

Aktuell wird an einem solchen E/A-Modul gearbeitet, das das Umwandeln von VHDL-Netzlisten in XML und umgekehrt ermöglicht. In einem zweiten Schritt wird die vMAGIC-Bibliothek das Auflösen von Generate-Statements beherrschen, so dass auch komplexe Designs eingelesen und als flache Netzlisten ausgegeben werden können.

- » Java: Ein weiteres Ausgabemodul wird für Java definiert, so dass eine VHDL-Datei in vMAGIC-Code überführt wird, der genau diese Datei erzeugen kann. Auf den ersten Blick sinnfrei erschließt sich der Nutzen erst dann, wenn man bereits mit Templates gearbeitet hat. Alle Objekte aus einem Template, auf die zugegriffen werden soll, müssen erst „gesucht“ (siehe Abbildung 1.04, Zeile 2) werden; mit Hilfe des Java- bzw. vMAGIC-Ausgabemoduls müssen Templates nicht mehr geparkt werden, sondern liegen sofort im Java-Quelltext vor und können direkt verwendet werden. Auch müssen häufig nur kleine Änderungen an dem so generierten Java- Quelltext vorgenommen werden, um mächtige VHDL-Generatoren zu entwickeln. Dadurch reduziert sich der Aufwand bei der Erstellung von vMAGIC-Anwendungen erheblich.
- » **Editor: Der effektivste Weg, um einem VHDL-Entwickler die Möglichkeiten von vMAGIC zu eröffnen,** liegt in der Entwicklung eines modernen VHDL-Editors. Aufbauend auf der NetBeans-Plattform wird ein Editor mit vielen Funktionen aus der Software-Welt entwickelt. Zusätzlich werden viele vMAGIC

Funktionen eingesetzt, um den Hardwareentwurf deutlich zu beschleunigen.

#### Fazit

vMAGIC hat das Potential, in vielen Bereichen der Hardwareentwicklung gewinnbringend eingesetzt zu werden. Von allgemein einsetzbaren Werkzeugen wie einem leistungsfähigen VHDL-Editor bis hin zu Spezialanwendungen wie einem Code-Generator können viele neue Lösungen entwickelt werden, denen eines gemein ist: sie verkürzen die Entwicklungszeit und führen zu verlässlicherem Code. Der Mehraufwand, der beim Programmieren einer vMAGIC-basierten Lösung entsteht, wird durch die Wiederverwendbarkeit schnell kompensiert. Derzeit wird die Bibliothek im Rahmen zahlreicher universitärer und industrieller Projekte eingesetzt, so auch an den Universitäten Paderborn und Bielefeld, hauptsächlich für die automatische Erzeugung von Teststrukturen. Für weitere Projekte möchten wir erneut auf die Projektwebseite hinweisen. vMAGIC ist freie Software und in seiner Funktion absolut einzigartig, daher bleibt uns an dieser Stelle nur, auf das Potential dieser Bibliothek hinzuweisen: give it a try!

[1] Pohl et al., vMAGIC – Automatic Code Generation for VHDL, *International Journal of Reconfigurable Computing*, 2009

[2] vMAGIC Webseite, <http://vmagic.sf.net>



## Prozessoptimierung mit IMPACT

Eine Projektidee zur Entwicklung von Methoden und Werkzeugen für Planung, Management und Optimierung von Entwicklungsprozessen

**Im Rahmen der Prozessmodellierung und -analyse zur optimalen Gestaltung von Entwicklungsprozessen steht die Frage im Mittelpunkt, wie man mit minimalem Aufwand möglichst genaue Planungen erstellt. Ziel des werkzeuggestützten IMPACT-Ansatzes ist die Entscheidungsunterstützung von Prozessgestaltern und Projektmanagern durch die Erstellung sogenannter „IMPACT-Modelle“ als Elemente der Prozessplanung insbesondere zur optimalen Anwendung neuer Methoden und Werkzeuge.**

#### Motivation

Für die Handhabung ständig steigender Produktkomplexität werden regelmäßig neue Methoden und zugehörige Werkzeuge entwickelt. So werden in der Hardwareentwicklung Beschreibungssprachen wie SystemC oder neue modellbasierte Verifikationsmethoden zur Durchführung von Strukturtests eingeführt. Auch formale Verifikationsmethoden wie Model Checking eignen sich für zahlreiche Verifikationsaufgaben, wenn eine gezielte und korrekte Abstraktion der zugrundeliegenden Modelle möglich ist. Um zu entscheiden, ob eine bestimmte Methode im Entwicklungsprozess effektiv anwendbar ist, wird exaktes Wissen über notwendige Einsatzbedingungen benötigt.

In der Studie CONTROL, die aktuell von OFFIS durchgeführt wird, gaben bislang ca. 60 Unternehmen an, dass ihr Planungshorizont in den durchgeführten IT-Projekten bei ca. 3 Monaten liegt.

Kontrolliert man nach drei Monaten jeweils Kosten und

**newsletter edacentrum - Probeauszug**  
Bestellen Sie sich den kompletten Artikel  
über [newsletter@edacentrum.de](mailto:newsletter@edacentrum.de)

edacentrum, Hannover, Juni 2010

**Autoren:**  
Stefan Häusler  
(OFFIS),  
Stephan Radke  
(Fraunhofer IIS/EAS)