Blackbox testing methodology for SafetyLON according to the IEC 61508 Standard

Pawel Kwasnowski, Grzegorz Wrobel, Zbigniew Mikos, Grzegorz Hayduk

Abstract—**The paper presents methodology applied for blackbox testing of SafetyLON components. Tests cover both hardware and software and are used to accomplish TÜV certification of certain SafetyLON components. The test specification is also a step in delivering test-bed which can greatly simplify development and certification of final SafetyLON devices. Test cases and testing tools used for testing activities in SafetyLON project are also presented.**

I. Introduction

THE aim of SafetyLON project is to build a hardware and software components that will compose safe system according to IEC 61508 standard. The safe is considered here as reduction of malfunction risk (as much as technically and organizationally possible) of the system or its component. There is a large group of applications (e.g. FLS-FireLifeSafety class of building systems, public transport, medical equipment, avionics, chemistry industry) where it is essential to assure such safetyness (for protecting human lives). The SafetyLON hardware employs redundant, dual architecture (called 1oo2 architecture) and self-checking in many places. On the other hand, software is also responsible for using this dual architecture and also uses redundancy for buffers, frames, CRCs, etc.

This architecture assures high safety level of products based on SafetyLON solutions. The idea is to create ready to use and certified core modules and supporting solutions, which then can be applied in different types of devices and applications requiring safety-related and certified products. TÜV is the institution where SafetyLON is planned to be certified. The step which must be taken before certification is thorough testing for conformity with safety-related standards, guidelines and recommendations. The basis of all safety-related industrial standards is the IEC 61508 standard: “Functional Safety of Electrical / Electronic / Programmable Electronic Safety Related Systems”. The standard defines a lifecycle model (called V-model) for safety-related systems and a measure of safety integrity, which is a probability that safety-related system fulfills the required safety functions on demand under all given conditions within a fixed time period [6]. On the base of the probability of dangerous failure occurrence Standard distinguishes four Safety Integrity Levels (SIL1-SIL4). The Safety Integrity Level can be specified in various ways, depending on how the risk can be estimated (quantitative or descriptive). It was elaborated that the SafetyLON solution must comply with SIL3, where for continuous mode of operation the probability of dangerous failures per hour has to be within 10^{-7} to 10^{-8} range. The standard specifies a set of features (techniques, measures) which have different recommendations (highly recommended, recommended, not recommended or no recommendation) for various SIL.

Important element of the lifecycle model are testing activities (on each step of the lifecycle). The standard describes a set of methods which have to be used for various tests on specific phases of the lifecycle. Among others, black box testing is used, as a part of conformance testing of devices in safety-related systems.

First the paper describes structure of documents used at various stages of testing. The structure arises from chosen standards. Then hardware tests and test equipment are described. Next communication tests are introduced, followed by specification of testing tools. In the conclusion further activities are outlined leading to establishing of test bed and performing of conformance tests.

II. Testing documents structure

Before starting to create test requirements and test documentation it was necessary to select documentation layout. The IEEE 829-1998 Standard was selected. It describes roles, objectives and contents of following eight types of documents. They can be used in three distinct phases of software testing:

1) Preparation Of Tests

- Test Plan
- Test Design Specification
- Test Case Specification
- Test Procedure Specification
- Test Item Transmittal Report

2) Running The Tests

- Test Log

Manuscript received January 24, 2007. This work was supported by the European Commission under Contract No 012611 – SafetyLON Collective Research Project within 6 FP.

Pawel Kwasnowski (kwasn@kaniup.agh.edu.pl), Grzegorz Wrobel (wrobel@kaniup.agh.edu.pl), Zbigniew Mikos (mikos@kaniup.agh.edu.pl) and Grzegorz Hayduk (hayduk@kaniup.agh.edu.pl) are with AGH-University of Science and Technology, al. Mickiewicza 30, 30-059 Krakow, Poland

1-4244-0865-2/07/$20.00 ©2007 IEEE.

- Test Incident Report

3) Completion of Testing

- Test Summary Report

A. Test Plan

This document describes how the testing will proceed. Especially it specifies

- what has to be done,
- to what quality standard,
- with what resource,
- to what time scale,
- risks and how they would be overcome.

B. Test Design Specification

This document describes what needs to be tested. It is based on the documents that come into the testing stage, i.e. requirements and designs specifications. It decides which features of a test item are to be tested, and how a successful test of these features would be recognized. The values to be entered for certain test are not specified here, just the requirements for defining those values are described.

C. Test Case Specification

The tests to be run are created in this document. Test cases specify for each testing requirement:

- the exact input values that are required to execute the test,
- the exact output values and changes of value of the internal system state that are expected,
- any special steps for setting up the tests.

D. Test Procedure Specification

This document describes how the tests are run. It is described here how the tester will physically run the test, the procedure steps that need to be done and the actual set-up to be done.

E. Test Item Transmittal Report

This document specifies the items released for testing. It describes the items being delivered for testing and gives approval for their release.

F. Test Log

This document records the details of tests in time order. It contains the details of what test cases have been run, in which order, and the results of the tests. The test results can be either the passed, meaning that the actual and expected results were identical, or failed meaning and that there was a discrepancy.

G. Test Incident Report

This document reports details of events that need to be investigated. The report consists of all details of the incident such as actual and expected results, when it failed, and any supporting evidence that will help in its resolution.

H. Test Summary Report

This document summarizes and evaluates tests. The Test Summary brings together all relevant information about the testing, including an assessment about how well the testing has been done, the number of incidents raised and outstanding, and crucially an assessment about the quality of the system. The decision whether the quality of the system is good enough to allow it to proceed to another stage is based on this document.

III. Hardware tests

A. SafetyLON hardware

SafetyLON node hardware must fulfill many special safety requirements defined in the specification. In case of hardware damage, power supply voltage loss or any other operation disturbances like network communication errors or hardware self tests errors, the SafetyLON node must reach the safe state mode. The most important thing in safe state mode is setting the node outputs into defined safe state. These requirements cause that the special hardware and software architecture is necessary to make possible the detection of improper node operation.

During the SafetyLON node hardware design, particular attention was paid on power supply and I/O circuits.

SafetyLON node power supply should have the following features:

- wide range of input voltage 18 – 32 V (24 V rating voltage),
- high disturbance immunity,
- ability of sustaining the output voltage for about 10 ms after input voltage drop to switch the node into safe sate mode.

Node's input unit structure allows the microcontrollers to check the correct operation of the node input channel. This structure gives the reliability that input state read by microcontroller (and finally the value of the variable connected with the input) really correspond with the state of the node input. This feature is obtained by applying special test pulses to the node input and reading the state of the input by the microcontroller. Proper operation of the node input unit may by confirmed by comparing the input state read by microcontroller with the generated test pulse.

The output unit structure must allow to check whether the state of the node output is consistent with the microcontroller demand. Additionally the output unit must switch node outputs to the safe state (inactive, low state) when node failure or communication error occurs. To this end the state of each node output is read by the microcontroller by the feedback path and compared with the microcontroller demand. Moreover the output unit is supplied by the transformer controlled by the node microcontroller. In the case of microcontroller program execution failure the transformer output voltage drops to zero, because transformer requires the dynamic control to operate.

B. Power supply tests

SafetyLON node must be protected by certain class power supply from supply voltage disturbances. Power supply must allow to enter safe state mode when it does not guarantee safe operation of the node.

To verify power supply properties the following features will be tested:

- input voltage range,
- input voltage disturbance,
- input voltage fluctuation,
- voltage monitoring,
- input voltage loss documentation,
- node behaviour after power down failure.

Power supply input voltage will be subject of various kinds of disturbances and the node will be monitored if it operates for correct power supply and if it enters safe state (and logs voltage loss) when a power supply is incorrect.

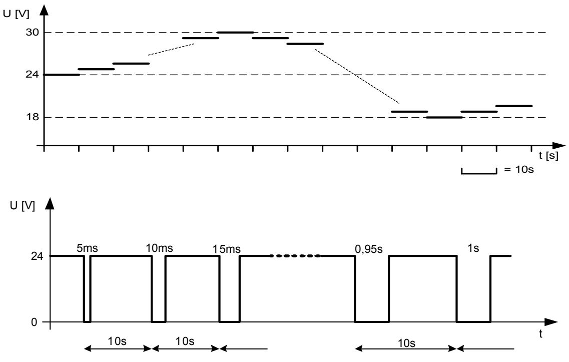

Fig. 1. Examples of power supply input voltage waveforms

Figure 1 shows some of the power supply input voltage test waveforms.

C. I/O unit tests

SafetyLON node is equipped with the special input and output circuits which allows the microcontrollers to check their proper operation. In case of hardware failure of I/O circuits the SafetyLON node must enter safe state mode and write failure into the error log.

To verify I/O unit properties the following features will be tested:

- input signal voltage range
- minimal input signal pulse duration
- input signal change reaction time
- input test pulses repetition rate
- node state after test pulses loss
- test pulses loss reaction time
- input test pulses loss documentation
- duration of output signal pulse
- node state after output circuit fail-safe unit failure
- node state after output circuit test feedback failure
- output circuit failure documentation

Node I/O unit will be subject of various kinds of tests and the node will be monitored if it operates correctly and if it enters safe state mode (and logs these events) when I/O unit failure occurs.

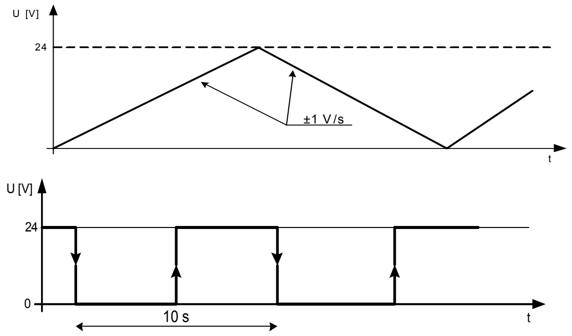

Fig. 2. Example of input unit test signal waveforms

Figure 2 shows some of the input unit test signal waveforms.

Some of I/O unit tests (node behaviour after input unit test pulses loss, node behaviour after output fail-safe unit or test feedback failure) are manual tests. They require human actions to damage I/O unit (for example to break test pulses circuit or fail-safe unit circuit).

D. Hardware tests equipment

Figure 3 shows diagram of hardware test equipment. Supply voltage will be generated by DC voltage laboratory power supply (LPS) controlled with PC. It must be capable to generate supply voltage of various waveforms such as voltage drops of required durations. The laboratory power supply should operate rather as DC power amplifier than voltage stabilizer. A computer program will be written to control the LPS and generate appropriate voltage modulated by certain disturbance or fluctuation. On the other hand, the computer program will be able to remotely monitor the state and error log of the node (via network interface). Both data (voltage and the type of its disturbance generated by LPS and data read from the node over the network) will be recorded into tests database, which then will be a basis for generating test reports and summaries. The database will also contain definitions of voltage allowed for normal operation of SafetyLON node, which also specifies pass/fail criteria for the test. The program controlling the LPS will try to find boundary value where node enters safe state mode (due to incorrect voltage). Following the test run, to judge if node passes the test, voltage parameters from database given as definitions must be equal (or fit into given range) to measured ones.

Static and dynamic properties of SafetyLON input unit will be tested using PC controlled function generator to

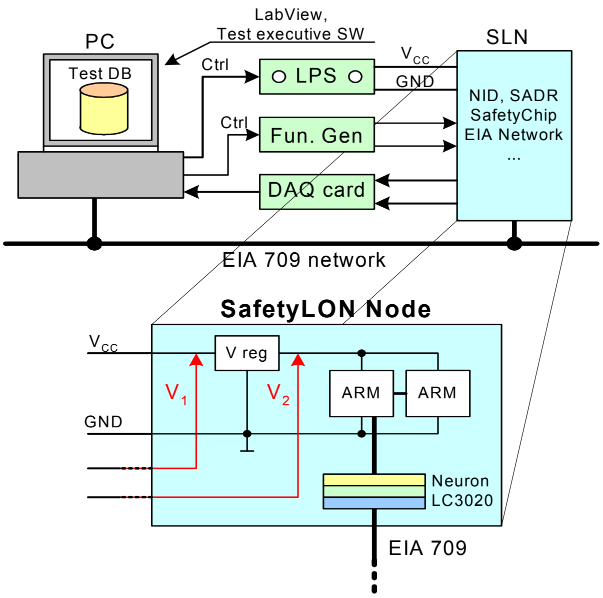

Fig. 3. Hardware test equipment

obtain signals with appropriate waveforms. A program on the PC will be written to control the input signal generator. Additionally the program will be remotely monitor the state and error log of the node via network interface. All data about input signals and about the node state read over the network will be recorded in a test database. These data will be a basis for generating test reports and summaries. The database will also contain the definitions of ranges of I/O ratings needed for normal SafetyLON node operation and specification of pass/fail criteria. The program controlling the input signal generator will find the input signal level changing associated node variable or will measure time between input signal and variable changes.

IV. Communication Tests

A. Tested features

SafetyLON devices communicate according to the producer-consumer model. The communication is tested against the following features being described in Test Design document:

1) recognition and signaling of the communication failures,
2) network transport performance (time resolution),
3) handling of Safe Frame Format,
4) handling of safety irrelevant (unsafe) services in a way that safe ones are not disturbed,
5) timestamps functionality,
6) time synchronization between nodes.

The test stand shown on figure 4 will be used for testing of the above features. It includes Virtual SafetyLON Node implemented on PC for generating intentionally corrupted test messages.

B. Producer-consumer, watchdog

Producer-consumer communication has also watchdogs implemented on consumer side. The watchdog which timeouts, causes consumer to enter safe state mode, therefore break in communication does not imply unsafe operation. Only valid and correct messages may trigger the watchdog. Consequently, equivalence class technique was used as an approach for testing of the communication watchdog: there exists a class of valid (and invalid) messages which trigger (or not) the watchdog. It is tested that as long as valid messages appear (even simultaneously with invalid messages – using fault insertion technique), the watchdog is triggered and the node is in Run mode. Also opposite condition is tested, i.e. when valid messages stop appearing (with or without invalid messages), the node enters safe state mode.

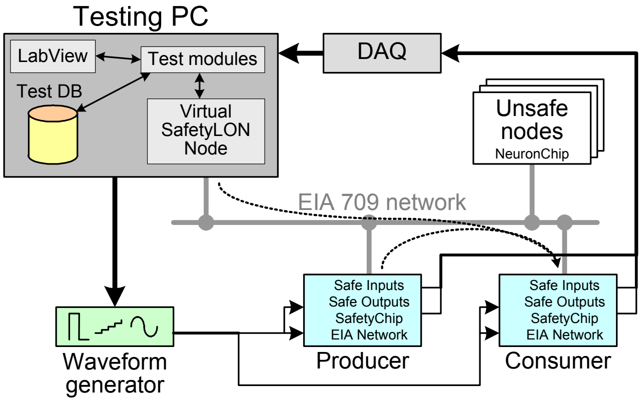

Fig. 4. Test stand architecture for testing SafetyLON communication functionality and network transport performance

C. Failures recognition and recording

Another tested feature is recording of the failures. Some failures when detected are simply discarded or ignored (and no other component of the system can see the failure), and other failures must be recorded, to identify the reason which led to reaching safe state mode. The tests will read error logs while producing conditions that must lead to safe state mode.

D. Network transport performance

Certain elements of the system concern timing aspects of the communication. To test communication responsiveness, other means of signaling events must be assured, i.e. by hardware I/O interface. Producer triggers its safe digital output while sending timing test message and consumer triggers its safe digital output when the message will be received. Time elapsed between changes of the safe digital outputs on producer and consumer is recorded by testing environment.

Communication performance is dependant also on activity of the other nodes coexisting on the same network and sharing communication medium. Influence of those nodes on tested safe components must also be tested.

Testing of network delays is done by stress tests. The delay is measured during externally generated and controlled, but realistic network load. The generated load varies from slight values to heavy load, which is considered as worst-case load testing. Stress tests are divided into two classes: testing under steady (light or heavy) load or testing under bursts of heavy load, which corresponds to case of events avalanche, occurring mostly in alarm or failure conditions, where safety systems must behave 100% predictably and reliably.

E. Safe Frame Format

SafetyLON frame format defines that data is sent redundantly with two subframes, each one having its header fields and CRCs. The frame format (especially CRC fields) is designed to meet certain (high enough) degree of error recognition.

SafetyLON frame format tests are done by sending test messages to the tested node using node emulated on a PC (Virtual SafetyLON Node). The tests are done in several steps. Each step uses technique known as fault insertion testing which generates messages with intentionally corrupted contents. Each intentionally inserted fault tests certain class (or subclass) of frame format faults and tests weather node will behave correctly and i.e. would not lock nor reset. Apart from faulty messages also valid messages are sent and it is tested if node correctly recognized such messages and acts as supposed to.

F. Safe and unsafe services

SafetyLON node might contain an application which is considered partly safe and partly unsafe (i.e. to realize monitoring or logging to unsafe SCADA systems; SCADA stands for Supervisory Control And Data Acquisition). Those unsafe parts may communicate using unsafe communication services and they cannot disturb safe communication. Also, unsafe services may exist on other nodes sharing the same network and they also cannot disturb safe communication. Therefore test cases comprise communication with safe and unsafe services simultaneously.

G. Timestamps

To validate if transmitted data is up-to-date, timestamps are used. Timestamp errors are tested using input partitioning approach. The input value is the timestamp contained in frames and it varies in such way to obtain discarded messages (timestamp is older than the last one or newer than allowed by certain timing window) and correct messages (timestamp fits in the timing window).

H. Time synchronization

For timestamps to work properly, time synchronization must be performed between communicating nodes. Time synchronization tests are forced by testing equipment (by means of I/O) and repeated. Also, accuracy of the synchronization must be measured by means other than network (because it is assumed that network transport may introduce random and unpredictable delays).

V. Testing Tools

As an environment for conducting tests, LabView with TestStand from National Instruments was chosen [3]. It uses instrument abstraction technique and allows to create control-measurement projects that interface various measurement and control hardware. On the other hand it allows also to cooperate with custom software, which is required in the project (i.e. LNS API for communication with LonTalk protocol described in EIA-709.1 standard [4]; LNS stands for LonWorks Network Services).

The test management tool also is considered. An Open-Source package named RTH was selected [5]. It is based on MySQL database and a set of PHP web pages for providing web interface. It is also beneficial that it is easy to interface test database from within LabView environment.

VI. Conclusions

The paper describes testing methodology set up in blackbox testing phase in SafetyLON project. It includes structure of testing documentation and types of test cases which arise from safety standard – IEC61508. At the time of writing of the paper, the test work was at stage of specification and preparation of test equipment. Further work includes programming of test applications for each specified test case in LabView environment, designing test sequences and test reporting in TestStand, test history recording in RTH database and finally execution and documentation of tests of reference SafetyLON devices, (designed and manufactured within the SafetyLON project). These steps will be a part of SafetyLON TÜV certification procedure.

References

[1] International Standard IEC 61508, *Functional safety of electrical / electronic / programmable electronic safety-related systems*, 1998.

[2] IEEE Standard 829-1998: *IEEE Standard for Software Test Documentation*, 1998

[3] http://www.ni.com/labview/

[4] EIA/CEA-709.1-B: *Control Network Protocol Specification* standard is a base for European standard EN 14908-1: *Open Data Communication in Building Automation, Controls and Building Management - Building Network Protocol - Part 1: Protocol Stack*

[5] http://www.rth-is-quality.com/

[6] Workshop. Application of IEC 61508, 13-14 III 2007. TÜV Nord.