

## TITELSTORY

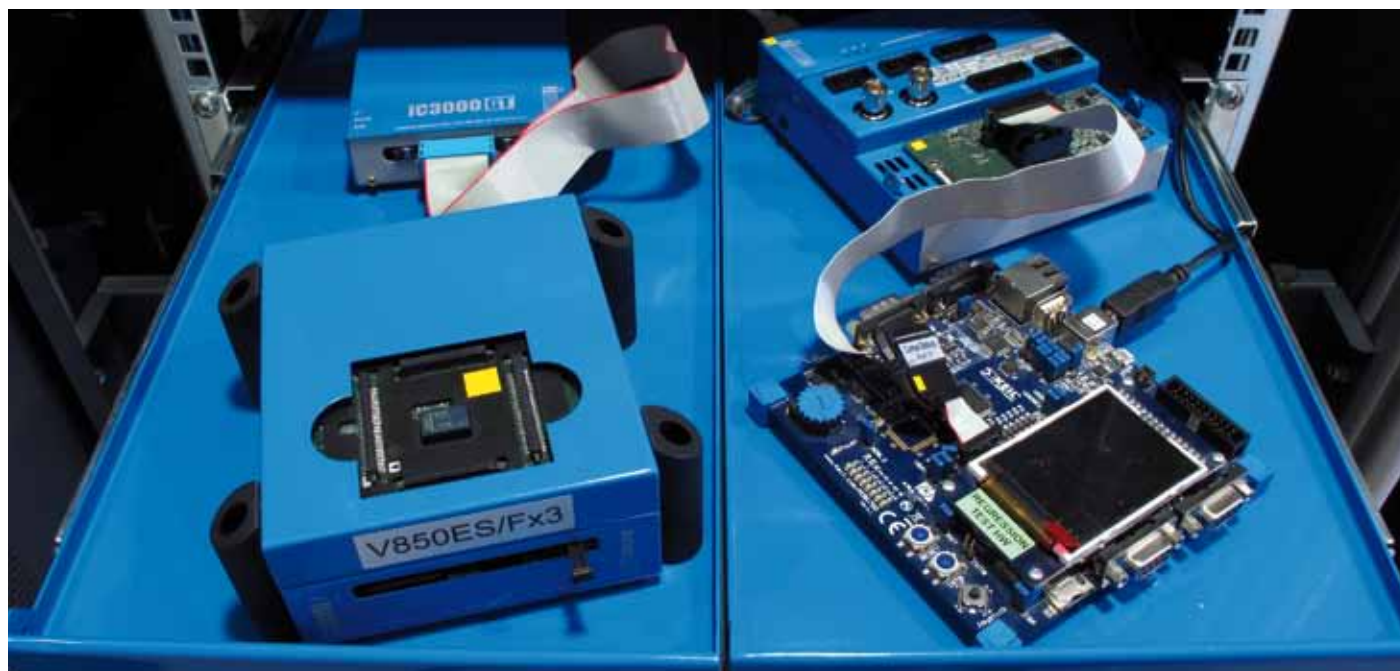
Das Vertrauen in den Umgang mit Softwarewerkzeugen ist Thema des Kapitels 8-11 der Norm ISO26262, einer für Automotive angepassten Version der IEC61508. Außer auf die Automobilhersteller und deren Zulieferer hat diese Norm konkrete Auswirkungen auf die Entwicklungs- und Testprozesse der Werkzeughersteller und damit auf den Entstehungsprozess des Software-Tools selbst.

Dieser Beitrag beschreibt die organisatorischen und technischen Maßnahmen, die iSYSTEM aus aktuellen Normen der Funktionalen Sicherheit abgeleitet hat, um die Softwareentwicklungsabteilungen in der Automobilindustrie im Vorfeld und bei der Tool-Qualifizierung zu unterstützen.

# Wie prüft ein Hersteller von Testwerkzeugen seine Produkte?

*Für sicherheitskritische Anwendungsfelder müssen die Tool-Hersteller ihre Softwarewerkzeuge rigorosen Tests unterziehen. Von dieser Erfahrung können auch die Kunden profitieren.*

EROL SIMSEK \*



**A**usgangspunkt für qualitativ hochwertige Soft- und Hardware sind interne Prozessdefinitionen, die genau beschreiben, wie diese Produkte entwickelt werden müssen. Im Hardwarebereich werden optische Testsysteme und Boundary-Scan-Tests bereits seit Jahren eingesetzt und um sogenannte dynamische Tests erweitert. In diesen weiterführenden Tests werden Speicher, Speichzugriffszeiten und Peripherie gleichermaßen geprüft. Hier stellt iSYSTEM sein Know-how auch Partnerfirmen zur Verfügung.

Das Generieren von neuer Software, Archivieren von alten und Testen von bestimmten Softwareständen sowie ein funktionierendes

Versionskontrollsystem stehen beim Testen der Software im Mittelpunkt. Durch die steigende Komplexität der Software, immer neue Microcontroller und Erweiterungen des Funktionsumfangs ist eine Automatisierung der Tests erforderlich. Eine weitere Motivation für die Testautomatisierung sind die funktionalen Sicherheitsstandards, die intensive Tests fordern. Damit schafft der Tool-Hersteller eine Vertrauensbasis für seine Produkte.

## Testsuite wird mit Skriptsprache Python angesprochen

iSYSTEM lebt die Testautomatisierung mit fitIDEA, einer Test-Toolsuite, die vollautomatisch die definierten Tests in den Embedded-Systemen ausführt. fitIDEA wurde ursprünglich für interne Zwecke entwickelt. Es benutzt die API (Application Programming Interface) der Entwicklungsumgebung winIDEA (*isystem.connect*) über die Skriptsprache Python. Hierüber werden alle kritischen

Aspekte des Embedded-Debuggers und der integrierten Unit-Test-Toolsuite getestet. Folgende Funktionen werden u. a. geprüft:

- Standard-Debugging- und IDE-Funktionen wie Download in den Flash-Speicher, Zielspeicherspeicherzugriff (Lesen, Schreiben, Echtzeitzugriff), Breakpoint-Funktionen (Ausführung und Zugriff),
- Spezielle Zeitmessungen zum Test von weiterführendem Debugging wie Trace und Profiler-Funktionalität,
- Software-Test mit Code Coverage und Unit-Test,
- API-Funktionalität.

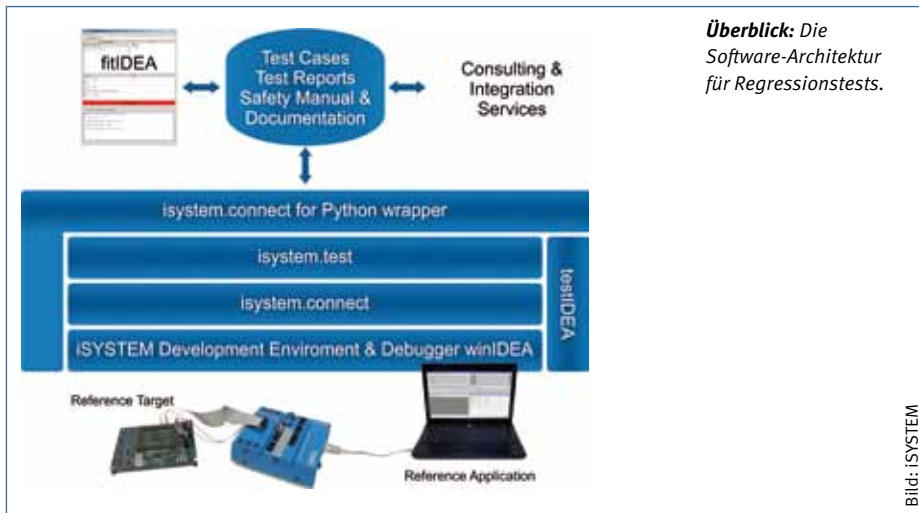
Das Konzept der Toolsuite basiert auf dem deterministischen Verhalten der Testumgebung, die aus diesen Komponenten besteht:

- Referenzsystem (Standard-Board vom Halbleiterhersteller, iSYSTEM-Board, Kundensystem),
- Referenzapplikation für solch ein Board/System,



\* Erol Simsek  
... ist Vorstandssprecher der iSYSTEM AG mit Sitz in Schwabhausen in der Nähe von München.





Überblick: Die Software-Architektur für Regressionstests.

tion nichts mehr im Weg. Es werden kontinuierlich neue Testfälle hinzugefügt, deren Basis entweder neue Funktionen, Anwendungsfälle von Kunden oder gelöste Supportfälle sind. Die Benefits liegen auf der Hand:

- Neue Software kann schneller verifiziert werden,
- Stabilere Software-Traceability und Dokumentation,
- CEO und CTO schlafen ruhiger,
- Transparenter Testprozess für den Kunden.

### Kunden können komplette Testumgebung nutzen

Der Kunde kann fitIDEA im eigenen Unternehmen einsetzen. Mit der Ausführungsplattform erhält er zusätzlich ein Referenzboard und eine Referenzapplikation mit den zugehörigen Testskripten für den jeweiligen Microcontroller. Dieses Referenzdesign mit den zugehörigen Testskripten wird Schritt für Schritt auf die Hard- und Software der Kunden übertragen. Die Tests werden entsprechend erweitert und angepasst.

Am Ende besitzt der Kunde eine komplette Testumgebung für seine Hardware und Applikationssoftware, in der er immer wieder alle Komponenten sehr schnell und effizient testen kann. Der Test nach Austausch oder Veränderung einer Komponente wie einer neuen Compiler- oder einer neuen Hardwareversion ist schnell umzusetzen. Abweichungen sind sofort ersichtlich.

Standards der funktionalen Sicherheit verankern interessanterweise mehr und mehr die Qualifizierung von Softwarewerkzeugen. Das bedeutet, dass man sich im Vorfeld einer Entwicklung über das Risiko des Einsatzes von Softwarewerkzeugen auf bestimmte Anforderungen und „Use Cases“ hin Gedanken machen muss. Der Tool-Hersteller wird hierbei nicht ausgeklammert. Das heißt: Es ist zu erwarten, dass diese Hersteller entsprechende Verbesserungen und Erweiterungen der eigenen Entwicklungs- und Test-Prozesse implementieren. iSYSTEM hat seine Prozesse bereits auf diese Entwicklungen hin ausgerichtet und lebt einen agilen Softwareentwicklungsprozess mit einer transparenten Testautomatisierung. // FG

iSystem: +49 (0)8138 69710

### InfoClick

- iTAG.FIFTY Test-Toolsets zu gewinnen!
- Wie sich Software ohne Testfälle testen lässt
- Hardware-Baugruppentests beschleunigen

www.elektronikpraxis.de

InfoClick 3712666

- Debugging-Hardware (Blue Box),
- Software (winIDEA und testIDEA auf der PC-Seite),
- Compiler.

Das Schlüsselement ist die Referenzapplikation für das jeweilige Zielsystem, die so geschrieben ist, dass ein deterministisches (definiertes und reproduzierbares) Verhalten aufgezeigt wird. Zum Beispiel:

```
char g_c;
void Increment()
{
    g_c++;
}
```

Die Ausführung dieser Funktion erhöht offensichtlich die Variable `g_c` um eins. fitIDEA macht nun Folgendes:

- Führe den Code bis zum Aufruf von `Increment()` aus,
- Lies den Wert der Variablen `g_c`,
- Führe die Funktion bis zum Ende aus,
- Lies den Wert der Variablen `g_c`.

Der Test war erfolgreich, wenn das zweite Lesen der Variablen `g_c` einen um eins erhöhten Wert anzeigt. Folgende Funktionen wurden mit diesem Test abgedeckt:

- Erkennen von Code- und Datensymbolen aus der Download-Datei (setze Breakpoints auf symbolische Werte, lies Variablenwerte über symbolische Namen),
- Unterbrechung der Ausführung (ein Breakpoint wurde auf den Funktionseintritt und -austritt gesetzt),
- Ausführung des Codes und Überprüfung

des CPU-Status (`Run Until`, Warten, bis die CPU stoppt),

- Speicherlesezugriff (Lesen der Variablenwerte),
- C-Compiler: Code- und Debuginformationen wurden generiert.

Wenn sich nun eine Komponente der Testumgebung ändert (iSYSTEM-Tool, Compiler, Zielsystem und Ähnliches), muss sich die Applikation immer noch gleich verhalten und das Test-Tool muss weiterhin das gleiche Resultat ausgeben.

Die Python-Testskripte werden von Testingenieuren erstellt. Dabei werden in der iSYSTEM-Referenzapplikation Referenzkommandos (`//rt`) benutzt, um erwartete Ergebnisse eines Tests zu speichern. Diese Referenzen werden im Quellcode hinterlegt und sind somit unabhängig vom benutzen Compiler:

```
char g_c;
void Increment()
{
    g_c++; //rt step_and_evaluate: value[+1]
}
```

Bis jetzt sind über 60 Hardwarekonfigurationen in den Testständen aktiv, und es sind ca. 100 Testfälle pro Konfiguration vorhanden. Jeder Entwickler hat Zugriff auf alle Konfigurationen und kann auch lokal Tests vornehmen, bevor er seinen geänderten Programmcode übernimmt. Die automatischen Tests werden über Nacht ausgeführt. Die Testergebnisse sind am nächsten Morgen verfügbar. Somit steht einer nahtlosen Integra-

„Alle Testmethoden sind irgendwie interessant. Allerdings muss die Methodik auch zum Unternehmen passen und umgekehrt.“

Erol Simsek, iSYSTEM

## INTERVIEW MIT EROL SIMSEK

## Testtools müssen einfach in den Prüfprozess integrierbar sein

**Welche Herausforderungen für Entwickler und Tester beantwortet die Toolsuite fitIDEA?**

fitIDEA ist zunächst einmal die interne Umsetzung des iSYSTEM Softwareentwicklungs- und Testprozesses. Wir entwickeln in großen Teilen agil und leben den Prozess der Continuous Integration. Hierfür sind wir das Thema Testautomatisierung konsequent angegangen. Des Weiteren ist fitIDEA eine Umgebung zur Nachweiserbringung der Korrektheit von iSYSTEM-Toolfunktionalitäten gemäß bestimmter Use Cases, meist kundenspezifisch. Im Rahmen von Standards der funktionalen Sicherheit wird dies zum Teil verlangt.

**Was macht ein gutes Testwerkzeug aus?**

Das ist schwer zu verallgemeinern. Es gibt ja unterschiedlichste Werkzeuge zur Abdeckung verschiedenster Methoden: vom Modellierungstool über statische Codeanalyse, Unit-/Integrationstestwerkzeuge und nicht zu vergessen Systemtesttools. Meist sind es Kombinationen aus Software und Hardware. Am Wichtigsten ist sicherlich die Einfachheit eines Werkzeugs in der Bedienung und der Integration in den Prozess. Dann schafft man Akzeptanz bei Entwicklern und Testern sowie im Management.

**Wie kann man die Komplexität heutiger Softwareanwendungen und die Gerätevielfalt testseitig in den Griff bekommen?**

Eine Antwort hierfür ist fitIDEA und die gelebte Testautomation im agilen Softwareentwicklungsumfeld. Sprich: Automation und moderne Methodik. Und ab und zu in Richtung Luftfahrt schauen. Hier gibt es viel Erfahrung im Bereich Modellbasiertes/Requirement-basiertes Testen.

**Es gibt unterschiedliche Ansätze und Philosophien, was Softwaretests angeht. Welche ist für Sie die interessanteste?**

Interessant sind irgendwie alle. Allerdings muss die Methodik auch zum Unternehmen passen und umgekehrt. Agile Softwareentwicklung und die damit verbundene Philosophie des Testens finde ich zur Zeit sehr interessant. Insbesondere deshalb, weil viele eingefahrene Vorgehensweisen und Verhaltensmuster aufgebrochen werden und recht frisch und munter sowie professionell an die Lösung von Softwareherausforderungen jeglicher Art herangegangen wird.

**Manche Experten erwarten für die Zukunft, dass die Bedeutung von statischen Analysewerkzeugen steigt. Wird dadurch die Bedeutung von Tests abnehmen?**

**Im Gespräch:** Erol Simsek, Vorstandssprecher der iSYSTEM AG, über die Notwendigkeit von Tests im Embedded-Umfeld und die Integration von Tests in den Entwicklungsprozess.

Stimmt! Die Bedeutung bzw. der Einsatz solcher Tools hat in den letzten 10 Jahren stark zugenommen und nimmt jetzt so richtig Fahrt auf. Solche Werkzeuge sind prinzipiell einfach in einen bestehenden Entwicklungs- und Testprozess einzubauen, ohne den Prozess stark verändern zu müssen. Statische Analyse-Tools decken einen Teil des Testprozesses ab und sind eine sinnvolle Ergänzung zur Erhöhung der Testtiefe.

## Spezial-Newsletter für Elektronik-Profis



**Tages-Newsletter** mit den Nachrichten der letzten 24 Stunden aus der Elektronikbranche

**Wochen-Newsletter** mit den besten Fachbeiträgen und Expertentipps für Entwicklung und Fertigung

kostenlos unter  
[www.elektronikpraxis.de/newsletter](http://www.elektronikpraxis.de/newsletter)

ELEKTRONIK  
**PRAXIS**