

Unsere Leistung

UNIT TEST FRAMEWORK FÜR C UND C++

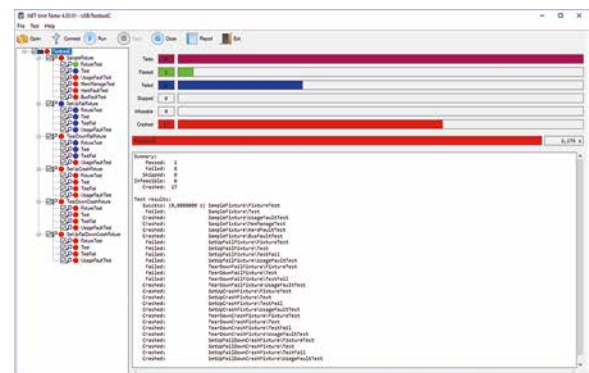
Unit Tests gehören mittlerweile zu den Standardmaßnahmen im Softwareentwicklungsprozess. Mit ihrer Hilfe wird die Fehlerquote einzelner Module reduziert und die spätere Integration vereinfacht. Unit Tests erlauben die Überprüfung von Fehlerkonditionen, die bei einem Systemtest nicht erreicht werden können oder dürfen.

FORTech setzt seit der Publikation der ersten Frameworks Unit Test im eigenen Entwicklungsprozess ein. Aufgrund der großen Diversität der Zielumgebungen im Embedded-Umfeld setzen wir hier auf ein eigenes, einfach und pragmatisch gehaltenes Framework für C und C++, das sich an die Konzepte von CppUnit anlehnt.

```
4 static void FilterTest1()
5 {
6     uint16_t resultLength;
7     float *resultData;
8     uint2_t windowWidth = 3;
9     float expectedData[] = {1.833, 2.5, 2.6, 2.866, 2.366, 2.3};
10    enum EResult result = FilterData(windowWidth, &resultData, &resultLength);
11
12
13    // filter should deliver 6 result values
14    AssertEqual(Success, result);
15    AssertEqual(resultLength, 6);
16
17    for (int i = 0; i < 6; i++)
18    { // check averaged values using epsilon of 0.001
19        Assert(fabs(resultData[i] - expectedData[i]) < 0.001);
20    }
21
22 }
```

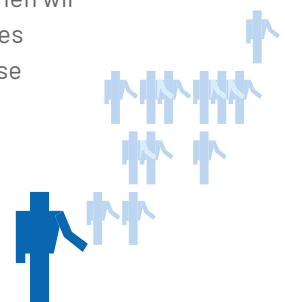
Im Gegensatz zu komplexen, integrierten Testsystemen gehen wir davon aus, dass Testbarkeit bereits mit dem Design beginnt. Module verwenden Schnittstellen, die einen isolierten Test ermöglichen. Dadurch kann der Produktcode für C und C++ in statischen Bibliotheken getestet werden, die in unveränderter Form im Produkt ausgeführt werden. Fehlerquellen wie die Verwendung unterschiedlicher Compileroptionen können so ausgeschlossen werden. Testergebnisse werden in einem JUnit-kompatiblen Format erzeugt, das eine einfache Integration in Continuous-Integration-Systeme wie Jenkins/Hudson erlaubt.

Auf leistungsfähigeren Zielsystemen mit eigenem Betriebssystem und Shell-Funktionalität wie etwa Windows Embedded Compact verwenden wir typisch skriptgesteuertes Deployment und Ergebnistransfer.



Für kleine Mikrocontrollerapplikationen zum Beispiel auf Basis der ARM-Cortex M0...M4 wird die Teststeuerung und -auswertung auf die Entwicklungsmaschine verlagert. Auf den Mikrocontroller werden nur der zu testende Code, die Testfälle und ein minimalistischer Server geladen. Die für die Kommunikation verwendete Schnittstelle richtet sich nach den Ressourcen des Zielsystems (USB, CAN, LIN, ...) und kann angepasst werden. Der Entwicklungsrechner steuert über eine grafische Oberfläche oder Kommandozeile die Testausführung. Entwicklungsumgebung und Teststeuerung arbeiten dabei parallel, so dass die gezielte Ausführung einzelner Unit Tests im Debugger möglich ist.

Wir unterstützen Sie gerne bei der Anpassung unserer Tools an Ihre konkrete Entwicklungsumgebung. Bei Bedarf stehen wir Ihnen bei der Integration des Unit Testing in Ihre Prozesse zur Verfügung.

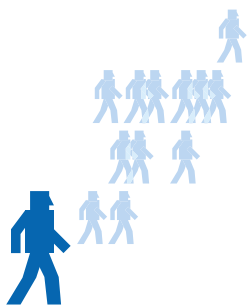


Unser Unternehmen

FORTECH SOFTWARE GMBH

Kurz gesagt

- Individuelle Softwarelösungen seit 1990 aus der Hansestadt Rostock
- Hochqualifiziertes Team von Elektrotechnikern, Informatikern und Physikern
- Qualitätsmanagement-System zertifiziert nach ISO 9001:2015



Kontakt

FORTECH Software GmbH

Tannenweg 22 m · 18059 Rostock

Telefon: +49 381 496800-0

Telefax: +49 381 496800-29

info@fortech.de · www.fortech.de



Mitgliedschaften

