Test tool qualification through fault injection

Q. Wang, A. Wallin, V. Izosimov and U. Ingelsson
EIS by Semcon AB, Sweden
E-mail: {andreas.wallin,viacheslav.izosimov,
urban.ingelsson}@eis.semcon.com

Z. Peng
Linköping University, Sweden
E-mail: zebo.peng@liu.se

Abstract—**According to ISO 26262, a recent automotive functional safety standard, verification tools shall undergo qualification, e.g. to ensure that they do not fail to detect faults that can lead to violation of functional safety requirements. We present a semi-automatic qualification method involving a monitor and fault injection that reduce cost in the qualification process. We experiment on a verification tool implemented in LabVIEW.**

I. Introduction

Verification tools employed on safety-critical automotive embedded systems must undergo qualification according to ISO 26262 [1]. This qualification effort can become a bottleneck in the development process due to frequent modifications of the verification tool and subsequent re-qualifications. To remove the bottleneck we present a semi-automatic qualification method that employs a monitor for checking for erroneous tool behavior and fault injection for gaining confidence in the capability of the monitor.

In this context, Conrad et al. [2] provide a way to qualify development tools by enforcing a development flow with checks applied in every step that involves a development tool. However, this flow is not suitable for verification tools, in particular Hardware-in-the-Loop test benches that are frequently adapted to the system-under-test.

II. Our Qualification Method

When qualifying a verification tool after a modification, using a monitor and fault injection, we iteratively apply functional stimuli that are designed to exercise the verification tool software. In each iteration we inject a known selected fault from a pre-defined fault list. If the injected fault is not detected by the monitor, one of the following actions is required.

1) Eliminate "bugs" in the verification tool, then re-start the qualification process, or
2) If no "bugs" can be found in the modified verification tool, analysis on conformance to the functional safety requirements will determine the next action as follows:
   - Requirements are met and we can reduce the lists of faults to inject, or
   - Requirements are not met and modification of the monitor is necessary to detect the "bug" undetected so far, followed by re-qualification of the monitor.

If all injected faults are detected as expected in the adapted verification tool, no re-qualification is necessary.

When applying the verification tool on a new system-under-test, the tool behavior is observed for each test case executed during a "golden run" (without fault injection) and then the tool is exercised with fault injection for the same test cases as in the "golden run". If a mismatch is observed, the test case shall be modified.

A key idea in enabling the qualification method is to keep down the complexity of the monitor and the fault injectors. We define the Injector and Monitor Placement problem. Solving this problem results in an efficient qualification and re-qualification of the verification tool.

Problem [Injector and Monitor Placement (IMP)]
Given a verification tool, a stimuli set and a set of critical parameters, place a minimal set of fault injectors and monitoring points into the tool, to cover all critical parameters. □

We solve this problem in three steps: (1) Hierarchy Graph Analysis, which places fault injectors into deliberately selected functions, (2) Profiling-based Monitor Placement, which identifies functions that make many calls to other functions and places monitoring points accordingly, and (3) Fault Injector Minimization, which reduces the number of fault injectors by formulating a minimum set covering problem, solved by using an ILP-solver.

We analyzed the effort required for qualification using our method and compared with a flow suggested in [2] and evaluated our qualification method on an in-house verification tool implemented in LabVIEW. We found that for a significant number of scenarios our method led to less qualification effort. The presented three steps can achieve full critical parameter coverage, while limiting the complexity in terms of monitoring points and fault injectors.

III. Conclusions

We have presented a method for semi-automatic qualification of verification tools and minimization of re-qualification effort. Our method is based on a monitor and fault injection that facilitate validation of the verification tool in presence of modifications. We have evaluated our method on a verification tool implemented in the graphical data flow language LabVIEW. Our results show that the method removes a qualification bottleneck in the development process.

References

[1] "ISO26262 - Road vehicles – Functional safety," http://www.iso.org/iso/catalogue_detail.htm?csnumber=43464, 2011, International Organisation for Standardization.

[2] M. Conrad, P. Munier, and F. Rauch, "Qualifying Software Tools According to ISO 26262," (white paper) http://www.mathworks.se/automotive/standards/iso-26262.html, 2010, The Mathworks, Inc.

978-1-4673-0697-3/12/$31.00 ©2012 IEEE