

EXtended Automation Method (EXAM) zur automatisierten Funktionserprobung von Steuergeräten in der Automobilindustrie

Sebastian Thiel, Dirk Zitterell

Hardware-in-the-Loop-Funktionserprobung
AUDI AG
I/EE-33
85045 Ingolstadt
sebastian1.thiel@audi.de
dirk.zitterell@audi.de

Abstract: Getrieben durch neue Innovationen, stetige Weiterentwicklung bestehender Funktionen und wachsende Kundenansprüche wird in modernen Automobilen eine stetig wachsende Anzahl immer komplexerer Elektronikkomponenten verbaut. Zur Qualitäts- und Funktionsabsicherung sind klar definierte Prozesse und prozessunterstützende Methoden notwendig. Im Volkswagen-Konzern wurde die Testautomatisierungsmethode EXAM entwickelt und ausgerollt. Die Methode und der Prozess werden in diesem Artikel kurz vorgestellt.

1 Einleitung

Die Fahrzeugentwicklung hat sich bedingt durch den rapiden Zuwachs an verwendeten Elektronikkomponenten in den letzten Jahren stark gewandelt. Elektrik, Elektronik, Mikrocontroller und Software entwickeln sich immer mehr zur Schlüsseltechnologie im Fahrzeug. Einen Überblick über den Einsatz von Elektronikkomponenten in der Geschichte des Automobilbaus und aktuelle Trends geben [WI08] und [RE07].

Die größte Herausforderung für Hersteller und Zulieferer ist es, die fehlerfreie Funktion ihrer Produkte zu gewährleisten. Zur Qualitäts- und Funktionsabsicherung sind automatisierte, regressionsfähige Testprozesse erforderlich, da klassische manuelle Testprozesse der wachsenden Komplexität, die maßgeblich durch die Vernetzung der Steuergeräte gesteigert wird, nicht mehr gerecht werden.

Im Volkswagenkonzern werden einzelne und vernetzte Elektronikkomponenten bezüglich ihres spezifizierten funktionalen Verhaltens unter anderem mithilfe von Hardware-in-the-Loop-Prüfständen (HiL-Prüfständen) geprüft. Die zu überprüfenden elektronischen Komponenten werden als Echteile verbaut und via Bussysteme miteinander vernetzt. Alle weiteren Fahrzeugkomponenten, das Fahrzeugverhalten und die Fahrzeugumgebung werden durch Verhaltensmodelle simuliert.

Die Automobilelektronik bietet ein großes Potential für Innovationen, jedoch muss gleichzeitig ein Prozess zur Sicherstellung der Qualität entwickelt und etabliert werden. Im Folgenden wird kurz der Testprozess der AUDI AG und die prozessunterstützende Methode EXAM vorgestellt.

2 Testprozess

Durch die gestiegene Fahrzeugkomplexität ist im selben Verhältnis auch der Aufwand zur Absicherung der Funktionen und der Qualität angestiegen. Zur Aufwandsbewältigung dieser veränderten Situation sind klar definierte Prozesse und Methoden von entscheidender Bedeutung.

2.1 Übersicht

Der Testprozess der AUDI AG für die funktionale Überprüfung von Steuergeräten mithilfe von HiL-Prüfständen gliedert sich auf oberster Ebene in die Bereiche Testanforderungsmanagement, Testausführungsmanagement und Testergebnismanagement. Die Bereiche werden von einem zentralen Testmanagement überwacht und umfassen ihrerseits eine Vielzahl von Teilprozessen. Aufgrund der Komplexität des Prozesses existiert eine Rollentrennung der verschiedenen Bereiche. Nur durch diese klare Abgrenzung können die spezifischen Anforderungen jeder Rolle erfüllt werden. In Abbildung 1 ist der Testprozess in einem Testprozessmodell dargestellt.

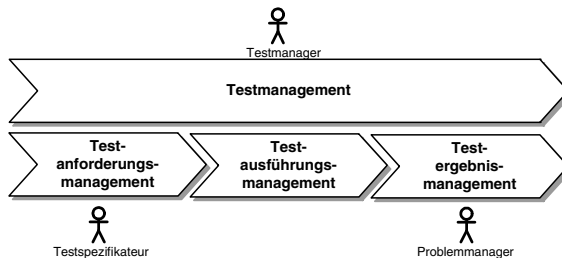


Abbildung 1: Toplevel Testprozess

Das Testanforderungsmanagement spezifiziert die Testfälle informell und verteilt die Testaufträge auf die verschiedenen Prüfplätze. Als Artefakte werden an den nächsten Prozessschritt der Testauftrag und die zugehörigen informellen Testspezifikationen weitergegeben. Die Aufgabe des Testausführungsmanagements besteht in der Formalisierung der Testfallspezifikationen (Testmodellierung), der Testausführung und der Bereitstellung der Testergebnisse. Die Teilprozesse des Testausführungsmanagements werden im nächsten Unterkapitel beschrieben. Das Testergebnismanagement erhält die Testergebnisse, bewertet diese und ergreift entsprechende Maßnahmen. Bei einem positiven Testergebnis wird die getestete Funktion für den Einsatz im Fahrzeug freigegeben. Bei einem negativen Testergebnis ist ein Fehlerabstellprozess einzuleiten.

2.2 Testausführungsmanagement

Nach Erhalt des Testauftrages müssen die informellen Testfallspezifikation, die in Form von natürlichsprachlichen Dokumenten in dem Anforderungsmanagementsystem Telelogic DOORS® [DO08] abgelegt sind, geprüft und in eine formale Testspezifikation überführt werden. Im nächsten Prozessschritt wird aus den formalen Testspezifikationen eine Menge von ausführbaren Testfällen bereitgestellt. Die Testfälle werden im folgenden Schritt an einem Prüfstand ausgeführt und liefern die Testergebnisse. Vor der Übergabe der Testergebnisse an das Testergebnismanagement werden diese im letzten Teilprozessschritt generiert, geprüft und zusammengefasst. Abbildung 2 zeigt die beschriebenen Teilprozessschritte des Testausführungsmanagements.

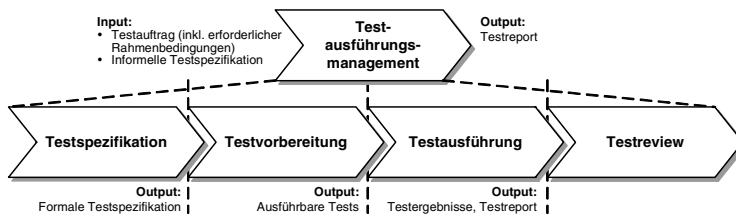


Abbildung 2: Unterprozesse des Testausführungsmanagements

3. Einheitliche Testautomatisierung

Zur Unterstützung des beschriebenen Prozesses wird im Volkswagenkonzern die EXtended Automation Method (EXAM) eingesetzt. Die Methode beschreibt die Modellierung, die Durchführung und die Auswertung der Testfälle. Dazu definiert sie die zu verwendenden Notationen und unterstützt den Prozess durch verschiedene Softwarewerkzeuge.

3.1 Probleme vor Einführung von EXAM

Entscheidend für eine effektive Prozessunterstützung ist der Einsatz einer einheitlichen Methode. In Analogie zum Testen in der Softwaretechnik spielt das Testen der einzelnen und vernetzten Steuergeräte eine entscheidende Rolle im Absicherungsprozess. Die Anforderungen an die Testfälle und Testtools und somit an die Methode sind vergleichbar mit den Anforderungen in der Softwaretechnik (vgl. [SD04]).

Ohne den Einsatz einer einheitlichen Methode ergeben sich die folgenden Probleme. Die verschiedenen Prüfstellen arbeiten isoliert, verwenden unterschiedliche Testautomatisierungstools und verschiedene Notationen für die Modellierung der Testfälle. Somit ist ein Austausch der Testfälle oder die Zusammenarbeit in einem Testthema nahezu unmöglich. Dies führt dazu, dass gleiche Testfälle an verschiedenen Prüfständen neu modelliert und implementiert werden, da keine einheitliche Funktionsbibliothek existiert.

Jedes Testautomatisierungstool muss zusätzlich an die jeweiligen Prüfstände angepasst werden, was zu einer redundanten Entwicklung von Kontroll- und Kommunikationsmechanismen führt. Ebenso muss jeder Prüfstandsverantwortlicher alle Aktivitäten der Prozesskette beherrschen, da keine Rollentrennung der einzelnen Aufgaben existiert. Weiterhin setzen die verschiedenen Verantwortlichen unterschiedliche Systeme zur Versionierung und Archivierung der Testfälle ein, was zu einer redundanten Entwicklung ähnlicher Systeme führt.

3.2 EXAM

Durch den Einsatz einer einheitlichen Methode können die zuvor angesprochenen Probleme gelöst werden. In erster Linie beschreibt EXAM ein einheitliches Vorgehen bei der Erstellung, Durchführung und Auswertung von Testfällen.

Durch die Verwendung einer einheitlichen Modellierungssprache und einheitlicher Tools für die Testerstellung, Testdurchführung und Testauswertung ergeben sich implizit die Lösungen für die angesprochenen Probleme. Für eine problemlose Zusammenarbeit der beteiligten Personen sind ein einheitlicher Sprachgebrauch und eine klare Begriffsdefinition zwingend erforderlich. Die verwendeten Begriffe der Methode wie Testfall, Testparameter oder Testergebnis müssen einheitlich verwendet werden. Alle Begriffe und deren Zusammenhänge sind in einem formalen Metamodell – vergleichbar mit dem UML-Metamodell – eindeutig beschrieben.

Als Notation für die Testfallmodellierung werden erweiterte UML-Diagramme verwendet, wobei die Semantik aller Sprachelemente im Metamodell spezifiziert ist. Mithilfe der Diagramme können die informell beschriebenen Testfallspezifikationen soweit formalisiert werden, dass sie maschinell interpretiert und automatisiert in ausführbare Testskripte transformiert werden können. Erfahrungsgemäß wird die Arbeit der Testfallentwickler durch die Verwendung einer grafischen Modellierungssprache vereinfacht.

Um die Austauschbarkeit und Wiederverwendbarkeit der modellierten Testfälle sicherzustellen, wird ein Interface-Konzept verwendet. Dieses Konzept ermöglicht die plattformunabhängige Modellierung der Testfälle, wobei alle Aufrufe innerhalb der formalen Testfallspezifikation gegen Interfaces modelliert sind und erst beim Start der Testausführung die prüfstandspezifischen Implementierungsklassen ausgewählt werden. Somit verfolgt EXAM im erweiterten Sinne den Model Driven Architecture-Ansatz der Object Management Group (OMG) [MD08].

Die Interaktion zwischen der Testautomatisierung und dem Prüfstand wird über ein Mapping-Konzept realisiert. Dabei werden die in den UML-Modellen verwendeten Testautomatisierungs-Variablen über Mapping-Klassen auf die Prüfstandsvariablen abgebildet. Die einzelnen Mappingvariablen werden als Attribute der Mapping-Klassen angelegt und automatisch die zugehörigen Get- und Set-Methoden generiert, die in den Testfällen verwendet werden. Den Mapping-Attributen werden die entsprechenden Prüfstandsvariablen zugeordnet.

Die Verwendung von Parametern in der formalen Testfallspezifikation ermöglicht es, aus einer Testfallbeschreibung unterschiedliche Testfälle zu generieren. Die ausführbaren Testfälle bestehen aus einer Komposition der Testablaufbeschreibung und dem zugehörigen Parametersatz. Ein Beispiel für die Verwendung von Parametern ist eine Testsequenz, welche das Fahrzeug auf eine Geschwindigkeit von x km/h beschleunigt. Für x können mithilfe von Parametersätzen verschiedene Geschwindigkeitswerte angegeben werden.

Durch Verwendung einer einheitlichen Modellierungssprache, einheitlicher Tools und einer einheitlichen Datenbasis können wiederkehrende Aktionen in den Testfällen, wie z.B. das Beschleunigen auf eine Geschwindigkeit x , die Ansteuerung der Prüfstandsnetzteile oder die Echtzeitsignalaufzeichnung als Bibliotheksfunktionen allen Anwendern zur Verfügung gestellt werden. An dieser Stelle wird das Interface-Konzept eingesetzt um die Entkopplung von einem konkreten Prüfstand zu erreichen.

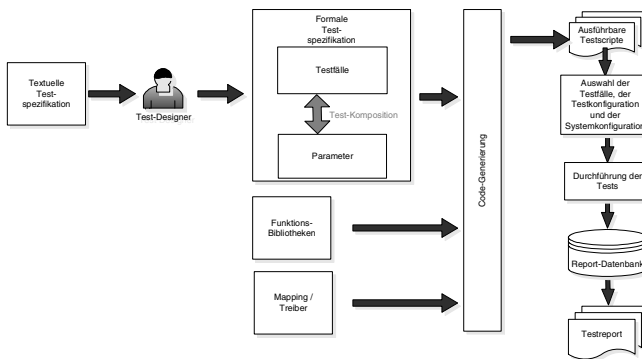


Abbildung 3: Testautomatisierung mit EXAM

Neben Funktionen für die Konfiguration und Stimulation des System under Test (SUT) enthält die Funktionsbibliothek eine Vielzahl an Funktionen für die Auswertung der Testergebnisse. Für die Auswertung der Testfälle zur Laufzeit muss das geforderte Sollverhalten des Systems in den Testfällen spezifiziert werden und die Auswertefunktionen entsprechend verwendet werden.

Die Ergebnisse der Auswertung werden in einem einheitlichen Format in einer zentralen Reportdatenbank abgelegt. Die einheitliche Darstellung und die zentrale Speicherung der Reportdaten ermöglichen eine schnelle Übersicht und Bewertung der Testergebnisse.

Abbildung 3 verdeutlicht den EXAM zugrunde liegenden Prozess. Die informellen Testspezifikationen werden vom Test-Designer in eine formale Testspezifikation übertragen. Dazu werden Use-Case-, Sequenz-, Objekt- und Klassendiagramme verwendet. Gleichzeitig erstellt der Test-Designer die für die Testfälle benötigten Parametersätze. Bei der Modellierung kann er auf eine Vielzahl von Bibliotheksfunktionen und bestehende Testfallfragmente zurückgreifen.

Im nächsten Schritt werden den Testfällen die Parameter zugeordnet (Testfallkomposition). Aus der Komposition, den verwendeten Bibliotheksfunktionen und den benötigten prüfstandsspezifischen Treibern und Mapping-Definitionen werden vollständig automatisiert ausführbare Testskripte generiert. Der Prüfstandsverantwortliche wählt werkzeugu-nterstützt eine Menge von Testfällen und führt diese aus. Nach erfolgreicher Ausführung werden automatisch Testreports generiert und in der Reportdatenbank gespeichert.

4 Zusammenfassung und Ausblick

Durch die Einführung der EXAM-Methode im VW-Konzern ist eine prüfstandsübergreifende Zusammenarbeit bei der Erstellung und Durchführung von Testfällen erreicht worden. Die einheitliche grafische Modellierung der Testfälle erleichtert die Arbeit der Testfallentwickler erheblich.

Die EXAM-Methode erlaubt, Testinhalte strukturiert und formal in einem Modell zu spezifizieren. Die Abstraktion sowie die Verwendung des Interface-Konzepts und der standardisierten Bibliotheksfunktionen erleichtern den Austausch von Testfällen. Ebenso werden durch die Verwendung eines einheitlichen Testautomatisierungssystems unnötige redundante Entwicklungen von Tools, Methoden und Testfällen vermieden. Neben den Vorteilen bei der Testfallerstellung, der Versionierung und Verwaltung der Testfälle werden auch die Testergebnisse zentral und in einem einheitlichen Format abgelegt und verwaltet.

Entlang der Prozesskette sind für die Unterstützung der EXAM-Methode verschiedene Tools im Einsatz. Gegen Ende des Jahres werden diese zu einem auf der Eclipse Rich Client Platform [EC08] basierenden Framework zusammengefasst. Dieses wird in Zukunft über eine Freeware-Lizenz zur Verfügung stehen.

Literaturverzeichnis

- [DO08] Telelogic: Doors, <http://www.telelogic.de/products/doors/>, 21.04.2008.
- [EC08] Eclipse Rich Client Platform, <http://www.eclipse.org/>, 21.04.2008.
- [MD08] Object Management Group: Model-Driven-Architecture, www.omg.org/mda, 21.04.2008.
- [RE07] Reif, Konrad: Automobilelektronik – Eine Einführung für Ingenieure. Vieweg-Teubner-Verlag, Wiesbaden, 2007.
- [SD04] Schieber, Reinhard; Derichsweiler, Frank: Testautomatisierung in der Automobilindustrie, ObjektSpektrum 04/2004.
- [WI08] Winzker, Marco: Elektronik für Entscheider – Grundwissen für Wirtschaft und Technik. Vieweg-Verlag, Wiesbaden, 2008.