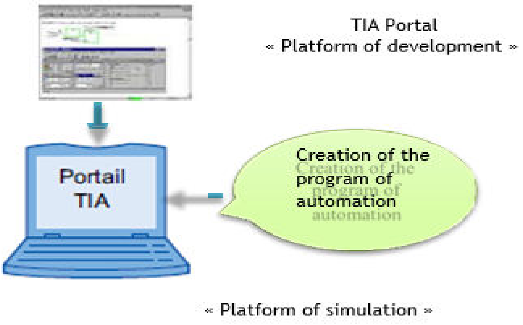

TIA Portal
« Platform of development »
Portail TIA
Creation of the program of automation
« Platform of simulation »

Figure 9. Station CPS

It allows creating a virtual environment identical to the industrial plant. It offers helpful toolboxes to try different situations and faithful failure conditions in order to see how in real situations, the system will react.

Two software tools are installed on the station: TIA Portal and its simulator (Figure 8, *supra*).

The environment of development of TIA Portal “Totally Integrated Automation Porta” is a new working environment of Siemens, which allows implementing solutions of automation [6, 17, 18, 27].

This tool is used in design of the proposed Supervisor system, which in turn is constituted by material part and software part.

The material part presents the configuration of the system and its components. The software tool depicts it by the “material view” (Figure 10).

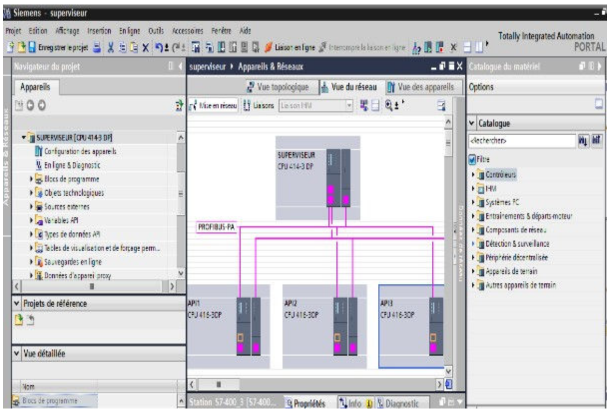


Figure 10. Configuration of connections

The right panel allows us to insert the system components such as Power block, Input/ Output cards or communication modules.

The inserted input cards allow transferring the information on the state of functioning of each PLC. For this purpose, we created a variables table to store data associated to each input system.

The first objective is to make the environment of simulation identical to the real control plant. Hence, we configured the material of three PLCs by choosing the same type of its components accordingly to the existing production installation [6, 29].

To configure the communication between the supervisor and three PLCs, we have used the “networks view” (as it is shown in Figure 10, *supra*).

We have chosen the same type of connection used in the GNL Company, i.e. the “Profibus-DP”, which is also the most used protocol in the industry. This type of connection allows a high throughput exchange of data between several physical devices and automata.

The software part represents the program executed by the supervisor. Three languages can be used by this tool [19]:

- Statement list (STL),
- Ladder Logic (LAD),
- Contact (CONT).

We used the third language because the program of the three PLCs has been already developed by the company.

“PLC Control” is one of the most important parts of this program. It allows controlling the functioning of each PLC from the stored data in the table of variables. Note that the program is divided into simple units; each unit treats a variable of this table (Figure 11).

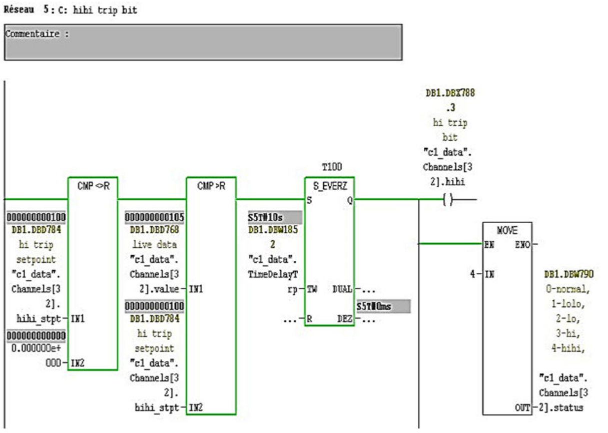


Figure 11. Treatment of state variable

The program compares the value of the variable measured with the set point. If it exceeds the value which is considered as a threshold value, an alarm is activated informing the PLC owner that this state variable is shutting down.

In this case, the second part of program “Choice-PLC” is called.

We created another table named “calculation-reliability” which stores monthly the reliability calculated by each PLC program. The data of both tables are stored in a block called Data Bloc “DB”.

We created also two counters, which accumulate the “Number of the components, which underwent a failure” (Figure 12) and “Number of the components functioning since the last initialization time”. The latter parameters are reinitialized monthly after the calculation of the PLCs’ reliability.

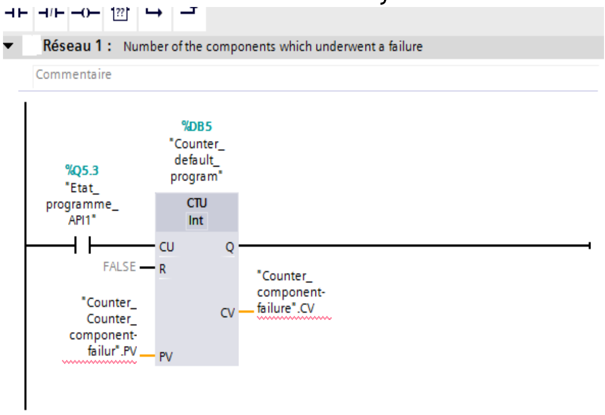
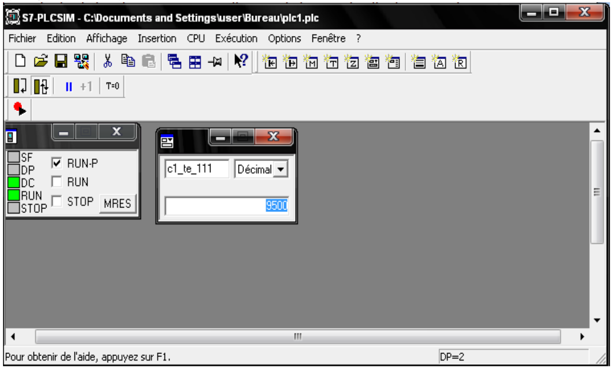

Réseau 1 : Number of the components which underwent a failure
Commentaire
%DB5
"Counter_default_program"
%Q5.3
"Etat_programme_API1"
CTU
Int
CU
Q
FALSE
R
"Counter_Counter_component-failur".PV
PV
CV
"Counter_component-failure".CV

Figure 12. Calculus of number of components that underwent a failure

The second software we used is simulator TIA in order to simulate the functioning of the designed system [17].

From its principal interface, we can allocate values to the stored variables (in the tables) of the PLCs, we can intervene to change these values during the execution, and see how the system reacts (Figure 13).


S7-PLCSIM - C:\Documents and Settings\user\Bureau\plc1.plc

c1_te_111 Décimal

9500

Pour obtenir de l'aide, appuyez sur F1.

DP=2

Figure 13. Simulator Protool interface

For example, we have simulated the following situation: provoke a failure in the PLC1 by affecting a value of 26 V in its state variable in the power blocks (nothing that the normal value is of 24 V).

The PLC2 detects a degradation of its reliability when allocating a big value to the variable “Number of the components which underwent a failure”

Finally, we noted the following scenario:

- detection of the failure by the program;
- execution of the block of “PLC Choice”;
- choice of the PLC3 to replace the PLC1.

8. Conclusions

The proposed approach in this paper offers a new solution to the architecture of standby PLC adopted by industrials companies.

Indeed, the not starting up of the replacement standby PLC during the failure of the principal PLC leads to many economic problems when the production system breaks down. So, it is necessary to ensure good functioning of the whole system including all the PLC machines. When the master PLC crashes, a replacement procedure has to decide which PLC becomes the new “Head” PLC. This decision task is done by the designed supervisor system.

This supervisor, when a failure of a principal PLC occurs, switches automatically to the standby PLC having the highest value of probability of good functioning. This kind of reliability analysis approach has been validated through simulation results and performed on real industrial application.

In the future works, we can investigate other theories proposed in the literature for the calculation of reliability, and should apply this method to another more complex systems.

9. References

[1] Laroux H. and Roussel M., “Algebraic Synthesis of Logical Controllers with Optimization Criteria”. 6th *International Workshop on Verification and Evaluation of Computer and Communication Systems* (VECoS’ 2012), CNAM, Paris, France, August 27-28, 2012.

[2] Guan-Chun L., “Control and Automation”, *Universal Journal*, September 2013.

[3] Hakiki R., Sekhri L., “Hybrid Petri Nets Based Approach for Analyzing Complex Dynamic Systems”. *First IEEE International Conference on Machine and Web Intelligence (ICMWI’2010)*. 3- 5 October, Algiers, Algeria, 2010.

[4] Technical report. “Premium Warm Standby”. Schneider-Electric, 2014.

[5] El Najjar M., Smaili C., Charpillet F. and Pomorski D., *Supervision and Safety of Complex Systems*. ISTE Ltd and John Wiley and Sons. August 2012.

[6] Bennani F.Z., Sekhri L. and Haffaf H., “Supervision Architecture Design for Programmer Logical Controller including Crash Mode”. *International Journal of Information Technology and Computer Science* (**IJITCS**), *Vol. 6, No. 11, October 2014, pp. 10-20.*

[7] Technical Report. “Control Logic Redundancy System”, Schneider-Electric, 2014.

[8] Ghasemi A. and S.Yacout “Calculation of the reliability function and the remaining life for equipment with unobservable states”. *Journal of Mathematical* P. 6079 3A7. 2011.

[9] Bennani, F. Z., Sekhri L. and Haffaf H., “Design of virtual PLC”, *International Conference on Information Systems and Technologies ‘ICIST’2013’*. March 22-24. Tangier, Morocco. 2013.

[10] Bennani, F. Z. *Design of Virtual PLC*. Magister Thesis. University of Oran, Algeria, 2011.

[11] Technical Report. “Need height availability in performance in your PLC”, Schneider-Electric, 2014.

[12] Segala, R. “Testing Probabilistic Automata”, Lecture Notes in *Computer Science*, Vol. 1119, 1996.

[13] Bourguignon, M. and Rodrigo B., “The Weibull-G Family of Probability Distributions”, *Journal of Data Science*, Vol. 53-68, 2014.

[14] Braglia M. “MAFMA : multi-attribute failure mode analysis”, *Journal of Data Science*, Vol. 60-102, 2013.

[15] Seymour B. “MTTF, reliability and life testing”. *Application bulletin* Vol.548-613, 2000.

[16] Technical report, “Reliability, Industrial parasonic”, *Schneider-Electric*, pp. 402-404. 2014.

[17] Merovci, F. and Elbatal I., “The Transmuted Generalized Inverse Weibull Distribution”. *Austrian Journal of Statistics*, Vol. 119-131. 2014.

[18] Gouin, A. and Ferier J.L., “Modeling and Supervisory Control of Timed Automata, *JESA*, Vol. 33, No. 8-9, MSR’99, pp. 1093-1110, November 1999.

[19] Technical report, “PLC Siemens TIA portal”, Schneider-Electric, 2014.

[20] Technical report, “PLC Siemens TIA for simulation program”, Schneider-Electric, 2014.

[21] Technical report, “Redundant system control for maximum availability”, Schneider-Electric, 2014.

[22] Technical report, “Failure modes, effects and criticality analyses”, 2013.

[23] Gourcuf O. Smet D. and Faure J.M., “Efficient Representation for Formal Verification of PLC Program”. In *Proceedings of 8th International Workshop on Discrete Event Systems (WODES"06)*, pages 182-187, Ann Arbor, USA, July 2006.

[24] Philippot A., Tajer A. and Carré-Ménétrier V. “From Centralized to Decentralized Approach for Optimal Controller of Discrete Manufacturing Systems”. *ARPN Journal of Science and Technology*. November 2012.

[25] Bennani F. Z. and Haffaf H., “Conception d’une Architecture de Supervision des Automates programmables Industriels”. *9éme Journées Scientifiques et Techniques (JST9)*, Sonatrach. April 8-10, 2013, Oran, Algeria.

[26] Lin C.T. and Wu S.J.S., “Monte Carlo methods for Bayesian inference on the linear hazard rate distribution, Communications in Statistics - Theory and Methods”. Vol. 575-590, 2006.

[27] Ana S.M. and Henrik O. “Mapping the Structure of Semantic Memory”. *Cognitive Science Society*, Vol. 125-145, 2013.

[28] Carré-Ménétrier V. and Tajer A., “Elaboration of Distributed Optimal Controller for Manufacturing Systems through Synthesis Approach”, International Conference on Communication, Computing and Control Applications (CCCA'11), *IEEE*, Hammamet, Tunisia, mars 2011.

[29] Stanley, S. “MTBF, MTTR, MTTF and FIT Explanation of Terms”. Senior Technical Support Engineer, Vol. 3011, 2011.

Biography

Fatima BENANI ZOHRA is a Post-graduate student for doctor degree for computer science in University of Oran Algeria.

In 2006 she obtained the computer engineering degree in “industrial IT” speciality.

In 2007, she occupied the administrator's post of databaseOracle in Sonatrach Company, Algeria.

In 2008, she assisted the automation specialists in their works in the same company, learnt the basic tools of automatism (Step7, Tia Portal, Protool, Wincc) and understood the functioning of the PLC and the process generally.

Correspondence address: fatima_ing_inf@yahoo.fr

Sekhri LARBI is a Professor at the Computer Science Department of Oran University.

His current research area of interests include formal modeling in distributed and mobile systems, wireless ad-hoc and sensor networks, systems modeling using Petri nets, diagnosability and monitoring of automated production systems.

He is member of the Industrial Computing and Networking Laboratory at Oran University.

He has been a visiting professor at Cedric-CNAM research laboratory, in Paris, France, and Ecole Centrale de Lille (LAGIS) where he worked in Diagnosis of Industrial systems; LIUPA Laboratory at the University of Pau, France; distinguished lecturer at University of Ottawa, Canada.

Correspondence address:
sekhrilarbi@yahoo.fr, larbi.sekhri@univ-oran.dz

Hafid HAFFAF Obtained Doctor degree in computer Science in 2000; is a Professor at the University of Oran Es-Senia (Algeria).

He actually heads the L.I.I.R Laboratory at Computer science department –Oran University.

His researchers concern different domain as Automatic control and diagnosis, optimization, reconfiguration using matroid theory, system of system approaches and their applications in Bond graph and monitoring.

He has many collaborations projects with European laboratories: Polytech lille where he worked in Intelligent transport systems infrastructures- and LIUPA, Pau (France) in the domain of Wireless sensor Networks.

Correspondence address:
haffaf_hafid@yahoo.fr, hafid.haffaf@univ-oran.dz