Seit Ende 2011 ist die Nachfolgenorm der DO-178B, die DO-178C, veröffentlicht und zeichnet sich gegenüber ihrem Vorgänger besonders dadurch aus, dass neuen Methoden zur Software-Entwicklung in Form von Standardergänzungen breiter Raum eingeräumt wird. Ganz wesentlich sind dabei die Standarderweiterungen zu Techniken des modellbasierten Entwurfs und der modellbasierten Verifikation, die im Supplement-Dokument DO-331 beschrieben sind. Diese Schlüsseltechniken des Software-Entwurfs bieten viel Potenzial, die Software-Entwicklung im Luftfahrtbereich hocheffizient zu gestalten und die hohen Qualitäts- und Sicherheitsansprüche an die Software nicht nur zu halten, sondern noch auszubauen. Im Folgenden wird näher gezeigt, wie sich der Einsatz von TargetLink im Kontext der DO-178C/DO-331 darstellt und welche Aspekte dabei zu beachten sind.

Modelle: Türöffner für den Einsatz innovativer Methoden

Ein entscheidender Fortschritt für eine effiziente und qualitätsbewusste Software-Entwicklung ergibt sich aus der Darstellung von Anforderungen durch Modelle entsprechend der DO-331. Ein Übergang von rein textuellen zu formalisierten Anforderungen in Form von Modellen eröffnet eine Fülle von Möglichkeiten zur automatisierten Analyse, Quellcode-Erzeugung sowie Verifikation. Software-Anforderungen gibt es nach DO-178B/C bzw. DO-331 in zwei unterschiedlichen Formen:

- *High-Level Requirements (HLR)*
  Sie beschreiben im Wesentlichen, „was“ die Software tun soll, aber nicht „wie“, sind also eine „Black-Box“-Sicht auf die Software. HLRs werden wiederum aus Anforderungen an das eigentliche System

>>

Code-Generator TargetLink
für Luftfahrtanwendungen

Sicherer Code nach DO-178C

Der dSPACE Seriencode-Generator TargetLink ist nicht nur exzellent für automotive Serienprojekte geeignet, sondern auch für solche in der zivilen und militärischen Luftfahrt. Speziell für die Anwendung von TargetLink in DO-178C-konformen Luftfahrtprojekten bietet dSPACE eine umfangreiche Workflow-Beschreibung an, die den Einsatz einer TargetLink-basierten Werkzeugkette zur vereinfachten Zertifizierung der Software beschreibt.

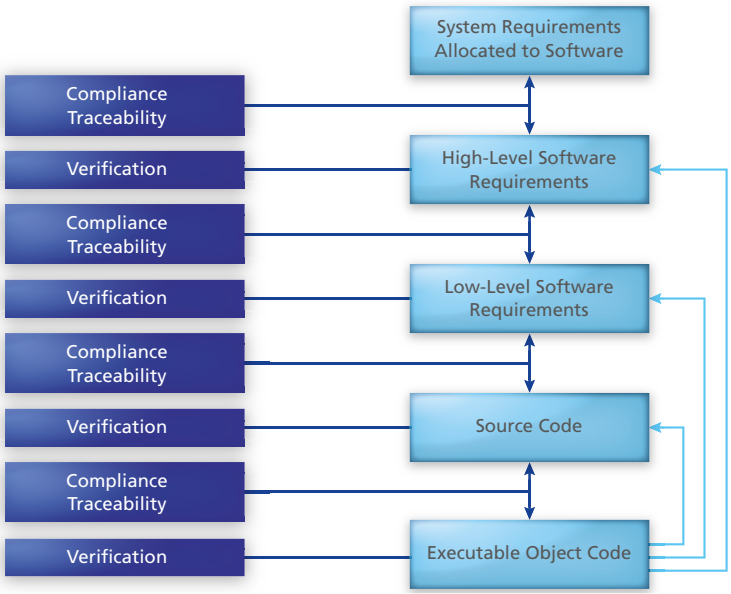


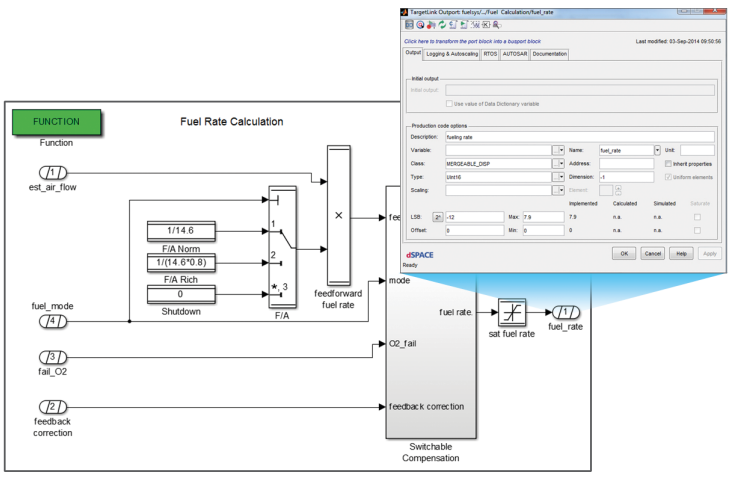
Abbildung 1: Wichtige Entwicklungsphasen entsprechend DO-178C, inklusive erforderlicher Verifikationsschritte.

abgeleitet, die im Systemprozess, etwa nach ARP 4754 (Aerospace Recommended Practice), aufgestellt wurden.

- *Low-Level Requirements (LLR)*
  Diese beschreiben das Innenleben der Software, also das „wie", und geben somit eine „White Box"-Sicht auf die Software. LLRs werden naturgemäß aus den HLRs abgeleitet. Aus LLRs muss sich direkt der eigentliche Quellcode erzeugen lassen.

Modelle zur Repräsentation von Anforderungen können nun auf diesen beiden Ebenen eingesetzt werden (Abbildung 1). Besonders verbreitet sind Simulink®/TargetLink®-Modelle zur Repräsentation von LLRs, aus denen anschließend direkt der eigentliche Quellcode durch automatische Code-Generierung erzeugt wird. Solche Modelle zur Repräsentation von LLRs werden nach DO-331 als Design-Modelle bezeichnet und beinhalten neben der Beschreibung der eigentlichen Funktionalität alle notwendigen Detailinformationen über die Software, wie beispielsweise interne Datenstrukturen, Kontrollflussinformationen und eventuelle Festkomma-Repräsentationen (Abbildung 2).

Abbildung 2: Simulink/TargetLink-Design-Modelle werden direkt zur automatischen Quellcode-Generierung mit TargetLink verwendet.



Vom Design-Modell zum Quellcode auf Knopfdruck

Die Repräsentation von Anforderungen als Design-Modelle (nach DO-331) eröffnet einen direkten Zugang zur Erstellung des Quellcodes für die Software: die Nutzung automatischer Code-Generierung statt manueller Codierung. TargetLink ist in Bezug auf Qualität und Zuverlässigkeit menschlichen Programmierern weit überlegen und produziert völlig deterministisch und auf Knopfdruck Quellcode:

- Der von TargetLink generierte Code ist sehr gut lesbar und geeignet für Reviews. Dies wird durch umfangreiche Quellcode-Kommentierung, leicht verständliche Symbolnamen und die Nutzung eines Sprach-Subsets der Sprache C gewährleistet.
- Er lässt sich direkt zum Entwurfsmodell zurückverfolgen. Hierdurch wird unmittelbar Nachverfolgbarkeit zwischen Quellcode und dem zugehörigen Modell, aus dem der Code erzeugt wurde, hergestellt.
- Der mit TargetLink zu generierende Code ist darüber hinaus in sehr hohem Maße konfigurierbar, um Kodierungsrichtlinien einzuhalten, den TargetLink-generierten Code mit existierendem Legacy-Code zu

verbinden und den zu generierenden Code optimal in die Software-Architektur ohne aufwendige Wrapper einzupassen.

Generell sind Qualität, Konfigurierbarkeit und Effizienz des generierten Codes herausragende Merkmale von TargetLink, die sich in allen Anwendungsbereichen zeigen.

Modellbasierte Verifikation – Schlüssel zur einfacheren Zertifizierung

Die großen Vorteile beim Einsatz von Modellen zur Spezifikation von Anforderungen (HLRs und LLRs) zeigen sich neben der automatischen Code-Generierung insbesondere im Bereich der Verifikationsschritte. Diese müssen entwicklungsbegleitend in den einzelnen Phasen zur Überprüfung der entwickelten Artefakte wie Modelle, Quellcode und Objekt-Code durchgeführt werden (Abbildung 1).
Zum Nachweis, dass die Modelle die Anforderungen erfüllen, aus denen sie abgeleitet wurden (Abbildung 1), bietet sich insbesondere eine Kombination aus Modellsimulationen, Abdeckungsanalyse und Testfallgenerierung an. Testfälle müssen nach DO-178B/C ausschließlich anforderungsbasiert erstellt werden. Ist nun die Anforderung selbst in Form eines Modells spezifiziert, etwa durch ein Simulink/TargetLink-Modell, so können Techniken zur automatischen Testvektorgenerierung angewandt werden, wie sie etwa der BTC EmbeddedTester® bietet. Zum Nachweis der Konformität des ausführbaren Objekt-Codes mit den HLRs und LLRs entsprechend Abbildung 1 wird der Objekt-Code typischerweise auf der Zielplattform ausgeführt. TargetLink bietet hierzu äußerst leistungsfähige Mechanismen in Form von Processor-in-the-Loop-Simulation, um den automatisch generierten Code direkt mit dem Target-Compiler zu übersetzen und auf einem Evaluation-Board des Zielprozessors zur Ausführung zu bringen (Abbildung 3).

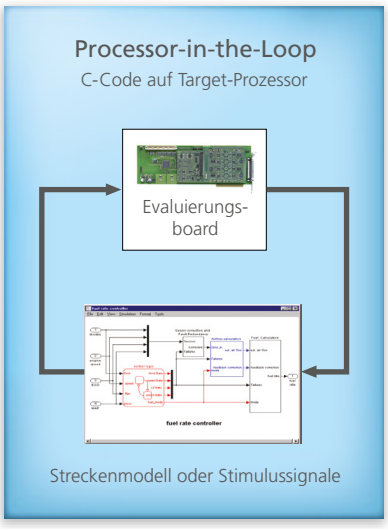


Abbildung 3: Ausführung des „Executable Object Codes" in Processor-in-the-Loop-Simulationen, um nachzuweisen, dass der Objekt-Code die Anforderungen erfüllt.

DO-178C/DO-331-Workflow-Dokument für TargetLink

Für den Einsatz von TargetLink in DO-178C/DO-331-kompatiblen Projekten stellt dSPACE das Workflow-Dokument „TargetLink – Model-Based Development and Verification of Airborne Software" bereit. Das Dokument beschreibt, wie sich die einzelnen Anforderungen bzw. „Objectives" der DO-178C/DO-331 besonders einfach erfüllen lassen. Betrachtet wird hierbei nicht nur TargetLink selbst, sondern die gesamte TargetLink-Umgebung in Form einer kompletten, modellbasierten Werkzeugkette unter Berücksichtigung von Drittanbieterwerkzeugen. Dazu gehören die Werkzeuge der TargetLink-Kooperationspartner BTC Embedded Systems, Model Engineering Solutions und AbsInt. Das Dokument kann per E-Mail an *TargetLink.Info@dspace.de* angefordert werden. ■

Fazit

Für Luftfahrtanwendungen lässt sich mit TargetLink in DO-178C-konformen Projekten hochqualitativer Quellcode auf Knopfdruck erzeugen. Der Code ist aufgrund seiner Kommentierung des Layouts und der Symbolnamen sehr gut lesbar, bietet nahtlose Rückverfolgbarkeit zu den Anforderungen und ist leicht konfigurierbar, zum Beispiel um geforderte Richtlinien zu erfüllen. Außerdem bieten TargetLink und dessen Integration mit Drittanbieterwerkzeugen eine ideale Umgebung für Verifikation, Simulation, Analyse und Tests. Von den Anforderungen bis hin zum fertigen Quellcode – mit TargetLink haben Anwender ihre DO-178C-konformen Entwicklungsprojekte im Griff.