METHODOLOGY OF DEVELOPMENT AND VALIDATION OF SOFTWARE FOR SAFETY-RELATED PARTS OF CONTROL SYSTEMS IN STAGE TECHNOLOGY

MICHAL DRLIK
Institute of Production Machines, Systems and Robotics
Faculty of Mechanical Engineering
Brno University of Technology
DOI: 10.17973/MMSJ.2019_12_2019154
e-mail: michal.drlik@vut.cz

The presented paper contributes to solution of safety assurance in complex mechatronic systems used in theatre stage technology. The aim is to solve the problem associated with reasonably foreseeable incorrect behaviour of persons and operators of these systems by application of safety measures respecting the nature of the specific conditions in which such mechanisms are being used. Special attention is paid to development, verification and validation of software for safety elements of the programmable electronic systems used to control the stage mechanisms. The validation is shown on the implementation of the methodology during development of the functional safety system of the stage technology in the National Theatre Brno.

KEYWORDS
Theatre, stage safety, functional safety, safety integrity level, safety-related software life cycle, reasonably foreseeable misuse, human factor.

1 INTRODUCTION

In recent years, safety of the mechatronic systems used within stage technology has been intensively discussed. Various tricks and visual effects are being created at the stage with the use of the theatrical scenery and properties, lighting and audio-visual technology in the presence of the actors. Such effects are usually realized with the application of mechatronic elements (Figure 2). Heavy properties (up to several tons) that are installed at the theatre stage often move as fast as 2 m/s. Weight of the hung theatre properties range from a few up to several hundreds of kilograms. The theatrical mechatronic systems are very different from the common machinery. During active scenes, the stage space is unclear. Movements of the individual objects take place in the dark and the actors do not always have a chance to react safely and in time. In bigger theatres, the individual properties perform tens of simultaneous movements. Actors and technicians must move under lifted weights or in close vicinity of moving parts or even stand on them. Due to the dynamics and weight of the mechanisms, the eventual injuries may lead to permanent consequences or even fatalities. In the Czech Republic, tens of injuries are reported every year from theatres with outdated stage technology without adequate safety functions.

An important phase of ensuring the safety of the persons present at a stage is the proposal of the safety elements applied in the control systems of the mechatronic devices. The methodology used for development and validation of the safety-related software plays a significant role.

As early as in 1991, Takeshi Nakajo and Hitoshi Kume [Takeshi Nakajo 1991] analysed the relations between the causes and consequences of the mistakes in software development and performed a systematic analysis of the causes and effects of the occurrence of errors in software (see Figure 1).

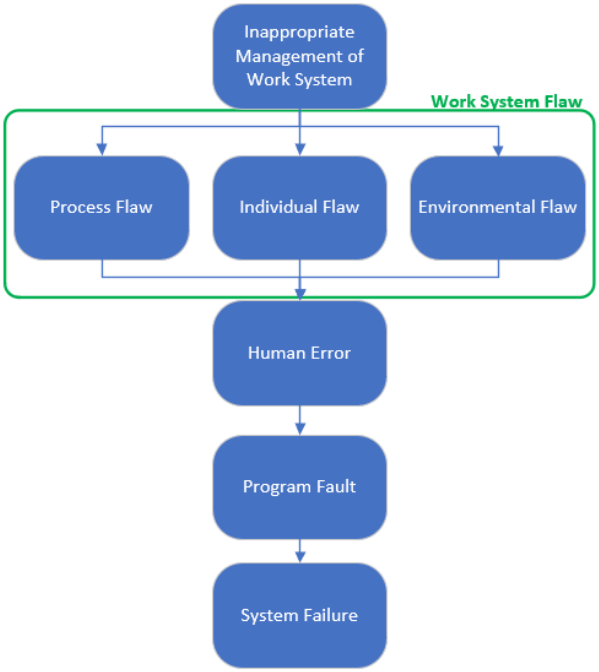

Figure 1. Cause-Effect Process of Software Errors [Takeshi Nakajo 1991]

System Failure – System behaviour mismatched with certain system operation specifications
Program Fault – Incorrect program codes defining software behaviour
Human Factor – Unintended deviation from work standards or targets caused by the carelessness of designers or programmers, such as forgetfulness, mistakes or misunderstanding
Work System Flaw – Inherent characteristics of methods, workers and environments affecting human error occurrence and deviation. Classified into three types: Process Flaws, Individual Flaws and Environmental Flaw
Inappropriate Management of Work System – Inappropriate system managing work system elements, such as methods, workers and environments.

In 2011, several papers were published dealing with risk management during development of machine tools as well as during development of the related safe software [Blecha 2011a], [Blecha 2011b], [Blecha 2011c]. In the same sphere of industry, the virtual reality technology has been successfully used for simulation and development of safe mechatronic systems [Tuma 2014].

However, the current good technical practice confines itself only to specification of the requirements and does not provide any suitable guidance how to fulfil them.

Due to the unique features of each theatre, it is always necessary to specify the requirements on stage technology safety both

from the aspect of reasonably foreseeable misuse by the actors and the technical staff, and the incorrect practice of the development engineers and assembly workers (Human errors). In each theatre, there is a different number of the devices connected to the central control system; therefore, it is necessary to develop new safety software for each project. Identification of hazards and risk analysis performed according to EN ISO 12100:2010 provide an overview of the risks that need to be reduced by functional safety of the stage technology.

Figure 2. Mechatronic systems – theatre

Analysis of errors presented in Safety Critical Embedded System Software published by Kaushik S., Gupta D., Kharb L. and Chahal D. in 2017 [K.V.N.S.L., Kumar S. 2017] revealed that the main source of errors lies in inappropriate documentation of the process of development, verification and validation of software (see Figure 3).

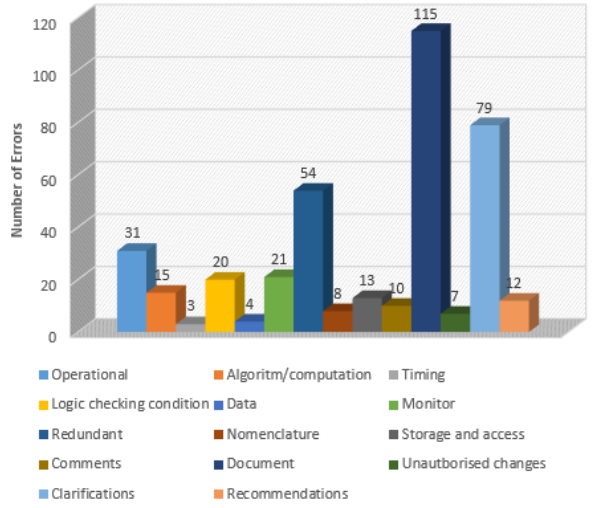

Figure 3. Types of errors [K.V.N.S.L., Kumar S. 2017]

Smruti P. Nanada and Emanuel S. Grant from the University of North Dakota [Smruti P. Nanada 2019] dealt with analysis of errors in development of Safety Critical Systems. Their analysis published in 2019 stressed the significance of the field-specific system approach and of the chosen methodology on reduction of the number of errors in critical systems in applications for:

- Medical systems
- Railway systems
- Automotive systems
- Avionic systems

However, no verified methodology has been published so far that would specify the good technical practice in the field of development and validation of software applications for safety-related parts of the control systems (SRP/CS) in stage technology.

2 CURRENT TECHNICAL PRACTICE

Vast majority of the EU member states have adopted their national technical standards for stage technology. The Czech Republic lacks such a standard and therefore it is necessary to follow the good technical practice of some other EU member state. The German national standard DIN 56950-1:2012 (Entertainment technology - Machinery installations - Part 1: Safety requirements and inspection) is generally considered one of the most sophisticated ones, however, it does not specify the recommended procedure for development of safe software for the control systems.

The benefit of this German national technical standard is in the fact that it defines the potential hazards and dangerous situations from the viewpoint of stage technology. Another important feature is its strict definition of the safety functions that need to be considered in relation with the control systems of stage technology. For these defined safety functions that shall be solved by programmable safety-related systems, it directly sets the required safety integrity level SIL3, if the required functions are programmable. This is the reason why meeting the requirements of this standard is included in the contract proceedings, even at the markets outside the EU.

A new standard prEN 17206 „Entertainment Technology – Lifting and Load-bearing Equipment for Stages and other Production Areas within the Entertainment Industry – Specifications for general requirements (excluding aluminium and steel trusses and towers)“ is currently being drafted and commented; however, it does not specify the procedure of safe software development either. This draft standard adopts the DIN 56950-1:2012 to a great extent and even significantly broadens its scope. The standard practices of occupational health and safety assurance at work cannot always be applied in theatre productions. A typical example is the stage trap (elevator). When the platform is in the position under the level of the stage, the trap creates a dangerous hole that cannot be enclosed by safety railing or marked by some other warning indicators; it is also not possible to equip the actors with safety harnesses and protective helmets (hard hats). Such railing or personal safety equipment would make the performance as well as the whole scenographic and artistic vision impossible.

3 SYSTEM REQUIREMENTS

Due to the size and complexity of the stage technology mechatronic systems, development of safe control system (both hardware and software) requires considerable human, financial and temporal resources. To reach the SIL3 in hardware and

software, it is necessary to employ commercially available certified components together with system approach, in order to keep the costs at such a level that would allow marketability of the whole system. Limiting conditions for successful development of the control system call for the need to equip the safety applications with:

- Certified hardware,
- Certified communication protocol,
- Certified software with limited variability of the programming language.

Thanks to implementation of commercially available hardware and software, it is possible to reduce the time of the control system development; however, it is still necessary to verify suitability of the used components and of the software for the developing safety functions.

The Integrated Development Environment used for safety-related software editing must be certified for the field of functional safety; still, the user must set the processes allowing to specify the requirements on the safety functions, plan the verification and validation procedure, control the changes of the developed software with the use of documented information, set the accesses and authorizations for the programmers.

Elimination of system errors in development of safety-related software within the management of functional safety is realized exclusively by application of the system approach to the development of the subprocesses and to documentation of their realization. For development of safety-related software and hardware, it is therefore necessary to work out a methodological procedure and to select a suitable way of documentation of this procedure that would facilitate significant reduction of potential systematic errors.

4 METHODOLOGY OF SOFTWARE DEVELOPMENT AND VALIDATION

The structure of this methodology corresponds to its incorporation into the life cycle of the overall safety and management of functional safety of software development and follows up on the already elaborated methodological procedures for the other phases within the life cycle of the overall safety of the mechatronic systems used in stage technology. A basic element of this methodology is the well-tested V-model of software development (Figure 4).

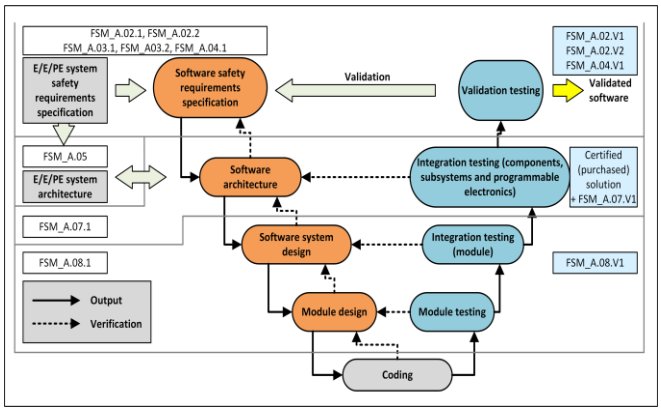

Figure 4. V-model with links to individual documents

In the scheme above, the individual blocks FSM_A.xx represent the essential documentation of the **F**unctional **S**afety **M**anagement, i.e. documentation of the performed development activities. Application of the system approach has proved itself useful for the structure of these documents as well as for their labelling, which helps to keep the **documentation records** well-arranged. The individual documentation groups are labelled with a prefix composed of the letters „FSM_A“ and a two-digit code separated by a dot. From the aspect of system approach to functional safety management, each such labelled block represents a defined closed documentation group. Eventual further subdivision of these blocks is marked with another separating dot. For example, the A.02 block is further divided to A.02.1 and the following document would be labelled A.02.2, etc.

The verification documents are marked with a capital “V” in their block label. Several verification documents may exist for each block depending on the number of the tests performed. Therefore, the letter “V” is followed by the number of the verification document. For example, the first verification document in the A.03.1 block is labelled A.03.1.V1 etc.

Below is a description of the individual steps within the process of safe SW development according to Figure 5.

Step 1 – Safety planning

Plan of the functional safety management describes the activities associated with safety of the whole development process and the documents that must be worked out for this purpose. This block is labelled FSM_A.02. Such procedure meets the requirements on elimination of error occurrence in accordance with the set of standards ČSN EN 61508 ed.2:2011 and the application standards mentioned in the specification of the safety requirements. Within the safety planning module, forms for internal and external simulation of errors are created.

Step 2 – Specification of safety-related requirements

In this step, all the requirements relevant to the safety of the stage technology components are defined, including the requirements from the application technical standards ČSN EN 61508 ed.2:2011. This block is labelled FSM_A.03. The FSM_A.03.1 document defines the specifications of the safety-related requirements.

Step 3 – Validation planning

Based on the specifications of the safety-related requirements FSM_A.03.1, a plan of the validation tests FSM_A.04.1 is prepared, including a list of all the required tests, their time schedule, methods of testing and the responsible persons.

Step 4 – Software architecture

The block labelled FSM_A.07 includes:

- software structure and its interfaces,
- dynamic and static behaviour of applied software,
- potential faults of hardware and behaviour of the software in case of their occurrence,
- verification of the suitability of the Integrated Development Environment and of the pre-programmed blocks.

Software architecture requirements are specified in FSM_A.03.1 document. Document FSM_A.05.1 contains description of the system architecture and defines all interfaces between the components. The system proposal includes the extent of the requirements, detection of errors and corresponding response, subsystem interfaces.

The output document FSM_A.07.1 describes the software architecture. Suitability of the use of the existing safety-related software and its components are described in FSM_A.07.V1 document.

Step 5 – Software system and proposal of modules

The block labelled FSM_A.08 consists in proposal of the individual software subsystems and the corresponding software modules.

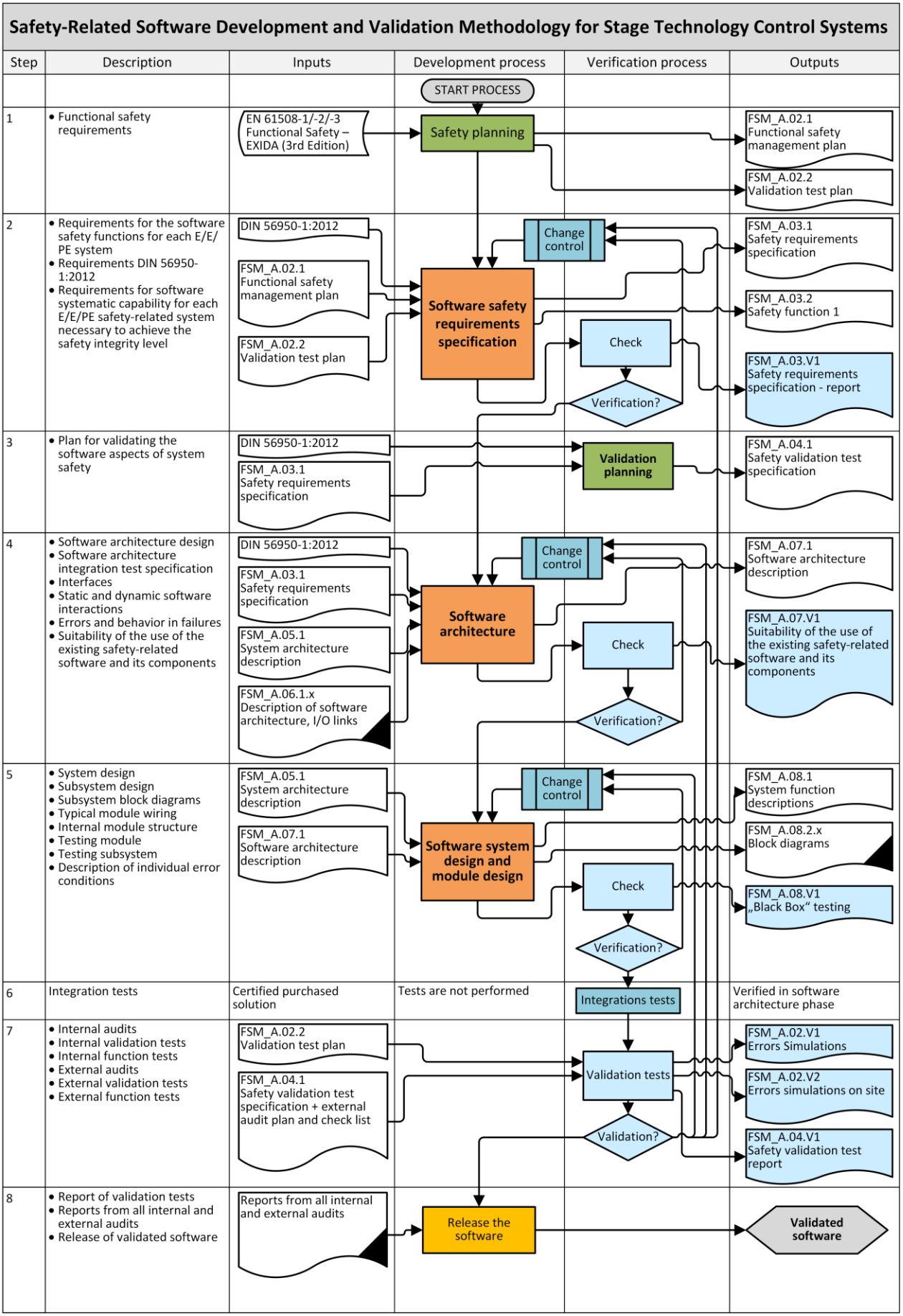

Figure 5. System and process approach to development and validation of safety-related software for stage technology in theatres

Software proposal includes several different integration tests and activities:

- SW – SW integration,
- HW – SW integration,
- subsystems and system integration,
- proposal of modules,
- „black box“ testing.

Step 6 - Integration tests

When certified Integrated Development Environment is used, integration tests at the level of software architecture are skipped as they are already incorporated in this environment. However, verification of the eligibility of the use of this existing software and its components is performed as soon as in step 4 during software architecture proposal.

Step 7 - Validation

Validation is carried out according to the FSM_A.04 validation test specification. The testing includes, for example, testing of the „black box“ subsystems, load testing, error simulation or other techniques. Specific internal and external audits are also performed within the validation step.

Step 8 – Release of valid software

After successful accomplishment of the validations in step 7, the reports from all internal and external audits are checked and based on the positive results of all audits, the finished and now also validated software can be released.

The proposed methodology of software development and validation combines the software development methods with the system and process approach. The proposed methodology is arranged to meet the requirements and recommendations as set by the EN 61508-3 ed.2:2010 standard. For the individual subprocesses (steps) of this methodology, the necessary documentation eventually the suitable tools needed for their realization are prepared. Figure 5 shows the flowchart of the safety-related software development process demonstrating incorporation of the system and process approach into the presented methodology.

5 IMPLEMENTATION OF THE METHODOLOGY IN THEATRE

The proposed methodology of safe software development has been successfully applied during development of the iTEMS control system for control of the mechatronic system of stage technology in the National Theatre Brno (Czech Republic). The control system together with the presented documentation were submitted to independent audit within the iTEMS system certification at the renowned company TÜV SÜD Czech s.r.o. (Figure 6).

Figure 6. Certificate National Theatre Brno and Moscow Kogalym theatre

6 CONCLUSION

The key principle of the presented methodology for safety-related software development and validation is the application of the system and process approach specified by individual steps that must be carried out in order to fulfil the requirements on stage technology safety and, at the same time, to eliminate the human factor impact on the occurrence of developmental or programming errors. This goal is significantly facilitated by the logically structured documentation of all the steps within the development process. The proposed methodology of software development and validation may be applied not only in development of software for the complex mechatronic systems used in stage technology, but in general for all machinery where it is necessary to reduce the risks by application of functional safety associated with development of safety-related software. The methodology may contribute not only to improved safety of the persons present near the stage technology (on stage, in auditorium or at backstage), but also in other entertainment areas, in other fields of engineering and many other workplaces of various specialization within the EU for example theatre in Russia Moscow "Кино-концертный комплекс" Янтарь "-Когалым. Certificate of successful completion of the control system Figure 6.

REFERENCES

[Takeshi Nakajo 1991] Takeshi Nakajo., et al. A Case History Analysis of Software Error Cause-Effect Relationships, August 1991, 0098-5589/91/0800-0830

[Blecha 2011a] Blecha, P., Blecha, R., Bradáč, F. Integration of risk management into the machinery design process (2011) Mechatronics: Recent Technological and Scientific Advances, pp. 473-482. ISBN: 978-364223243-5

[Blecha 2011b] Blecha, P., Prostrednik, D. Influence on the failure probability (2011) Annals of DAAAM and Proceedings of the International DAAAM Symposium, pp. 11-12. ISBN: 978-390150983-4

[Blecha 2011c] Blecha, P., Novotný, L.W. Integration of risk management into the process of PLC-software development in machine tools (2011) Mechatronics: Recent Technological and Scientific Advances, pp. 19-24. ISBN: 978-364223243-5

[Tuma 2014] Tuma, Z., et al. The process simulation using by virtual reality (2014) Procedia Engineering, 69, pp. 1015-1020. doi: 10.1016/j.proeng.2014.03.084

[K.V.N.S.L., Kumar S. 2017] Analysis of Errors in Safety Critical Embedded System Software in Aerial Vehicle.

[Smruti P. Nanada 2019] Smruti P., et al. Survey of Formal Specification Application to Safety Critical Systems. University of North Dakota. 978-1-7281-3323-2

[MEDOFF1 2014] Rainer Faller. FUNCTIONAL SAFETY: Compliant Development Process. 3rd Edition. Sellersville, PA, USA: Exida, 2014. ISBN 978-1-934977-08-8.

CONTACTS:

Ing. Michal Drlik
Brno University of technology
Ujezd u Brna, 9. kvetna 806, 664 53, Czech Republic
+420 608 730 895, michal.drlik@vut.cz