12th IMEKO TC4 International Symposium
Electrical Measurements and Instrumentation
September 25–27, 2002, Zagreb, Croatia

TESTING THE DISTRIBUTED SYSTEMS FOR THE HARSH ENVIRONMENT

Jiří Novák, Petr Kocourek

CTU in Prague, FEE, Dept. of Measurement, Technická 2, 166 27 Prague 6, Czech Republic

Abstract – This paper is focused on comparison of three proposed methods for testing the modules and the complex installations of distributed systems (fieldbuses) for the use in a harsh industrial environment and applications, where the system reliability is a crucial requirement. The methods are defined, their advantages and disadvantages are explained and expected test results are presented. Special auxiliary test equipment required for testing is described, especially in terms of its necessary features. Both the test methods are intended for the equipment manufacturers, system integrators and possibly users of distributed systems. The Controller Area Network (CAN) standard is used as an application example, but most of the results are commonly valid.

Keywords: EMC, fieldbus, industrial environment.

1. INTRODUCTION

Distributed measurement, control and data acquisition systems are often used in a harsh industrial environment. Even though most of these systems are based on the standards intended for the industrial use (commonly called fieldbuses), their users are often surprised when they observe decreased throughput and consequent discrete time algorithm problems or the erroneous module or system behaviour. The reason is usually not in the standard itself (exceptions does exist), but in its concrete implementation. The goal of this paper is to suggest and provide the way of how to test the particular modules and the full systems before they are installed at their final location (heavy industry, power production, transportation systems…). The test results can help in finding the potential weaknesses and eliminating them before they cause economical losses. Additionally they help the system integrator to decide which devices from those available have the highest immunity level and are therefore suitable for the target environment.

2. PROBLEM DEFINITION

In a common laboratory or even a light industrial environment level of external electromagnetic disturbances is usually low and it does not have a visible impact on communication, even if seldom-random communication errors occur. The error rate is relatively low and mechanisms intended for solving the error situations doesn't really take place. In the harsh environment with a high level of external electromagnetic disturbances the situation is quite different. The communication error rate is relatively high and error control and solving mechanisms often participate in the communication.

Often the device manufacturers very poorly test the device functionality that cares about the solving communication errors or they do not test it at all. There are two main reasons of such an approach. The first is that the devices are rarely used in a really harsh environment, the second and probably the more important is that manufacturers either don't have the know-how necessary for such testing or they miss the required test instruments. In a harsh environment there is a high probability of the occurrence of the situation, which was not properly tested and its implementation is thus not error free.

The errors in implementation of distributed systems can be found at several levels (in particular protocol layers). They can be divided into two groups:

1. errors causing the violation of the communication standard
2. errors that do not cause the standard violation, but have a high negative impact on communication parameters in terms of data integrity, throughput and transmission timing.

The first group of errors is usually caused by an erroneous implementation of particular device, while the second group of errors usually has its origin in an unsatisfactory interoperability of two or more devices in the system. The border between these two groups is not clear and sometimes the first group error leads to the second group error and vice versa.

3. COMMUNICATION ERROR SOURCES

Of course, all the communication errors have their source in the physical layer disturbance. The immunity of the physical layer implementation is the basic requirement for the device intended for use in the harsh environment. On the other side the physical immunity has no impact on the higher protocol layers from the point of view of the quality of their implementation. It only moves the limit that crossing allows for the higher protocol implementation errors to discover.

In the fieldbus technology the link layer protocol is responsible for physical layer data processing and removing the communication errors. Unfortunately, it is not able to recover from all possible errors and some of them are passed to the higher protocol layers. The typical problems at the link layer are undetected data errors and/or changes in data transmission timing caused by the re-transmissions of data with the detected errors. The probability of the undetected data error depends on the quality of error checking mechanism. In the harsh environment a weak error checking mechanism often passes an erroneous data to the application while a strong error checking enlarges the link frames and thus increases the probability of the frame error occurrence and thus a number of frames that must be re-transmitted. The link layer transmission timing is therefore more influenced. If the communication is based on certain standard, the error checking mechanism is defined and it is just the physical layer immunity, that affect the communication error rate and thus the rate of errors passed to the application layer.

Above mentioned problems can lead to errors in the application layer (incorrect data) and/or in the above lying application (incorrect data, data behind schedule). Sometimes the problems in the communication causes the network management takes place and errors in its data transmission can lead to the total system disorganisation, especially in cases when the reactions on the communication error reporting are not carefully analysed and serviced by the system managing entity.

4. PROTOCOL LAYERS TESTING

The tests presented below can be focused on testing particular protocol layers implementation. The complete protocol stack has not always be tested as there could be some layers that already were successfully tested in other device and there is no reason to expect their different behaviour.

Testing the physical layer immunity is always important as it provides the generic device immunity. High level of the physical immunity can avoid upper layer problems (even if there are some implementation issues in them), low immunity causes the transmission timing problems and increased the probability of undetected data errors at the link layer. Testing the physical layer immunity makes sense especially for particular devices, not for the complete systems.

The link layer test can often be omitted if a standard link layer protocol controller (implemented in silicon) is used. Nevertheless, the wide application of programmable circuits (CPLDs, FPGAs) in today digital designs leads many designers to their own implementation of link layer controllers. These implementations rarely are error free and although their authors should test them, they either don’t have enough time, knowledge or test equipment to run and pass really complete testing. Sometimes the implementation conforms to the standard, but it doesn’t behave in the traditional manner in situations where the standard is not precise enough (or which are behind of the scope of the standard) and issues the interoperability problems. Such controllers often work well in a conventional industrial environment, but in a really harsh environment they can introduce errors that influence not only their parent module but also the entire distributed system. When the link layer protocol is implemented in software (in case of low speed communication), the previous statements are valid as well. The reason why these implementations are more sensitive than those in silicon, even if designer and/or a competent test body has tested them, is quite simple. They are used in much more variable ways than the implementations in silicon. For the FPGA and CPLD designs different programmable circuits (different manufacturers, topologies, speeds) are used, additionally each fitting process of the same design can produce a slightly different behaviour. Similar (and probably worse) situation can be found in software implementations of the link layer protocol (different microprocessors, operating environments…).

The application layer protocol and application algorithm implementation testing is always necessary. The problems are caused either by the software errors (application, error management) or by issues passed from the link layer (transmission timing problems, where data values are not updated in regular or required intervals, undetected errors in data). Additional communication, initiated by application software, which tries to solve the problem, can have the opposite effect. The interoperability issues can also be quite often observed, which occur only under the heavy conditions.

5. TEST METHODS DESCRIPTION

Presented test methods are not new. Some of them are commonly known, others were published e.g. in [1]. The comparison below shows that none of them is sufficient and only their combination can bring the correct and complete results, when the reliable system functionality is required in the harsh industrial environment.

The first one can be called deterministic functional test, the second one the random functional test and the third the immunity test. At first the test set-ups and measurement methods will be specified together with the available test results and their interpretation. Concrete examples based on the CAN will be given, as we have ten years experience with this standard and also all the necessary test equipment available.

5.1. Deterministic functional test

The deterministic functional test can be simply defined as a standard conformity test, focused especially on the device behaviour under the extreme level of communication errors that simulate the harsh environment effects. It can be used for the link layer protocol implementation test as well as for the application layer protocol and application implementation test

To realise the complete link layer functional test, a set of the necessary test equipment has to be available. It must be able to issue well-defined and repeatable test sequences and to observe simultaneously the device response. The second part of the test task is a bit easier, as there is usually several monitoring or even analysing instruments available on the market for any fieldbus standard. Nevertheless, for the link layer test the link protocol analyser is necessary that can detect and capture the link protocol violations. Low cost monitoring devices usually capture only successful data transfers and transmission errors and retransmissions are ignored. On the other side it is quite difficult and often impossible to find a suitable test generator, that can insert all necessary types of errors (by a deterministic and repeatable manner) and still influence only the part of communication required by user. The most difficult work is to find all error types (test vectors) that may occur in order to reach the maximum test coverage. For some commonly used fieldbus standards these tests were already developed and can be used, but the test equipment is still necessary.

Functional test of the application layer protocol is easier than the previous one, as it does not require special test equipment. Standard PC plug-in cards with an appropriate interface and test software can be used. The test should cover such situations that are typical for the harsh environment: inability to send data (for some period of time), required data not received in time, errors in data (application layer protocol and application itself should ignore the data that are clearly bad). The error management functionality should be completely tested (if it is part of implementation). Some of these tests require the special test generator again.

Our experience shows that it always makes sense to pass the functional link layer tests, when the link layer controller is not implemented in the already tested silicon. In some fieldbus standards the part (most) of the link layer protocol is typically implemented in the hardware but a small part also in the software. In these cases the testing of this software functionality is necessary too. The functional tests of application layer protocol implementation and especially the application software are always necessary.

The functional test should always be run with the single tested device. For the application layer test it makes sense to test the device together with other already tested devices in the system (to observe the interoperability issues). In some special cases this makes sense for the link layer tests too (broadcast based communication, e.g. CAN). In this case it is necessary to generate additional load to simulate the in-time data delivery problems.

Finally, the advantages of this test method should be summarised. It is deterministic; it means the test conditions can be defined and the test repeated with the same results (not always, the results also depend on the behaviour of the tested device). It helps in detecting the possible implementation error but especially in finding its reason and fixing it. The main disadvantage is that only those errors can be found for which the test is run. When some feature is not tested, implementation error is not observed and can't be fixed. Another disadvantage of this test is that it emulates the effects of the harsh environment only by the communication problems and the other influences are omitted.

5.2. Pseudo-random functional test

As mentioned above, the deterministic functional test cover just the errors those test vectors are used. Nearly never (with exception of trivial examples) the complete test coverage can be reached especially when not only single device but the distributed system is tested. Pseudo-random functional test should issue further error conditions, which are not covered by the previous test.

To realise such test the similar test equipment is necessary as for the deterministic testing. A standard PC plug-in card with an appropriate interface and test software is used to generate the test sequence. To issue the communication errors, a very simple device can be used that forces one of the signalling levels into the communication path (in the pseudo-random interval, with pseudo-random length and level). For the analysis the same type of the link layer protocol analyser as in the previous test can be used (in case of the link layer protocol implementation testing). A simple monitoring device covers the needs in case of application layer or application test (the same device that generate the test can be used).

The scope of the tests (link and application layers and the application) is the same as for the previous test.

The advantage of this method is the additional coverage of possible implementation errors. The main disadvantage is the randomness, which does not allow the test repetition. The identification of the reason of error is also rather difficult. On the other side the test results (if some implementation error is found) can be used to increase the coverage of the deterministic functional test.

5.3. Immunity test

The immunity test was introduced to check the insusceptibility of particular modules of distributed systems to the external electromagnetic disturbances. The test is strongly focused on disturbance effects that appear in data communication and doesn't care about those decreasing the instrumentation quality of tested devices (e.g. the accuracy of measurement).

Currently valid regulations require for each device on the market to be accompanying with a declaration of conformity. A part of it is devoted to the EMC, particularly to the electromagnetic immunity. Unfortunately the standard immunity tests do not say very much about the immunity of the communication subsystem of the tested devices in terms of the reaction time, total data throughput and so on.

To eliminate this insufficiency the new test approach was developed [1], which focuses on testing the communication parameters under the influence of external electromagnetic disturbances. The block diagram for the single module test is shown at Fig. 1.

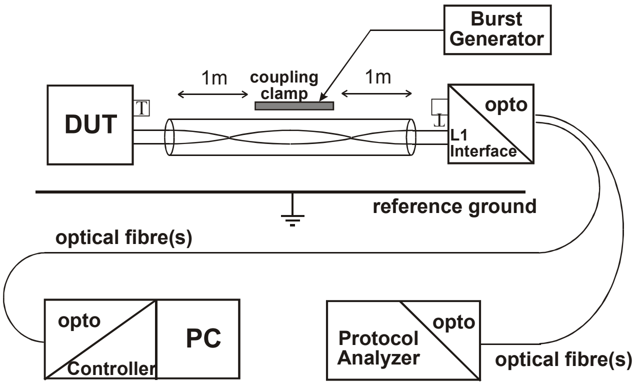

Figure 1. The immunity test set-up

Electromagnetic disturbances are simulated by the burst generator, which produces test signal according to [2]. The optical coupling may not introduce any errors caused by these pulses, they all are therefore introduced by the DUT. The software running on the PC controls the test. An appropriate analyser can be optionally used, which simultaneously monitors the communication. The test results at the link layer (the ratio between correctly and wrong received frames, frequency of errors in the sent frames) and at the application layer (request response times, data throughput) allows easy comparison of particular modules immunity and choosing the best one for final implementation.

The main difference between this and the previous tests is in testing the physical layer immunity, even as it is observed at the link or application layer interfaces. As mentioned in chapter 4, the physical layer implementation provides the elementary device immunity to the external disturbances and cannot be replaced in higher protocol layers. Our experience says that there are quite high immunity differences in particular implementations of functionally same or similar modules from different manufacturers. This fact approves the use of this test for devices intended into the harsh environment.

The biggest advantage of this test is that its results take into the account the actual immunity to the really applied disturbances. The disadvantage is that the test cannot be deterministically repeated and its results (single error events) used to pinpoint the reason of possible failures, as the dependence between the test conditions and particular failures is not close enough. However, other test results like the mean value of the data throughput or mean value of the response time (and their variance) are repeatable and can be therefore used for the device immunity evaluation.

6. CAN TEST EXAMPLES

The realisation of the described tests rather differs for different fieldbus standards. The CAN test examples primarily demonstrates the features of required test instruments and methods of the results evaluation.

The link layer functional test is required only when not tested controller implementation is used. The test generator is the most important instrument (described e.g. in [3]), as it has to be able to generate both the errors in timing (at the bit level) and logical errors. It must be able to transmit frames with required bit timing (different lengths of particular bits) to test the synchronisation mechanism, to support CAN medium access mechanism (CSMA/CA) and issue the errors in it, to generate errors in CRC, bit stuffing errors, frame format errors and their arbitrary combinations. Concurrently the appropriate monitoring and analysis instrument must be available to check the behaviour of tested components and compare it to the standard. The logic analyser with a deep memory and data analysis software can be used, specialised link layer protocol analyser seems to be the best solution. There is an issue for the protocol analyser firmware to distinguish between the errors inserted by the test generator and errors produced by the DUT. In our case the analyser provides an additional input, which is used by the test generator to report the forced error.

As there is several application layer standards for the CAN and many other proprietary solutions, deterministic functional test has to comply with the implemented one. For all of them the reactions on fault confinement status changes should be tested, as there is a big potential for bad implementation, which then takes place in a harsh environment. Also the network management implementation has to be tested with respect to the influence of external disturbances. Another important test should be focused on what happens when the limit values of application variables are exceeded or when the meaningless data are delivered to the application layer protocol. To evaluate this type of test is rather difficult, as user (system integrator) can only observe what happens at the communication interface and module I/Os.

The pseudo-random functional test is implemented in the same way as the previous one, the test instrument described in chapter 5.2 generates the errors. Again it is necessary to distinguish between the errors inserted by the generator and errors produced by DUT on the protocol analyser side.

The immunity test is implemented according to the Fig. 1. If the functional tests of devices has passed, the immunity test is the main criterion for the user (system integrator) for selection the most immune device which causes minimum communication problems. Statistical parameters are measured as mentioned in chapter 5.3. This test is also valuable for device manufacturers, as they can compare the physical layer immunity of their devices with competitors or test the effects of changes in the implementation of communication interface.

7. CONCLUSIONS

The last paragraphs of chapters 5.1, 5.2 and 5.3 summarise the features of particular test methods, their advantages and disadvantages. Only the combination of two or even all three test methods can sufficiently guarantee the required immunity of tested device to the harsh industrial environment and thus its reliable communication within the system.

REFERENCES

[1] J. Novak, Z. Stepka, P. Kocourek, H. Schumny, N. Zisky, J. Neumann, "Electromagnetic Susceptibility of Controller Area Network Modules", IMEKO TC-4 Symposium, Glasgow 1997

[2] EN61000-4-4, "Electromagnetic Compatibility (EMC) - part 4: testing and measurement techniques - electrical fast transient/burst immunity test"

[3] J. Novak, A. Fried, M. Vacek, “CAN Generator and Error Injector”, 9th IEEE International Conference on Electronics, Circuits and Systems, ICECS 2002 Dubrovnik, in press

Authors: Ing. Jiří Novák, Ph.D.[1], Doc. Petr Kocourek, CSc.[2], Department of Measurement, FEE, Czech Technical University in Prague, Technická 2, CZ-166 27, Prague 6, Czech Republic, Phone: +420 2 24352807[1], +420 2 24352190[2], Fax: +420 2 33339929, E-mail: jnovak@fel.cvut.cz, kocourek@fel.cvut.cz