



In der Medizintechnik kommt zunehmend die modellbasierte Softwareentwicklung zum Einsatz, um Komplexität und Kosten zu beherrschen.

Modellbasierte Software-Entwicklung

Die Erwartungen an eine sichere und intuitive Nutzung medizinischer Geräte erhöhen den Entwicklungsaufwand maßgeblich. Neben regulatorischen Bestimmungen auf nationaler und internationaler Ebene wird die Systemkomplexität durch steigende Funktionsumfänge erhöht, welche durch größere verfügbare Rechenleistung möglich werden. Beispiele hierfür sind Vernetzung und moderne Benutzerschnittstellen. Unter Berücksichtigung wachsenden Kostendrucks und kürzeren Entwicklungszyklen werden Methoden benötigt, die eine effiziente Softwareentwicklung ermöglichen.

Diesen Herausforderungen gewachsen ist die modellbasierte Entwicklung, die über den gesamten Softwareentwicklungsprozess eingesetzt werden kann. Aufgrund des hohen Automatisierungsgrads, sowohl bei der Code-Erzeugung als auch beim Testen, erleichtert sie die Umsetzung von Nutzeranforderungen über alle Entwicklungsschritte hinweg nachzuvollziehen. Damit wird die Traceability, wie sie zum Beispiel durch IEC 62304 und ISO 14971 gefordert wird, erleichtert. Der modellbasierte Entwicklungsansatz fördert den modularisierten Entwurf von Software und trägt dadurch zu wiederverwendbaren Modulen und einer verbesserten Wartbarkeit des Endproduktes bei. In der Praxis zeigt sich, dass ein strukturelles Redesign und Anpassungen der Software-Architektur im Modell leichter als im Code möglich sind.

Erfahrungswerten bisheriger Projekte zufolge können durch die sich daraus ergebende leichtere Einarbeitung neuer Mitarbeiter und effizientere Pflege bestehender Produkte die Kosten für Entwicklung und Wartung um 20-30% reduziert werden. Darüber hinaus können schon in einem sehr frühen Entwicklungsstadium durch Simulation und Testen des Modells Erkenntnisse über die Qualität des Entwurfs und der Implementierung gewonnen werden. Die Erkennung und Behebung von Fehlerursachen zu einem möglichst frühen Zeitpunkt im Entwicklungsprozess trägt ebenfalls maßgeblich dazu bei, Entwicklungskosten einzusparen.

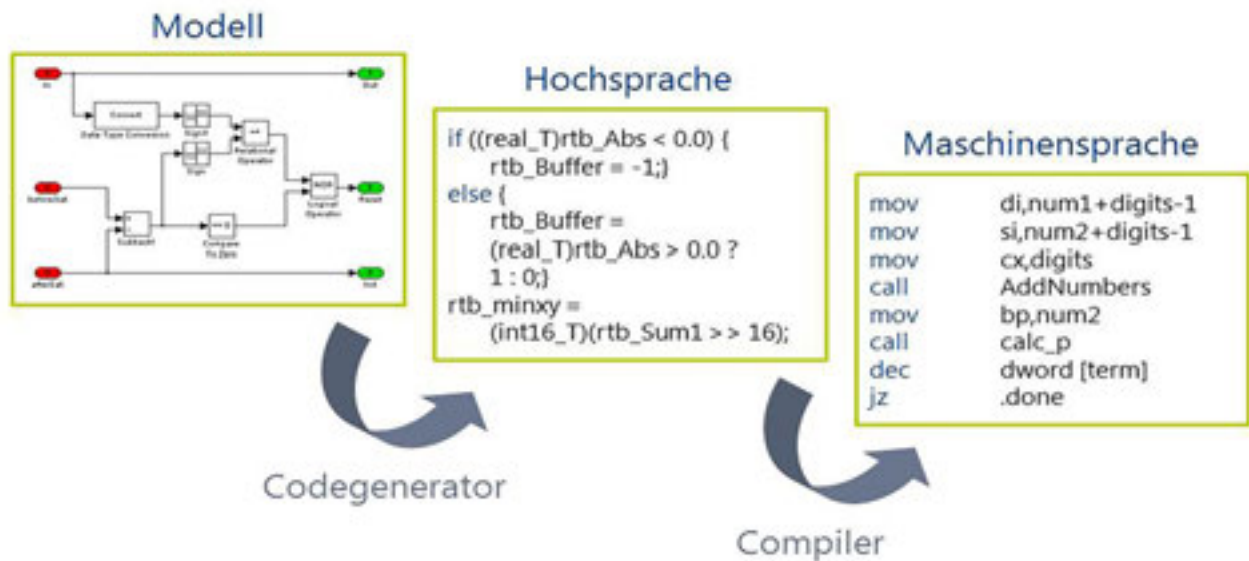
In anderen Branchen wie z. B. Automotive, Luft- und Raumfahrt wird die modellbasierte Entwicklung schon lange und umfangreich eingesetzt.

Modellbasierter Entwicklungsprozess

Modellbasierte Software-Entwicklung sieht vor, die zu entwickelnde Software in einer graphischen Sprache formal zu spezifizieren. Dies gilt sowohl für die Architektur als auch für die konkrete Umsetzung von Anforderungen.

Durch die graphische Spezifikation entsteht eine zusätzliche Abstraktionsebene über den aus der klassischen Software-Entwicklung vertrauten Ebenen des Maschinen-Codes (Assembler) und Programmcode in einer Hochsprache (z. B. C). Die graphische Darstellung abstrahiert beispielsweise von der verwendeten Hardware oder der konkreten Umsetzung von Arithmetik-Operationen. Dies fördert ein intuitives Verständnis der entstehenden Software und ermöglicht effektivere Reviews. Die Einarbeitung neuer Mitarbeiter in große Projekte wird so erleichtert und die Fehleranfälligkeit im Rahmen von Erweiterungen oder Wartungsmaßnahmen wird reduziert. Die Umsetzung der graphischen Software-Spezifikation in Programmcode kann analog zur Kompilierung von Maschinencode automatisiert durch einen Codegenerator erfolgen.

Entwicklungswerkzeuge wie z. B. MATLAB, bei denen der Schwerpunkt auf der Simulation liegt, ermöglichen die Erstellung eines physikalischen Umgebungsmodells, um die Wechselwirkung zwischen Software (beispielsweise ein Regelungsalgorithmus) und Umgebung zu simulieren. So können Software-Module, die nicht unmittelbar von der eingesetzten Hardware abhängen, schon entwickelt und getestet werden, bevor die letztendlich zum Einsatz kommende Hardware verfügbar ist. Zudem können Tests auf Systemebene, für die lange Laufzeiten wie mehrere Stunden oder Tage notwendig sind, in der Simulation mit unterschiedlichen Stimuli reproduzier-



Bei der modellbasierten Software-Entwicklung wird der Programmcode aus einem (graphischen) Modell erzeugt.

bar getestet werden. Beispiele hierfür sind Therapieverfahren aus der Schlafmedizin (eine Therapienacht dauert 6-12 Stunden) oder Temperaturregelungen für medizinische Geräte.

Modultest

Für die normkonforme Entwicklung sind Tests auf Modulebene essentiell. Sie bieten die Möglichkeit, die Testabdeckung einer Software zu erhöhen, da Zustände, die bei Systemtests nur schwer oder gar nicht getestet werden können, gezielt herbeigeführt werden können. Die Durchführung kann bei der modellbasierten Entwicklung sowohl im Modell als auch mit dem daraus generierten Code erfolgen. Im ersten Fall spricht man von MiL (Model-in-the-Loop) Tests, im zweiten von SiL (Software-in-the-Loop) Tests. Durch den Vergleich der Ergebnisse, die MiL und SiL Tests liefern, kann auch nachgeprüft werden, ob das Verhalten des automatisch erzeugten Codes dem im Modell spezifizierten Verhalten entspricht. Stimmt das Verhalten des Modells mit dem des generierten Codes überein, hat der Codegenerator ein valides Ergebnis geliefert. PiL (Processor-in-the-Loop) Tests ermöglichen schließlich die Verifikation des Verhaltens auf der Ziel-Hardware.

Automatisierte Testprozesse

Für den isolierten Modultest ist ein Umgebungsmodell notwendig, das die notwendigen Eingangswerte bereitstellt und die Ausgangswerte auf vorgegebene Akzeptanzkriterien hin auswertet. Für das Anlegen, die Pflege und die effiziente Stimuli-Erzeugung kommen Automatisierungswerkzeuge zum Einsatz. Wiederkehrende Testaufgaben können auf diese Weise reproduzierbar und effizient durchgeführt werden. Mit dem automatisierten Vorgang können zudem vereinbarte Konventionen für die Gestaltung und Dokumentation von Tests mit geringem Aufwand umgesetzt werden.

Die Automatisierung von Testerstellung, -durchführung und -auswertung reduziert die Zeit, die für das Testen der Systeme veranschlagt werden muss. Da schon in einem frühen Entwicklungsstadium am Modell umfangreich getestet werden kann, wird das Risiko reduziert, dass Fehler erst bei Validierung und Markteintritt am Projektende erkannt werden.

Mit modellbasierter Software-Entwicklung kann medizinische Software effizient und normkonform erstellt werden. Im Vergleich zur klassischen Software-Entwicklung erweist sich besonders die Möglichkeit frühzeitiger Tests als wesentlicher Vorteil bei der Qualitätssicherung. Leistungsstarke Werkzeuge unterstützen die verschiedenen Schritte des Entwicklungsprozesses und gewährleisten dessen Nachvollziehbarkeit und Durchgängigkeit. ■



Autor:
Dr.-Ing. Jochen Furthmüller, Senior Engineer
Medical Software Engineering



Autor:
Dr.-Ing. Michael Schwarz, Senior Engineer
Medical Control Systems and Model-Based Design



ITK Engineering GmbH
Im Speyerer Tal 6
D-76761 Rülzheim
Tel. +49 7272 77 03
www.itk-engineering.de