A Framework for Dependability Evaluation of PROFIBUS Networks

José Carvalho[1], Paulo Portugal[2], Adriano Carvalho[3]

[1]ESTG, IPB, Campus Sta. Apolónia, Bragança, Portugal, e-mail : jac@ipb.pt
[2,3] DEEC, FEUP, Rua Dr. Roberto Frias, Porto, Portugal, e-mail: (pportugal [2], asc [3])@fe.up.pt

Abstract— Fieldbus networks have been assuming a high acceptance in the industrial environment, replacing the old centralized control architectures. Due to time critical nature of the tasks involved in these environments, the fulfillment of dependability attributes is usually required. Therefore the dependability is an important parameter on system design, which should be evaluated.

Several factors can affect system dependability. The environmental ones are the most common and due to the particularity of the industrial environment this susceptibility is increased. In this paper it is proposed a framework based on fault injection techniques, supported by a hardware platform which emulates a fault set, representative of industrial environment scenarios, intending to disturb data communications on a PROFIBUS network. From these fault injection experiments, relevant data is gathered and a further analysis is carried out to evaluate dependability attributes.

***Index Terms*—Fieldbus, Dependability, Fault Injection, Profibus.**

I. Introduction

Nowadays fieldbuses have a prominent role in industrial automation systems with large application domains which extend to almost every area in manufacturing and process industries. They are presently the backbone of distributed industrial control systems, providing a communication infrastructure, which supports the execution of control, monitoring and supervision applications [1].

Dependability attributes like availability, reliability and safety, have become essential parameters on industrial automation systems design. Economic benefits of a dependable system include less lost production, higher product quality, reduced maintenance costs and lower risks [2]. In such systems a dependable behavior can only be accomplished if the communication infrastructure is also dependable.

Industrial environments are characterized by the existence of high diversity of equipments that are sources of large patterns of electrical and electromagnetic interference which can induce faults on existing electronics systems. In data communication systems this could lead to data signal destruction and cause erroneous states at transceivers and other electronics circuits. Most of these systems have fault tolerant mechanisms (e.g. CRC, retransmissions) which are able to cope most of those situations. Nevertheless, there are fault scenarios that cannot be handled, enabling fault propagation, which could lead to a complete system failure.

In this context, an evaluation of fieldbus networks should be performed, enabling the identification of the most important parameters from a dependability viewpoint.

In this paper the behavior of a fieldbus network under fault scenarios is addressed, focusing on PROFIBUS-DP which is a widely used and accepted fieldbus, oriented for communications in small cell networks and designed for cycle data fast exchange with field devices.

The present paper is structured as follows: In section II dependability evaluation context is discussed. In section III it is given a description of the PROFIBUS network. A brief discussion of the related work on dependability, inaccessibility and performance degradation of PROFIBUS is given in section IV. In section V a fault injection framework is proposed, where dependability parameters and performance metrics of the interest to the proposed work are identified. The hardware platform to support this analysis is described in section VI. In Section VII the use of analytical models for dependability evaluation is briefly discussed. Finally conclusions are presented in section VIII.

II. Dependability Evaluation Context

The design, development and operation of a dependable system can be achieved by combined utilization of a set of methods and techniques [3], which can be grouped in the following classes:

- Fault prevention**,** to anticipate the possibility of fault introduction or occurrence.
- Fault tolerance**,** to enable the fulfilling of the system service in spite of faults.
- Fault removal**,** by reducing the number and severity of faults.
- Fault forecasting**,** to estimate the present number, future incidence and consequences of system faults.

Dependability evaluation is a necessary process to verify the conformity of the system behavior with its specification. To certify system dependability it is necessary to submit it to a validation process. This process can be performed through two viewpoints: verification techniques and forecasting techniques.

Dependability verification techniques are based on fault injection methods. They consist on an accomplishment of a set of controlled experiments, where faults are intentionally injected into the system in order to analyze its behavior related with fault conditions. These faults emulate unexpected events that may occur in the system operation and enable either purging design faults – by fault removal, or obtaining data to feed dependability evaluation models – by fault forecasting.

Techniques to perform fault forecasting are developed through assessment of system behavior with respect to fault

activation and occurrence. When this assessment is performed through system modeling, a formal mathematical representation of the system is developed and solved to evaluate system dependability attributes.

Sections V and VII discuss respectively the use of the previous techniques to evaluate PROFIBUS dependability.

III. The Profibus

The PROFIBUS is a fieldbus network oriented to interconnect low level control devices such as Programmable Logical Controllers (PLC), Numerical Controllers (NC) and raw field devices (e.g. sensors and actuators) [4].

The transmission medium is a shielded twisted pair cable. The maximum network length and data rate are respectively 1200m and 12Mb. Both are interrelated.

The PROFIBUS network contains masters and slave stations, also referred as active and passive ones. Active stations transfer their own messages without requesting a remote station. In contrast, a passive station only accesses the bus on request of an active station. This behaviour is accomplished by use of a hybrid access method: a decentralised one according to the Token Passing principle and subordinated centrally according to the Master-Slave principle.

The active stations are responsible for building and maintaining the logical ring. The bus access is gained when a station possesses the token. The token is passed from active to active stations in ascending address mode, except for the High Station Address (HAS) that passes the token to the lowest address active station. So each active station knows the address of its predecessor station (PS) and the address of the next station (NS). The communication process is initiated after an active station receives the token.

The PROFIBUS establishes two priorities for the messages: high and low priority messages. Possessing the token, an active station is allowed to transmit messages during a well-established time (Holding Time). The high priority messages are transmitted first. If an active station receives a token with negative holding time it will be allowed to transmit only one high priority message.

The Holding Time results from arithmetic difference between Target Rotation Time (T_{TR}) and Real Rotation Time (T_{RR}). T_{TR} is a set point that must be into account the time needed for transmitting high and low priority messages and maintenance ones. T_{RR} is obtained from the last two reception of the token by an active station.

IV. Related work

According to application domain, fieldbus networks have to fulfill time constrains imposed by control laws where they provide its communication services.

This problem is firstly addressed to analysis of a pre-runtime communication scheduling conditions, which guarantee that control laws are totally respected by fulfilling the associated deadlines. Nevertheless, the time guidelines for building this schedule are based on performability characteristics of the filedbus system, excluding any fault scenarios [5].

Several analyses of PROFIBUS real-time behaviour are made based on performance protocol features. In [6] a performance evaluation of the PROFIBUS in distributed computer control systems is made by means of an experimental model. The message time delay is evaluated as function of their length, generation time and the Token Rotation Time (T_{TR}) on a PROFIBUS FMS, a subset of the PROFIBUS protocol. A methodology to set the T_{TR} in order to make the PROFIBUS response time to adhere to the real-time distributed applications requirements is presented in [7].

Design of real-time applications can get important benefits from this approach, but a gap on the knowledge of its behavior in faulty scenarios still remains. This fact could lead application designers to assume a worst-case scenario that cannot be necessarily the same one that occurs in presence of faults. In this way it always exist a probability of not fulfilling the system deadlines, which can lead to a system failure.

Most of faults which affect the communication infrastructure lead to inaccessibility problems, degrading the communication performance. The problem of the inaccessibility of fieldbus networks is addressed in [8]. In this work an analytical study of the inaccessibility of CAN and PROFIBUS is presented. The worst-case inaccessibility due to station insertion, station failure and bit error are derived. As suggestion it is considered important the analysis of a general performability of CAN and PROFIBUS by means of a tool, to evaluate the inaccessibility phenomena, under complex fault scenarios.

In [9] the proprieties of PROFIBUS MAC (Medium Access Control) protocol over error prone links are analysed. Several metrics of the ring stability are captured by simulation. The results show high ring instability when the PROFIBUS MAC operates over high error prone medium as the used ones by the wireless communications.

The previous works show that the behavior of PROFIBUS protocol it is fully characterized from the viewpoint of the necessary conditions to achieve real-time communication scenarios. But this behavior when submitted to fault conditions was not yet fully addressed. Therefore, it becomes relevant to provide means to lack this absence by establishing a methodology that enables to evaluate the protocol behavior in those fault scenarios.

In this paper it is proposed a framework based on fault injection techniques, supported by a hardware platform which emulates a fault set representative of industrial environment scenarios, affecting directly data communications on PROFIBUS networks. From these fault injection experiments, relevant data is gathered and a further analysis is carried out to evaluate dependability attributes.

V. Fault Injection framework

A fault injection framework needs to be researched in order to identify and characterize a set of faults representative of abnormal operation of PROFIBUS.

Protocol behaviour can be affected by faults, which occurs frequently either at bus or at transceivers. At the bus, data frames are changed by faults, caused by some kind ex-

ternal factor that leads to inconsistency of the information conveyed. This affects two main groups of frames:

- Token Frames, which are used to manage the logical ring. Faults at these frames could produce errors at token passing and loss of the token that could lead to ring instability [9].
- Action/Reply Frames, which are used to exchange data between stations. Faults at these frames could delay the data exchange process until a maximum number of retries, or to make it impracticable.

At transceivers, faults can be both permanent and transient. Permanent faults are associated to continuous malfunction of a transceiver. Transient faults cause a temporary effect on a transceiver and can be the result of noise or another electrical disturb. Faults at transceivers have high impact on the related station. In order to detect abnormal behaviour of transceivers, the protocol read back bit by bit the token frames transmitted by the station. In case of two consecutive errors on the token frame read back, the protocol assumes a defective transceiver and leaves the logical ring.

For a given scenario the pre-run time scheduling establishes the time restrictions that must be fulfilled by the communication system. The restrictions appear as deadlines, which represent the time bound to complete the communication service. The deadline should consider all times components to complete the message cycle. It includes queuing time and others latencies that occurs at active and passive stations and transmission medium, on end to end communication process. This time must be minor or equal to a bounded time witch is known as the worst-case execution time (WCET).

In this context, the analysis of fault effects on the fulfilment of the WCET on PROFIBUS-DP is an important research topic of the present work, and therefore the WCET fault coverage it is considered an important metric that should be experimentally evaluated. The WCET fault coverage is the probability of the fulfilment of the deadlines imposed by the control laws which support system operation, when this is affected by a given fault.

In order to analyze the impact of those faults on the WCET fault coverage, it is necessary to produce a set of statistical independent experiments, in respect to each one of the identified faults. In those experiments, faults are injected according to a probabilistic distribution established according to real fault scenarios. The number of violations of the WCET is computed for a given fault scenario, and from that the WCET fault coverage is obtained.

To perform the previous experiments a hardware platform is necessary. Although physical fault injection in the bus can be performed be using a simple transceiver probe, it becomes difficult to collect the necessary data to evaluate fault coverage if standard equipment is used (e.g. PLCs), since the access to the protocol stack it normally hidden. Therefore, it was necessary to develop a special node, full compatible with the PROFIBUS protocol and totally instrumented, enabling the gathering of the necessary data.

VI. HARDWARE PLATFORM

In this section the hardware platform to support fault injection on the PROFIBUS network is presented. It is composed by an injector module, a monitor module and PROFIBUS-DP communication nodes. Communication nodes are part of communication infrastructure. Injector and monitor modules compose the fault injection setup (Fig.1).

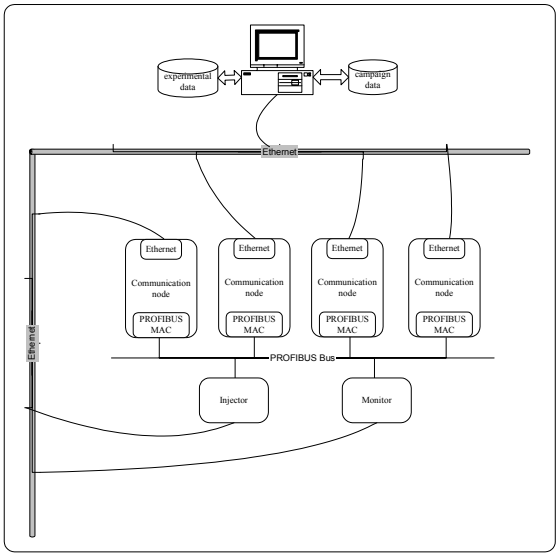

Figure 1- Hardware Platform

Communication Infrastructure

The communication infrastructure provides a test bed for intended dependability experiments. The network is a set of EM277 PROFIBUS-DP slaves from Siemens and masters and slaves based on DSTni-LX-002 microcontroller from Lantronix.

The DSTni-LX is a complete communication single chip solution that features high performance Turbo-186 compatible microprocessor and features a wide range of on-chip peripherals to support the most popular embedded networking technologies. The communication channels includes: Ethernet MAC, Dual CAN, PROFIBUS, SPI, and Dual Serial ports to handle the most demanding embedded applications. DSTni-LX implements the ASPC2 ASIC from Siemens. The ASPC2 is the MAC that carries out the bus protocol of the Data Link Layer (DLL). The interface service and management functions of the DLL are handled by software. The node configuration allows easy access to DLL and user interface, allowing collection relevant data to dependability experiments.

Fault Injection Setup

A capability to inject faults in the communication infrastructure is necessary in order to carry out experiments. This capability is achieved by the use of dedicated modules that are connected to the communication infrastructure. The modules are an injector and a monitor.

The injector is a dedicated module oriented to inject faults on the communication bus. It is composed by a trans-

ceiver witch acts as an injection prove. This transceiver is commanded by a microcontroller that has the role to perform the fault injection activities in a controlled way. The injector can perform either asynchronous or synchronous fault injections. In order to execute a synchronous fault injection it is needed to supply the injector with bus information. This is achieved through permanent observation of the bus activity.

The monitor module has the mission to register all relevant events on the bus, mainly to verify the fault activation and fault effect at the bus. It is based on a transceiver controlled by a dedicated CPU.

Monitor and injector are modules oriented to manage bus injection faults. To inject faults on a transceiver it is necessary to design a node with dedicated hardware. This hardware is then commanded by the microcontroller that sends signals to direction bits to emulate faults either at reception or transmission of the transceiver.

In order to facilitate the management of the experimental platform, an Ethernet connection between all members of experimental platform and a computer host is established. This allows uploading of all information collected on fault injection experiments to the host computer and downloading of fault injection camping files to the injector module.

Fault Injection Experiments

Experiments are performed according to the physical fault injection approach [10]. It consists on the application of physical faults by imposing an electrical level to the physical device - e.g. stuck-at-0, stuck-at-1 and bit flips. The injector module uses the bit flip in order to corrupt the information conveyed in the digital signal. Faults are placed at two levels:

- Bus Level: to emulate the corruption of the PROFIBUS frames due to external factors such as electromagnetic interference. It allows at focusing the corruption of action/reply and token frames.
- Physical Level: with this kind of fault it is possible to emulate malfunctions of the transceiver. These faults are in respect to transceiver function of transmitting and receiving.

For each fault injection experiment, the behavior of the system is observed to determine whether or not faults have any impact on the system. This is done from two viewpoints:

- At the bus, by monitoring the communication bus and verifying whether or not the fault corrupts a message on the bus.
- At the communication nodes, by registering the outputs of the PROFIBUS MAC interface (ASPC2).

At communication nodes the main goal is to verify the fulfillment of deadlines, thus the ending time of a message cycle is the most important information. In order to obtain information about the protocol behavior a set of information supplied by special registers of ASPC2 are collected. The results could be used to sustain conclusions about the fault effects on WCET and other dependability analysis (e.g. latency and dormancy). This includes:

- Detected faults by the MAC fault tolerance mechanisms such as: Double Token, Pass-Token error, HSA-error, Sync-Error, Response error, etc.
- MAC states: Hold-Token, Listen-Token, Not Hold-Token.

Since the experiments are occurring on a distributed system, it is important to maintain the consistence of the collect data. Therefore it is necessary to guarantee time synchronization of all communication modules in relation to the experimental data, and time stamping of the collected events.

Finally all data is collected on the host computer and put available for analysis process, enabling the computation of the respective dependability attributes.

VII. Dependability Evaluation Modeling

Dependability evaluation by using fault forecasting techniques is performed from two viewpoints:

- Qualitative in which are identified and classified the system failure modes according their attributes (e.g. cause, effect, critically, quality of service, etc.) and the event combinations that could lead to a system failure.
- Quantitative in which measures of dependability attributes, as reliability, availability, safety and performability, are evaluated.

Although qualitative and quantitative assessments have the same aim - to evaluate dependability -, they differ significantly in several aspects, such as: (i) application scope, from simple to complex systems; (ii) input data, by assuming distinct assumptions about their attributes; (iii) output data, by producing data with different characteristics and precision degrees; (iv) lifecylce stage employment, from design to commissioning. Therefore, it is necessary to have detailed knowing about the main factors that support the implementation of the assessment methods, to perform an adequate choosing of the most appropriate one [11]. These factors can be presented as following:

- System Structure: It is fundamental to have a complete understanding of the engineering implications of the system. This implies to know how the system operates, to identify the ways in which it can fail (failure modes) and to deduce the consequences of these failures (severity, risk factor, degraded service, etc.). To accomplish these tasks is necessary, from a dependability viewpoint, to know the system structure. This will help to identify the most important system elements, their behavior and respective interactions. From a structural viewpoint, the system is observed as a set of components bound together in order to interact [12]. A component is another system and the recursion stops when a system is considered as being atomic, i.e. indivisible. Another perspective [2], closer from an engineering viewpoint, decomposes the system into four embedded levels: (i) units, if the system is redundant multiple

units are used; (ii) modules, units are built from modules; (iii) components, models are built from components. Both viewpoints define a hierarchical structure, which enables the establishment of several analysis layers, and permits the use of the most appropriate evaluation method at each one. At a layer, dependability evaluation uses data from the previous one and provides data to the next. Some structural aspects should have a special attention, such as: (i) redundancy employment at the different levels, active or standby; (ii) behavior of the fault treatment mechanisms (e.g. reconfiguration when a failure occurs); (iii) component interactions (e.g. dependency).

- Fault Characterization: Faults and their sources are extremely diverse. A current classification [3], shows five main viewpoints, which are: phenomenological cause, nature, phase of creation or of occurrence, limits with respect to the system boundaries and temporal persistence. From a dependability evaluation viewpoint, the most important fault attributes are related with their stochastic nature, such as: (i) inter-arrival distribution, which establishes the time interval between fault occurrences; (ii) temporal persistence, from transient to permanent faults; (iii) correlation, from independent failures, affecting just one component to common-cause failures, affecting simultaneously several components (iv) coverage factors, related with the efficiency of the error processing and fault treatment mechanisms. The previous characterization enables the establishment of a fault model.
- External Actions: System dependability can be improved by repairing a failed component, by performing preventive maintenance or by stocking spares for future repairs/maintenances. From a dependability evaluation viewpoint, these actions are characterized by: (i) repair/maintenance interval, which defines the time necessary to repair/between maintenance actions, of a component; (ii) repair/maintenance scope, local (component) versus global (system); (iii) characteristics (e.g. number, type, etc.) of resources available to perform these actions.

From the previous data, a dependability model that describes the behavior of the components of the system and their interactions is developed. This model can be either, functional, more appropriate to a qualitative evaluation, or analytical, to a quantitative evaluation. In the former case, the model is processed to obtain expressions and values of the dependability measures of the system.

Modeling Techniques

Several modeling techniques to perform dependability evaluation by developing a mathematical representation of the system are available:

A. Basic Probability Theory
B. Failure Mode and Effect Analysis (FMEA)
C. Network Modeling
D. Fault and Event Trees Analysis (FTA, ETA)
E. Markov Modeling
F. Stochastic Petri Nets
G. Stochastic Simulation
H. Fuzzy Possibility Theory

Industrial automation systems dependability is typically evaluated through the use of a methodology that typically requires a quantitative assessment of dependability attributes, such reliability (MTTF – Mean Time to Failure) and availability (MTBT – Mean Time Between Failures) [2].

From a dependability viewpoint, these systems also present complex interactions between system components (e.g. multiple failures, repairs) that must be taking into account to establish an accurate model. Therefore, dependability modeling techniques must be powerful enough to allow the inclusion of these scenarios. Since these models could be complex, the use of simulation techniques could be hard to apply, due to the amount of time needed to obtain results. Therefore, it would be preferable to use analytical techniques since the results are easy to obtain. Nevertheless, it is important to notice that analytical techniques have a more limited modeling power than simulation, which is can be overcome by using simpler, but sufficiently accurate and representative, model.

In this context, the modeling techniques should be restricted to those that provide quantitative assessments and a high modeling power (E and F).

Among the dependability modeling techniques presented, Stochastic Petri Nets (SPN) become in the last decade a widely used framework for the performance and dependability evaluation of various kinds of systems by several reasons: (i) an intuitive description of the system behavior; (ii) representation of complex systems by very compact models; (iii) hierarchical modeling; (iv) a formal basis; (v) full representation of stochastic processes; (vi) analytical and/or simulation solutions (vii) large number of tools available. Therefore, SPNs appears as a very promising and powerful modeling technique to perform dependability evaluation, and the models further proposed will be based on this modeling technique [13].

Since fieldbus networks are mainly presented on industrial automation systems, their dependability assessment methods are inherited from those types of systems.

Dependability Models

Several approaches to tackle fieldbus dependability evaluation have been proposed, through the definition of adequate models.

Some approaches proposes dependability models of the behavior of the Profibus Data Link Layer (FDL) protocol when submitted to several types of faults, affecting communication [9][14]. This methodology has the advantage to establish an accurate fault behavior of the FDL protocol, but imposing some difficulties in using it on a hierarchical modeling (since these are simulation models). In this scenario the use of analytical techniques to get precise models

would be very difficult (or even impossible) due to the complexity of the FDL protocol, unless if simpler, but representative of the most important fault behavior, models are used [14][15].

Other approaches consider the fieldbus as a sub-system embraced into a wider system, in which their macro-behavior, from a dependability viewpoint, is considered [14][16][17]. In this case only the high level fieldbus failures that affect the system are modeled, hiding less relevant details of the FDL protocol behavior. This approach has important benefits such: (i) the global system behavior can be full analyzed, (ii) it results on an analytical model; (iii) less experimental data is necessary to establish a realistic model. The use of a reduced fieldbus dependability model can be full justified by two reasons: (i) it is sufficiently accurate and representative of the main fault conditions; (ii) the huge time scale differences between fault occurrences/external actions and the FDL protocol behavior.

From previous discussion, the use of an analytical model of the latter type (sub-system) is intended to use on dependability evaluation. Experimental data obtained through the fault injection methodology previously described, it will be used to establish the necessary parameters of the analytical model. The model computation will both enable to evaluate a large number of dependability attributes and to identify the most important parameters.

VIII. CONCLUSION

In this paper it is addressed the need for dependability analysis of the PROFIBUS network, and the problems of the inaccessibility and performance degradation when it is affected by faults.

Some discussion on dependability evaluation is structured in order to clarify how to proceed in order to validate models and associated techniques.

A set of dependability parameters and performance metrics are identified. The WCET fault coverage factor within real distributed control systems are discussed as needing to be experimentally evaluated.

To obtain the referred parameters an infrastructure to inject faults in PROFIBUS networks was developed. The infrastructure and the presented methodology constitute the framework to research the PROFIBUS protocol behavior under well defined fault scenarios.

IX. REFERENCES

[1] J. Decotignie and P. Pleinevaux, "A survey on industrial communications networks," in *Annales des Télécommunications*, vol. 48, no. 9-10, 1993, pp. 435–448.

[2] W. M. Goble, *Control Systems Safety Evaluation*, Instrumentation Society of America, 1998.

[3] J.-C. Laprie, "Dependability - its attributes, impairments and means," in *Predictable Dependable Computing Systems*, Eds. Springer Verlag, 1995, pp. 3–18.

[4] EN 50170 European Standard, *General Purpose Field Communication System*, Volume 2, CENELEC, 1996.

[5] E. Tovar, F. Vasques, "Real-time fieldbus communications using Profibus networks"; in *IEEE Transactions on Industrial Electronics*, vol. 46, December 1999, pp. 1241 -1251.

[6] H. Seung, K. Ki, "Implementation and Performance Evaluation of Profibus in the Automation Systems" in *Proceedings of the IEEE International Workshop on Factory Communication Systems*, 1997 pp. 187 –192.

[7] E. Tovar and F. Vasques, "Setting Target Rotation Time in Profibus Based Real-Time Distributed Applications" in *Proceedings of the 15th IFAC Workshop on Distributed Computer Control Systems*, 1998, pp. 1-6.

[8] P. Veríssimo, "How hard is had real-time communication on fieldbuses", in *Proceedings of the 27th International Symposium on Fault-Tolerant Computing*, 1997, pp. 112 –121.

[9] A. Willig, "A Ring stability of the PROFIBUS token-passing protocol over error-prone links", in *IEEE Transactions on Industrial Electronics*, vol. 48, October 2001, pp. 1025 –1033.

[10] J. Arlat, "Fault Injection and Dependability Evaluation of Fault-Tolerant Systems", in *IEEE Transactions on Computers*, vol. 42, no. 8, August 1993, pp. 913 – 923.

[11] B. Billinton and R. Allan, *Reliability Evaluation of Engineering Systems- Concepts and Techniques*, Plenum Press, 1992.

[12] P. Lee, T. Anderson, *Fault Tolerance, Principles and Practice*, Second revised edition, Springer-Verlag.

[13] J. Trivedi,, M. Malhorta, "Dependability modeling using petri-nets, *in IEEE Transactions on Reliability*, vol. 44, 1995, pp. 428–440.

[14] L. Lo Bello, O. Mirabella, "A Fault Tolerance Analysis of Profibus Systems by Means of Generalized Stochastic Petri-Nets", in *Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society*, 1999.

[15] P. Portugal, A. Carvalho, "On Dependability Evaluation of Fieldbus Networks: a Transient Fault Analysis*",* in *Proceedings of the 4th IFAC International Conference On Fieldbus Systems and their Applications*, 2001.

[16] P. Portugal, A. Carvalho, "On Dependability Evaluation of Fieldbus Networks: a Permanent Fault Analysis", in *Proceedings of the 27th Annual Conference of the IEEE Industrial Electronics Society*, 2001.

[17] J. Pimentel, M. Salazar, "Dependability of Distributed Control System Fault Tolerant Units", in *Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society*, 2002.