Kombinatorische State-Transition Tests für Embedded Systems

Hohe Testabdeckungen schnell erreicht

Thomas Schütz, PROTOS Software GmbH

State-Transition Tests werden in vielen Standards für sicherheitskritische Systeme empfohlen. Sie sind aber für alle Embedded Systems eine hervorragende Methode, um schnell und strukturiert hohe Testabdeckungen zu erreichen.

Methoden zur Ableitung von Testcases

Die bekanntesten Methoden zur Ableitung von Testcases sind *Äquivalenzklassen* und *Grenzwertanalyse*. Typischerweise werden Sie in Unit Tests in Verbindung mit Code Coverage Messungen *(C0, C1, …)* verwendet, um die Vollständigkeit der Tests zu prüfen. Asynchrone, nebenläufige Komponenten wie sie in Embedded Systemen verwendet werden, können allerdings nur sehr schlecht mit synchronen, sequenziellen Unit Tests getestet werden. Auch für Integrations- oder Systemtests auf Basis von Hardware oder Software in the Loop sind diese Testmethoden kaum geeignet.

State Transition Tests sind weniger weit verbreitet. Sie bieten allerdings hervorragende Möglichkeiten komplexe, nebenläufige Software mit hohen Abdeckungen zu testen. Man ist damit in der Lage eine Pfadabdeckung für den Zustandsraum der zu testenden Applikation zu definieren (*n-switch*). Die dafür benötigten Testcases können manuell abgeleitet oder generiert werden. Durch geeignete Anwendung der Methode lassen sich sogar Tests zur Datenkombinatorik (*n-wise*) durchführen.

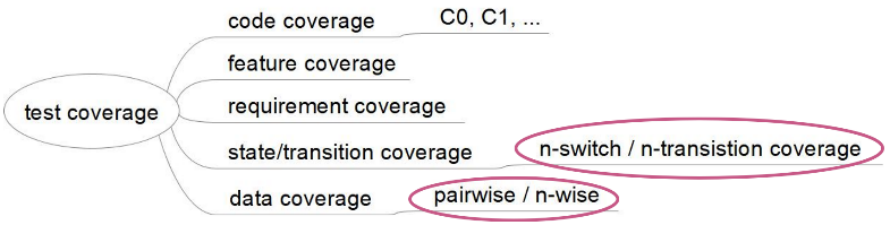

Abb. 1: Metriken zur Test Coverage für State Transition Tests

Die Methodik des State Transition Tests mit Pfadabdeckung

State Transition Tests als Black Box Tests definieren in einem State Transition Diagramm zunächst alle, von außen erreichbaren Zustände des System under Test (SUT). Transitionen beschreiben alle Zustandsübergänge zwischen den Systemzuständen. Jeder Pfad vom initialen Zustand (initial) bis zu Endzustand (end) ist ein möglicher Testpfad und damit ein möglicher Testcase. Um komplette Transition und State Abdeckung (0-switch) zu erreichen, müssen die Pfade aller erzeugten Testcases zusammen jede Transition mindestens einmal durchlaufen haben.

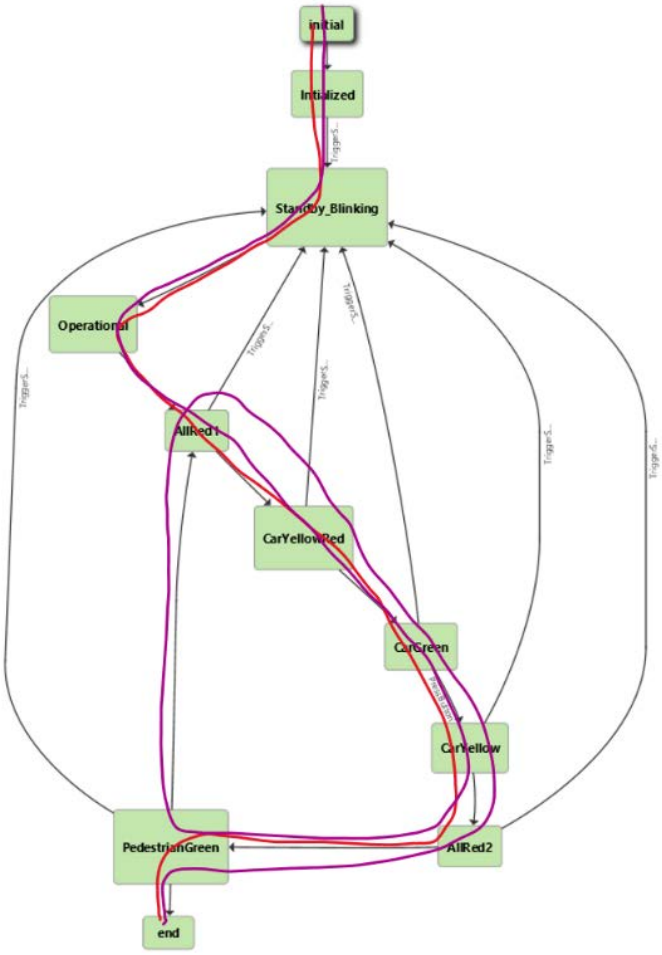

Abb. 2: Ausführungspfade für Test Cases im State Transition Diagramm

Höhere Abdeckungen definieren die Kombination aus aufeinander folgenden Transitionen:

- *0-switch:* Jede Transition wurde mindestens 1 Mal durchlaufen. Diese Abdeckung enthält auch bereits die vollständige Abdeckung der States.
- *1-switch:* Alle Kombinationen von zwei aufeinander folgenden Transitionen.
- *n-switch:* Alle Kombinationen von n+1 aufeinander folgenden Transitionen.

Durch die hohe Pfadabdeckung können die Tests viele Probleme aufdecken. Ein Beispiel hierfür sind sporadisch auftretende *Race Conditions*. Diese sind mit den meisten anderen Methoden nur schwer zu entdecken.

Tests für Datenkombinatorik

State Transition Tests sind ebenfalls gut geeignet um Datenkombinatorik zu testen. Im Beispiel werden alle Kombinationen von zwei Parametern mit jeweils drei Werten gegeneinander getestet. Dies entspricht dem *pairwise* Test. Auch Kombinationen von mehr Parametern (*n-wise*) können einfach erzeugt werden. Die dafür nötigen Werte der einzelnen Parameter können gut mit *Äquivalenzklassen* und *Grenzwertanalyse* abgeleitet werden.

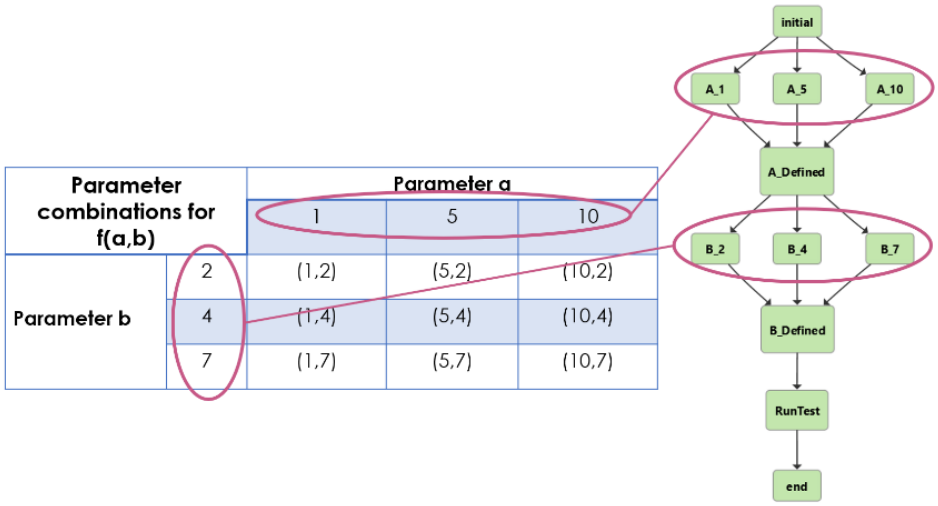

Abb. 3: Beschreibung von Datenkombinatorik mit State Transition Diagrammen

Kombination und Generierung von Daten und Pfadabdeckung

Man kann die Abdeckung von Daten und Pfaden auch gut miteinander kombinieren. So werden im Beispiel für alle Kombinationen aus 3 Parametern (*n-wise*) sämtliche Ablaufpfade (*n-switch*) getestet.

Da die große Anzahl von nötigen Test Cases manuell kaum mehr entwickelt werden kann, empfiehlt sich für größere Abdeckungen die Verwendung geeigneter Werkzeuge mit Test Case Generatoren.

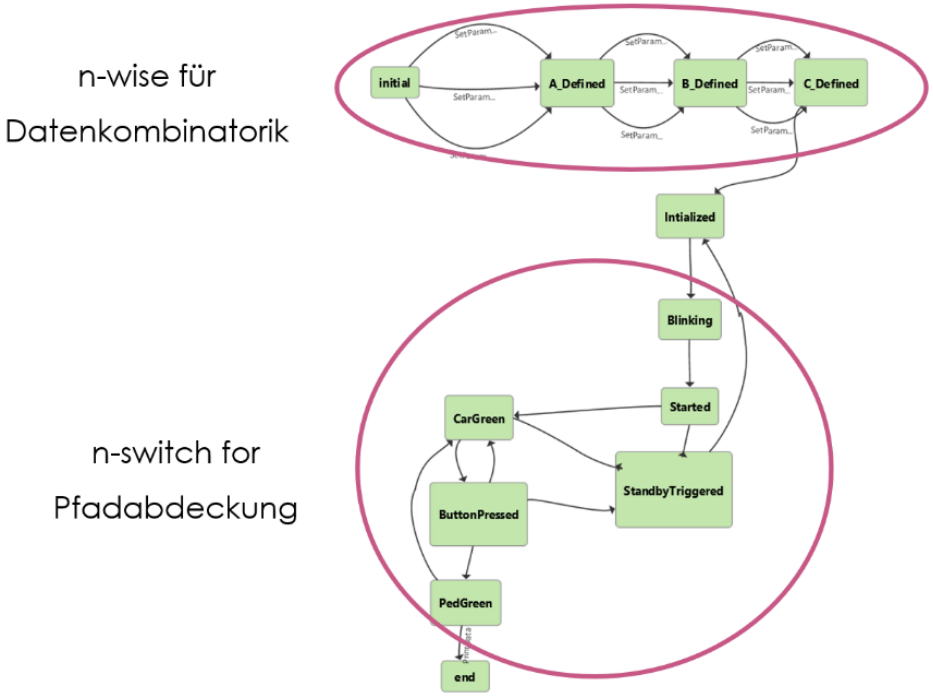

Abb. 4: Kombination von Daten und Pfadabdeckung

Analyse und Dokumentation von Testabläufen

Um die entstehenden Test Cases zu analysieren, zu verstehen und zu dokumentieren, können unterschiedliche Sichten verwendet werden.

- *Ausführungsbäume* liefern einen Überblick über alle generierten Testpfade und ihre unterschiedlichen Abläufe
- *Sequenzdiagramme* stellen sehr detailliert einzelne Testabläufe und ihre Interaktion mit dem System under Test dar. Ein Soll/Ist Vergleich der Sequenzen ermöglicht die schnelle Analyse und Eingrenzung von Fehlerursachen

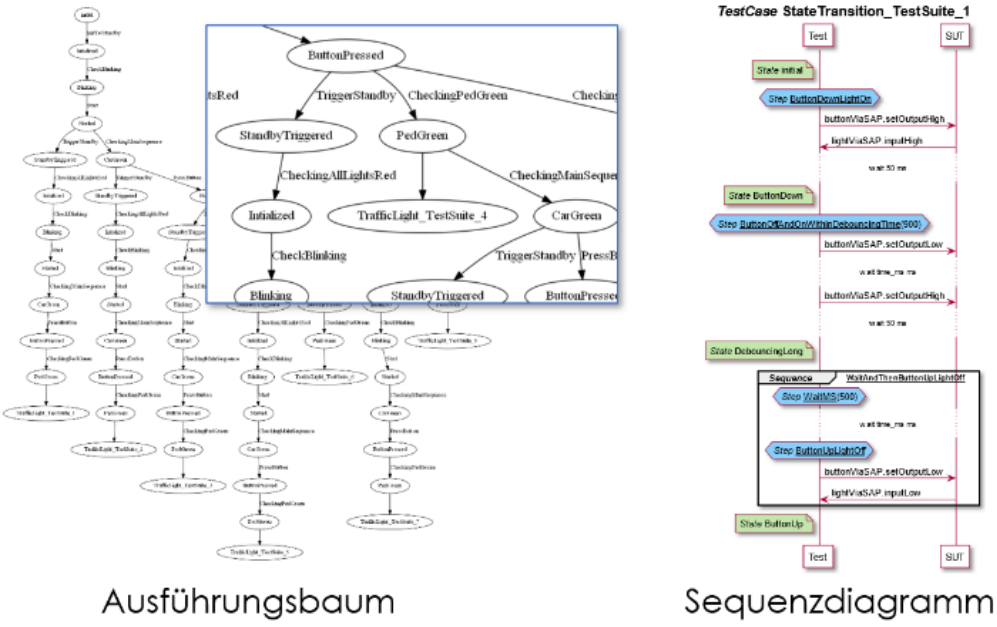

Abb. 5: Unterschiedliche Sichten auf die erzeugten Test Cases

Fazit

- State Transition Tests sind eine strukturierte Methode zur Entwicklung von Tests für nebenläufige Embedded Systeme.
- State Transition Tests ermöglichen die Entwicklung von kombinatorischen Test Cases für hohe Pfad und Datenabdeckungen
- Geringe Abdeckungen können manuell entwickelt werden. Für höhere Abdeckungen empfiehlt sich der Einsatz von Werkzeugen.

Autor

Thomas Schütz studierte Luft- und Raumfahrttechnik in München und gründete 1997 die PROTOS Software GmbH. Als Softwareprojektleiter oder Architekt konnte er seine Erfahrung in der Verbindung modellbasierter Ansätze mit den Anforderungen von Embedded Systemen in zahlreiche Projekte einbringen. Thomas Schütz berät Firmen beim Aufbau von domänenspezifischen Werkzeugketten und Testsystemen für Embedded Systeme und ist Projektleiter des Eclipse Projektes eTrice.

Kontakt

Internet: http://www.protos.de
E-Mail: thomas.schuetz@protos.de