

Automotive Software Certification: Current Status and Challenges

Huafeng Yu, Chung-Wei Lin, and BaekGyu Kim
Toyota InfoTechnology Center USA

ABSTRACT

Modern vehicles can have millions of lines of software, for vehicle control, infotainment, etc. The correctness and quality of the software play a key role in the safety of whole vehicles. In order to assure the safety, engineers give an effort to prove correctness of individual subsystems or their integration using testing or verification methods. One needs to eventually certify that the developed vehicle as a whole is indeed safe using the artifacts and evidences produced throughout the development cycle. Such a certification process helps to increase the safety confidence of the developed software and reduce OEM's liability. However, software certification in automotive domain is not yet well established, compared to other safety-critical domains, such as avionics and medical devices. At the same time, safety-relevant standards and techniques, including ISO 26262 and assurance cases, have been well adopted. It finally promotes the adoption and development of software certification in the automotive industry. In this paper, we first present a survey of recent research in the domains of aviation, medical devices, and railway systems. After this survey, we summarize current status as well as existing challenges in the automotive software certification. Assurance cases are also presented as a promising technique to automotive software certification.

CITATION: Yu, H., Lin, C., and Kim, B., "Automotive Software Certification: Current Status and Challenges," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 9(1):2016, doi:10.4271/2016-01-0050.

INTRODUCTION

As systems become more complicated than ever before, there can be millions of lines of source codes, and thus software plays a more and more important role in modern vehicles. It performs not only infotainment applications but also vehicle controls, so the guarantee of software correctness and quality becomes one of the most challenging and desired tasks. To fulfill the demand of software guarantees, software certification is therefore proposed and developed. A software certification is issued according to the artifacts and evidences produced throughout the development cycle. It helps to increase the confidence of the software in terms of safety and quality.

However, different from other safety-critical domains, the software certification in the automotive domain is not yet well established. In the domain of avionics, certification has a long history. With well-established safety regulation and standards, certification of aircraft software has been well developed, and software seldom poses severe safety problems. In the domain of medical devices, for certain types of safety-relevant devices, manufacturers in the United States need to make a 510(k) submission to Food and Drug Administration (FDA) in order to demonstrate that the device is substantially equivalent to a legally marketed device [1]. Compared with these domains, the progress of automotive software certification is still in an early stage.

ISO 26262 [2] is recognized as the state-of-the-art safety standard for automotive systems. In the standard, the Automotive Safety Integrity Level (ASIL) is defined, and safety-relevant requirements in a

lifecycle from concept and development to production, operation, and decommissioning are provided. Regarding software certification, it may be the first step toward automotive software certification as some software structures, such as unconditional jumps and recursions, are not recommended for some designs in the standard. However, even with ISO 26262, there is still no quantified and rigorous process for automotive software certification. As a result, although automotive software certification is an urgent issue, it is still a quite open problem at this point.

In this paper, we first take a look at software certification in the domains of avionics, medical devices, and railway systems and learn from their certification experiences. After introducing the automotive standard ISO 26262, we point out the current status and challenges of automotive software certification, such as lack of regulators, automated certification, and compositional certification. Lastly, we demonstrate some existing works based on assurance case patterns and templates, which are potential solutions to the problems of automotive software certification.

SOFTWARE CERTIFICATION IN OTHER DOMAINS

Aviation

According to Federal Aviation Administration (FAA), the certification authority in the United States, "The Aircraft Certification Service is concerned with the approval of software and its documentation."

hardware for airborne systems (e.g., autopilots, flight controls, engine controls), as well as that used to produce, test, or manufacture equipment to be installed on airborne products” [3].

Certification has a long history in the aviation domain. Certification of aircraft software is mainly based on standards, e.g., DO-178B [4] [5] and SAE Aerospace Recommended Practice (ARP) 4754/4754A [6] [7], SAE ARP 4761 [8]. The mentioned standards provide the guidelines for software considerations, development process, and safety assessment, respectively. These guidelines, together with other standards, are used to support certification of aircraft systems, including software systems.

SAE ARP 4761 specifies guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment. This ARP specifies a process to apply model-based safety assessment techniques on an integrated system. The techniques include Functional Hazard Assessment (FHA), Failure Mode and Effects Analysis (FMEA), Fault Tree Analysis (FTA), etc. SAE ARP4754 (guidelines for development of civil aircraft and systems) mainly addresses development processes, particularly for safety aspects. ARP 4754A introduces Functional Development Assurance Level (FDAL). It also defines error, failure, and failure condition.

DO-178B/C, software considerations in airborne systems and equipment certification, was jointly developed by RTCA and EUROCAE. It provides the guidance to the development of safety-critical avionics software systems. It defines five different Design Assurance Levels (DALs): catastrophic, hazardous, major, minor, and no effect levels. These levels indicate the severity of the failure from crash to no impact on safety. However, DO-178B does not define how to design the safety-related software and how we can guarantee the required safety. It is also expected to work with other software-related standards, such as ARP 4754A to complete the system safety assessment. DO-178B includes some software V&V methods, e.g., testing to ensure safety. FAA states that DO-178B/C provides “acceptable means, but not the only means, for showing compliance with the applicable airworthiness regulations for the software aspects of airborne systems and equipment certification.”

Conformance to the guidelines defined in ARP 4761 and ARP 4754A is expected to reduce/minimize the appearance and severity of design flaws and failure conditions at a system level; The quantitative classification of failure conditions, regarding severity and probability, adopted by ARP 4754A and ARP 4761 are defined in AMC 25.1309/ AC 25.1309 [9]. At a component level, or inside the component, particularly software component, the certification of software is generally not performed independently from their running electronics. The software may include control algorithms for the flight or devices, but software flaws should be seriously considered as they may have great impact on the overall safety of the system. Requirements are used, instead of standard, to assess the correctness. DO-178B/C provides hundreds of objectives to be achieved to ensure the software correctness, which are related to documentation, analysis and verification techniques.

In the certification process in the United States, in general, FAA works with certification applicants to define the regulations, conditions and processes that are applicable to the certification. The applicants then propose certain means of compliance that defines how development will satisfy the previous certification requirement.

Certification of aircraft software has been well developed and software seldom poses severe safety problems. This is mainly because of the well-established safety regulation and standards, as well as long history of certification culture and experience [10]. However, new languages and methods, such as formal methods and model-based design, to develop software still bring big challenges.

Medical Devices

Medical devices are also considered as safety-critical systems that require serious certification. Compared to aircraft systems, medical devices have many different features which make the certification quite different.

Certification in medical domain intends to make sure medical devices to be marketed improve the effectiveness of medical treatments without harming patients. The government agency, FDA is responsible for regulating companies that manufacture medical devices in the United States. Such safety regulations typically rely on the international standard, for example, IEC 62304 addresses software life-cycle processes and identifies core processes for high-integrity medical device software. The certification process, length, and cost vary depending on the device classification that has been established by FDA. Each medical device is belonged to one of three regulatory classes (Classes I, II, III) based on the level of control necessary to assure the safety and effectiveness of the device [11]. Class I devices require general regulatory controls that typically do not need FDA review. The examples of such medical devices are non-powered breast pumps and elastic bandages. Class II devices require general controls and special regulatory controls that typically need Premarket Notification, called 510(k) submission and clearance. A 510(k) is a premarket submission made to FDA to demonstrate that the device is at least as safe and effective (i.e., substantially equivalent) as a legally marketed device (i.e., predicate device). The examples of such medical devices are CT scanners, infusion pumps and powered wheelchairs. Class III devices require general controls and Premarket Approval (PMA) [12]. These devices have substantial importance in preventing impairment of human health. Hence, FDA determined that general and special controls are not sufficient enough to assure the safety and effectiveness of the Class III devices. The examples of such medical devices are implantable pacemakers and external defibrillators. In general, regulatory control increases from Class I to Class III; that is, the cost, review length, and clearance standard become higher, longer, and tougher as a medical device belongs to the higher level of classification. In order to give more detailed intuition about this regulatory control, we explain the 510(k) process as an example.

510(k) Process of Medical Devices

Manufacturers who want to sell medical devices belonged to Class II typically need to make a 510(k) submission to FDA in order to demonstrate that the submitted device is substantially equivalent to a legally marketed device. The claim for such a substantial equivalence is typically made by choosing a predicate device that was legally marketed previously and showing that the new device is at least as safe and effective as the predicate. More specifically, the manufacturers should convince the FDA that the device has the same intended use as the predicate device, and has the same technological characteristics as the predicate. The determination of the 510(k) clearance usually takes 90 days based on the information submitted by the manufacturers. Recently, the FDA typically receives more than 3,000 devices annually that need to be determined for the premarket clearance by the 510(k) process. According to the survey [13], however, the average length from submission to clearance for 510(k) takes 167 days during the period between 2011 and 2014, which is significantly higher than the average of 96 days during the period between 2001 and 2005. This implies that the FDA has been facing significant challenges to clear 510(k) submissions.

Software Challenges of Medical Devices

Software contributes to the certification challenges in the medical device domain. Modern medical devices operate large amounts of software in order to perform various monitoring and care delivery tasks. For example, modern infusion pump systems may contain more than 100,000 lines of source code [14], and large amount of infusion pumps have been being recalled due to software problem according to the FDA's infusion pump improvement initiative [15]. The risks related to such complex software are not well understood by device manufacturer and government regulators [16]. For this reason, it is not trivial to sort out whether a certain type of software modification is significant or not in terms of the safety. Therefore, a minor software change (e.g., firmware upgrade, user-interface design change) may need to go through the certification process again even though the device has been certified before. In addition, the human and budget resources that are required to perform the certification seem not grow as fast as the pace of the growth of the software-oriented medical industries due to increasing importance of health care.

Current Approaches of Medical Devices

In order to address those certification issues, researchers have been conducting various researches for the medical device certification. One of the efforts is to use the safety assurance case (safety case) in the certification process. The safety case is a structured argument, supported by evidence, intended to justify that a system is acceptably safe for a specific application in a specific operating environment [17]. According to the new guidance of infusion pumps total product life cycle [18], the FDA recommends manufacturers to submit the safety cases in preparation of their 510(k) submission. In addition, the FDA encourages to develop, maintain, and refine the safety assurance cases throughout the development of the device. More detailed information about the safety cases and related research will be given in the later section.

Railway Systems

IEC 61508 is a general standard for the functional safety of electrical/electronic/programmable electronic safety-related systems [19]. In the standard, the safety life cycle of a system is introduced, including 16 phases and 3 groups: analysis, realization, and operation. The Safety Integrity Level (SIL) is also defined, and it has been adapted by IEC 62279 which is specifically for railway applications. On the other hand, there is also a set of EN standards for railway applications. Most of them are on installations, electromagnetism, communication, and signal processing. Among them, EN 5012X family is most relevant to software certification [20]. EN 50126 defines the specification and demonstration of reliability, availability, maintainability, and safety (RAMS) and provides the requirements regarding RAMS with the lifecycle of a system. EN 50128, especially for software of railway control and protection systems, recommends or strongly recommends techniques and measures according to different SILs. EN 50129 focuses on safety-related electronic systems for signaling and the evidence to support on safety-related systems, and thus it is highly-related to EN 50126.

The aforementioned three domains, together with the automotive domain, are quite different from one to the other, and it leads to the different adoption, process, standards, confidence, liability, etc., in the software certification. The aviation system certification has many advantages than others due to the long history in certification culture and experience, as well as matured safety standards and development processes. These advantages cannot be found in the automotive domain. Medical device domain shares some similarities to automotive domain, e.g., fast product design and manufacturing, mass production, and short history of certification. In the next sections, we will give a summary of the automotive safety standard ISO26262, followed by current status and challenges for automotive software certification, which can be very different from previously listed domains.

AUTOMOTIVE STANDARD - ISO 26262

ISO 26262 [2] is recognized as the state-of-the-art standard for automotive systems. It is an adaption of IEC 61508. ISO 26262 defines the required activities within automotive lifecycle from the stages of concept and development to the stages of production, operation, and decommissioning. It also covers management of functional safety, supporting processes, safety analysis, and some guidelines. Throughout the standard, ASIL is referred, and there are three questions addressed by the ASIL: (1) How to define a hazardous event? (2) How to prevent a hazardous event? (3) How to control a hazardous event if it happens? To address these questions, the ASIL is assessed by three classifications:

1. Severity Classifications (S)
 - S0: No injuries.
 - S1: Light and moderate injuries.
 - S2: Severe & life-threatening injuries (survival probable).
 - S3: Life-threatening (survival uncertain) & fatal injuries.
2. Probability of Exposure Classifications (E)
 - E0: Incredibly.

