Software Testing in Critical Embedded Systems: a Systematic Review of Adherence to the DO-178B Standard

Jacson Rodrigues Barbosa*, Auri Marcelo Rizzo Vincenzi*
*Instituto de Informática
Universidade Federal de Goiás, UFG
Goiânia-GO, Brazil
E-mail: {jacsonbarbosa,auri}@inf.ufg.br

Márcio Eduardo Delamaro†, José Carlos Maldonado†
†Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo, USP
São Carlos-SP, Brazil
E-mail: {delamaro,jcmaldon}@icmc.usp.br

***Abstract*—Computing is becoming increasingly critical as far as embedded applications are concerned. Depending on the software, its malfunction may have consequences varying from serious financial problems to the loss of human lives. In view of this, this paper presents a systematic review that investigates the evolution of the work-related activity of embedded software critical tests in order to assess the level of compliance of the works found in relation to the DO-178B standard (*Software Considerations in Airborne Systems and Equipment Certification*). The ultimate goal of this research is the composition of existing works to define a test process that incorporates the quality and DO-178B requirements considering the different levels of criticality.**

***Keywords*-software testing; critical embedded system; DO-178B.**

I. Introduction

Embedded systems are often critical computational modules for monitoring and control used together with physical devices such as robots, autonomous vehicles and unmanned aircraft. Some systems impose restrictions with regard to security, performance, reliability and other factors, since failures in these systems may result in danger to human lives, environmental hazards or high financial losses.

By aiming to ensure quality levels which will reduce the chances of these tragic events, the *Radio Technical Commission for Aeronautics* (RTCA), together with the *European Organization for Civil Aviation Equipment* (EUROCAE) have created the DO-178B standard, which provides a set of guidelines for the development and certification of embedded software systems and applications, since these devices cannot be marketed by the industry without the latter's approval of this standard [1].

Because of this, the National Institute of Science and Technology Critical Embedded Systems (INCT-SEC) has recently been created to establish a network of collaboration and research in critical embedded systems (CES) [2]. The present work supports the goals of the INCT-SEC, investigating the evolution of research in software testing of critical embedded systems through a systematic review (SR) and assessing the level of compliance of such research with the DO-178B, in an attempt to identify a set of works which could be used together in a methodology for CES testing.

This paper is organized into four sections. Section II presents the main concepts related to the DO-178B standard. Section III describes the SR planning. Section IV shows the results obtained after conducting the review. Section V presents our conclusions on the topic and suggests future work to be carried out in the field.

II. Background: the DO-178B standard

The DO-178B standard defines the software's demand levels by considering the effects (failure condition) produced if the software behaves abnormally. Table I shows this relationship.

Table I
Software levels and failure condition (adapted from [3])

Software Level	*Failure Condition*
A	Catastrophic
B	Hazardous
C	Major
D	Minor
E	No effect

In DO-178B, the test on critical embedded systems has aims that complement the software verification process, showing that such software meets the relevant requirements and reveals a high degree of certainty that the defects which could lead to unacceptable failure conditions were removed [1].

To meet these goals in the software testing process, the standard defines a set of five requirements.

A. Normal range test cases

Normal range test cases show the software's ability to respond to inputs and normal conditions; for instance, an entire input variable should be executed by using valid equivalence classes and limit values.

ISBN: 978-1-61208-168-7

B. Robustness test cases

Robustness test cases demonstrate the software's ability to respond to inputs and abnormal conditions; for instance, an input variable must be performed by using values of invalid equivalence classes.

C. Requirement-based testing methods

There are three testing methods based on requirements: integration of requirement-based hardware/software, integration testing of requirement-based software and low-level requirement-based test.

D. Requirement-based test coverage analysis

The purpose of this analysis is to determine how the implemented software requirements were verified with the requirement-based tests.

E. Structural coverage analysis

This analysis aims to show how the code structure was not executed by the requirement-based test.

Given the importance of the DO-178B standard as regards software certification for critical embedded systems used in aviation, one of the goals of INCT-SEC is to develop software for the control of unmanned aerial vehicles. The following section shows the planning of the systematic review conducted to identify previously developed research in this area of expertise.

III. Systematic review planning

The systematic review (SR) was planned following the model proposed by [4]. Figure 1 shows the SR's development process.

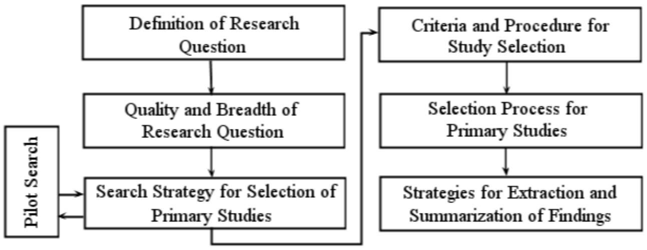
Definition of Research Question
Quality and Breadth of Research Question
Search Strategy for Selection of Primary Studies
Pilot Search
Criteria and Procedure for Study Selection
Selection Process for Primary Studies
Strategies for Extraction and Summarization of Findings

Figure 1. Systematic review process (adapted from [5])

A. Definition of Research Question

The purpose of the SR was to find answers to the following questions:

- **Primary Research Question 1 (PRQ1):** What techniques and software testing criteria have been proposed for software testing in critical embedded systems?
- **Secondary Research Question 1.1 (SRQ1.1):** What standards have been proposed for software testing in critical embedded systems?
- **Primary Research Question 2 (PRQ2):** What is the degree of adherence of experimental studies related to the objectives and activities of the software testing process defined in DO-178B?
- **Secondary Research Question 2.1 (SRQ2.1):** What evidence is there to confirm that the objectives and activities of the software testing process defined in DO-178B provide high quality standards in critical embedded systems?
- **Primary Research Question 3 (PRQ3):** What has been the strength of evidence supporting the conclusions drawn?

B. Quality and Breadth of Research Question

A well-formulated research question includes the following elements:

1) Keywords and Synonyms: the following were regarded as keywords in English:

- critical embedded, safety-critical, mission-critical, embedded software
- software test, system test

2) Intervention: software testing processes, techniques and criteria were observed in this review.

3) Control: we identified eight articles relevant to the context of this work , which served as control items of the search string. If the search string came up with all these articles, then that would confirm its appropriateness.

4) Population: the group was observed by researchers and software developers working on the design and construction of critical embedded systems.

5) Findings: software verification and validation (V&V) activities, software testing methodology, techniques and criteria for testing software used in the context of critical embedded systems.

6) Application: software development projects implemented within the context of critical embedded systems.

C. Search Strategy for Selection of Primary Studies

By taking into account the keywords, study sources, language and types of primary study, the following were selected for review:

1) Listing sources: electronic indexed databases IEEE Xplore (IEEE) and ACM Digital Library (ACM).

2) Language of primary studies: English.

3) Type of primary studies: reference lists of primary studies, journals, technical reports and conference proceedings.

D. Pilot Search

From the research questions and their respective attributes of quality and breadth, a search string was defined in order to perform an initial evaluation: *(critical embedded OR safety-critical OR mission-critical OR embedded software). AND (test) AND (software OR system)*

E. Criteria and Procedure for Selection of Studies

1) Inclusion criteria: the following inclusion criteria were defined:

- IC_1 – implementation of software V&V (static and/or dynamic) activities in the context of critical embedded systems;
- IC_2 – application of techniques and test criteria (dynamic) for software in critical embedded systems.

2) Exclusion criteria: the following exclusion criteria were defined:

- EC_1 – implementation of software V&V activities in the context of non-critical embedded systems e.g. mobile applications;
- EC_2 – application of techniques and criteria for software testing in non-critical embedded systems;
- EC_3 – does not address the activities of software V&V or techniques and criteria for software testing in the context of critical embedded systems.

F. Selection Process for Primary Studies

1) Primary Selection Process: search strings were formed by combining synonyms of the keywords identified. These strings were used to conduct searches in the search engines mentioned. The studies found through this research were analyzed by two reviewers (co-authors of this paper), who read and reviewed their titles and abstracts to rate them in terms of importance. If the reviewers reached an agreement over a given article, the manuscript was selected to be read in full.

2) Final Selection Process: we performed a thorough reading of papers selected in the preliminary stage by at least one of the reviewers.

3) Evaluation of Primary Studies: all primary studies were assessed individually by the reviewers based on the criteria defined in [5]. Reviewers then produced a document containing the summary, methodology and testing techniques mentioned in the primary studies, as well as other related concepts.

G. Strategies for Extraction and Summarization of Findings

For each primary study selected, we used the JabRef tool to store the collected data [6].

The summary of the results collected was organized into a tabular format.

IV. Data analysis

In Figure 2, Phase 1 amounts to the number of primary studies found by indexed electronic databases after submitting the query string (n=872). Phase 2 shows the number of studies resulting from the primary selection process (n=285); the remaining n=587 were excluded because their titles and summaries did not address the SR's scope of research questions. In Phase 3, n=185 were eliminated after the reading because they failed to meet the SR's full scope, thereby leaving n=100 primary studies. Finally, in Phase 4 n=3 were eliminated following the evaluation of primary studies according to quality criteria defined in the SR planning; they were considered of low quality. Thus, n=97 primary studies selected for extraction and summary of results remained. These phases were carried out by the authors during a period of five months. Further details about the primary studies selected are available in [7].

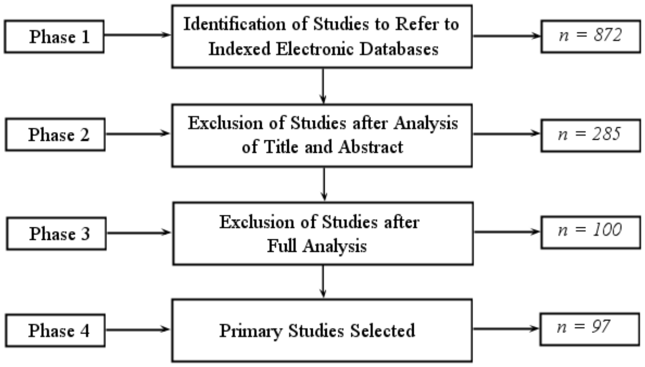
Phase 1
Identification of Studies to Refer to Indexed Electronic Databases
n = 872
Phase 2
Exclusion of Studies after Analysis of Title and Abstract
n = 285
Phase 3
Exclusion of Studies after Full Analysis
n = 100
Phase 4
Primary Studies Selected
n = 97

Figure 2. Phases of the final selection, adapted from [4]

Tables II, III and IV summarize the data of 97 primary studies and show partial quantities (IEEE and ACM), total studies (n) and total percentage of studies (%).

Table II presents quantitative information about the type of experimental study employed in the papers selected. This classification is based on the terminology defined by [4], according to which multiple-case refers to projects that include more than one case. By examining the table, it seems that 59.79% of the studies are observational (single-case and multiple-case), thus indicating that the majority are a result of monitoring one or more projects in depth.

Table II
Type of experimental study

Experimental Study	IEEE	ACM	n	%
Single-case	28	20	48	49.48
Multiple-case	4	6	10	10.31
Experiment	5	12	17	17.53
Survey	3	0	3	3.09
Not mentioned	12	7	19	19.59
Total	52	45	97	100

Table III shows the software testing techniques employed by the studies; if one approaches more than one technique, for instance quantity (q) equal to 3, a value of 0.33 (q^{-1}) would be assigned for each technique. It appears that the functional testing technique is most frequently used (36.77%), followed by model-based testing (28.36%), which allows us to eliminate ambiguities and to derive test cases from the model. The paper in [8] proposes the transition coverage criterion based on security requirements as a new alternative for the model-based testing technique.

Table III
Software testing techniques explored

Software testing	IEEE	ACM	n	%
Model-based testing	14.5	13	27.5	28.36
Mutation testing	0.33	1.83	2.16	2.23
Structural testing	8.83	14.83	23.66	24.4
Functional testing	21.33	14.33	35.66	36.77
Not mentioned	7	1	8	8.25
Total	51.99	44.99	96.98	100

Among the primary studies selected in the SR, the *Safety Critical Application Development Environment* (SCADE) has been widely quoted to specify critical embedded software, since the SCADE Suite allows the automatic generation of C code from specific models, such as state machines.

As regards the software testing criteria explored by the studies in question, the equivalence partition criterion was the most frequently used (12.37%). As far as structural testing criteria are concerned, the *Modified Condition/Decision Coverage* MC/DC was the most frequently used (10.23%). The remaining structural criteria required by DO-178B have been explored as follows: *Decision Coverage* DC (0.68%) and *Statement Coverage* SC (5.82%) , in response to PRQ1.

With respect to the norms and standards related to the development of critical embedded software (SRQ1.1), the DO-178B standard was the most frequently used (20.92%), thus indicating that the objectives and activities defined in DO-178B provide conditions to build a critical embedded software quality (in response to SRQ2.1). Furthermore, standards were used for specific industrial contexts; for instance, the standards set by the *European Committee for Electrotechnical Standardization* (CENELEC) are recommendations for the development and testing of rail transport systems [9].

As can be seen in Table IV, the requirement *Structural coverage analysis* of DO-178B has been extensively investigated - this is possibly due to the complexity associated with it. It has been observed that 36.09% of the studies carried out are not directly mappable to DO-178B test requirements, due to the fact that most studies address issues related to the definition of software life cycle models or other critical embedded software processes (responding to PRQ2). Further detailed information on this topic can be found in [7].

To meet software level *A*, the criteria for structural testing (MC/DC) must be adhered to, as these are the most rigorous structural criteria defined by DO-178B. The article in [10] presents a case study that meets the criteria when looking for MC/DC. Important errors that failed to be identified by functional technique were found by MC/DC, thus demonstrating its effectiveness for identifying critical bugs and for complementing the functional technique.

However, the study in [8] presents a subsumption hierarchy which compares the MC/DC and other criteria, thereby confirming that the *Multiple-Condition Coverage* test (M-CC) is more stringent than the MC/DC. In terms of overall effectiveness for fault detection, the following tests stand in decreasing order: MC/DC < CUTPNFP < MUMCUT < M-CC.

Table IV
Adherence to testing process requirements

DO-178B Testing Process Requirements	IEEE	ACM	n	%
Normal range test cases	4.61	2.91	7.52	7.75
Robustness test cases	4.48	2.58	7.06	7.28
Requirement-based testing methods	3.11	4.33	7.44	7.67
Requirement-based test coverage analysis	9.65	3.58	13.23	13.64
Structural coverage analysis	10.15	16.58	26.73	27.56
No direct mapping to DO-178B requirements	20	15	35	36.09
Total	52	44.98	96.98	100

By comparing Tables III and IV, it appears that the functional testing technique was the most frequently used, but the first two requirements of the DO-178B testing process shown in Table IV, which adhere to the functional test criteria (partitioning equivalence and boundary value analysis), have few related works. This is because the corresponding functional test criteria employed were not specified in this subset of primary studies.

As regards study characteristics, 59.79% of the studies selected are observational (as shown in Table II), whereas only 17.53% correspond to experiments. In accordance with the guidelines of the Grading of Recommendations Assessment, Development and Evaluation (GRADE), the evidence obtained by the SR concerning study characteristics are considered low (refer to [4], [5] for an overview).

Regarding the quality of the studies selected, their approaches to data analysis were explained in a moderate way, including issues such as potential bias, credibility and study limitations. Only in one study did the researcher critically assess his own role. Credibility was discussed in 98.45% of the studies, whereas study limitations were discussed in 72.68% of them. Based on the quality criterion results, the studies show moderate evidence.

As far as the consistency criterion is concerned, we identified similarities between 63.91% of the studies regarding DO-178B requirements (as shown in Table IV), since at least one of these requirements could be associated in more than a single primary study; the rest do not emphasize the requirements (36.09%). As a result, the strength of evidence regarding consistency may be classified as moderate.

Finally, we focused on the directness criterion, which assesses whether the people involved (students or software professionals), interventions and study results are consistent with the area of interest. In the SR, most studies were carried out in an academic context, and only one study mentioned the level of knowledge and experience of the students involved - both undergraduate and experienced graduate students. As regards intervention, objective comparisons were carried out with 63.91% of the studies concerning DO-178B requirements, even though only 20.92% of the studies clearly mentioned the use of the standard in question. Finally, as for the results obtained, even though most of the studies are observational, they are not trivial, being thus comparable with the software/systems developed by the industry. This information allows us to regard the strength of evidence related to objectivity as low to moderate.

Once the four criteria (characteristics, quality, consistency and directness) are combined to determine the strength of evidence for this RS , the latter may be classified as moderate, hence responding to PRQ3. Therefore, defining the strength of evidence may help future research to have a crucial impact on the reliability of effect estimates, hence changing current estimates [5].

V. Final considerations and future work

In the analysis of the studies selected, we found that all DO-178B testing process requirements have been explored, but few studies have discussed how to solve problems of structural coverage analysis (n=2), such as dead code and deactivated code.

In the future, we intend to propose a software test methodology that supports the INCT-SEC projects compliant with DO-178B and uses the activities of software verification and validation as well as the techniques and criteria for software testing identified in the systematic review. Since DO-178B does not define how to implement the respective processes, the methodology should state how the processes are to be implemented in accordance with the necessary requirement levels.

Moreover, it is possible to reuse the SR protocol further, collecting more data aiming at identifying how the state of the art evolved and the still missing pitfalls in the context of testing of critical embedded system (CES).

Acknowledgments

We are indebted to CNPq (573963/2008-8), FAPESP (08/57870-9), FAPEG (200910267000662) and CAPES for their financial support. We also thank professor Plínio de Sá Leitão Júnior for reviewing this paper.

References

[1] *Software considerations in airbone systems and equipament certification*, RTCA SC-167/EUROCAE WG-12, 1992.

[2] J. C. Maldonado, “National institute of science and technology critical embedded systems (INCT-SEC),” *São Carlos/SP - Brazil*, 2008. [Online]. Available: http://www.inct-sec.org/?q=en-us. Accessed on: [08/18/2011].

[3] T. K. Ferrel and U. D. Ferrel, *The Avionics Handbook.* CRC Press LLC, 2001.

[4] T. Dybå and T. Dingsøyr, “Empirical studies of agile software development: A systematic review,” *Information and Software Technology*, 2008.

[5] M. S. Ali, M. Ali Babar, L. Chen, and K.-J. Stol, “A systematic review of comparative evidence of aspect-oriented programming,” *Inf. Softw. Technol.*, vol. 52, pp. 871–887, September 2010.

[6] *JabRef 2.4*, 2008. [Online]. Available: http://jabref.sourceforce.net/. Accessed on: [08/18/2011].

[7] J. R. Barbosa and A. M. R. Vincenzi, “Software testing in the context of critical embedded systems,” Web page, july 2011. [Online]. Available: http://www.inf.ufg.br/˜auri/sec-en/. Accessed on: [08/18/2011].

[8] Y. T. Yu and M. F. Lau, “A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions,” *J. Syst. Softw.*, vol. 79, pp. 577–590, May 2006. [Online]. Available: http://dx.doi.org/10.1016/j.jss.2005.05.030. Accessed on: [08/18/2011].

[9] J. Kloos and R. Eschbach, “Generating system models for a highly configurable train control system using a domain-specific language: A case study,” in *Proceedings of the IEEE International Conference on Software Testing, Verification, and Validation Workshops.* Washington, DC, USA: IEEE Computer Society, 2009, pp. 39–47.

[10] A. Dupuy and N. Leveson, “An empirical evaluation of the MC/DC coverage criterion on the HETE-2 satellite software,” in *Digital Avionics Systems Conferences, 2000. Proceedings. DASC. The 19th*, vol. 1, 2000, pp. 1B6/1 –1B6/7 vol.1.