

## Antriebsregler leicht gemacht

*Mit der steigenden Komplexität der Antriebsregelungen wird es notwendig, moderne Entwicklungsmethoden wie Rapid Control Prototyping anzuwenden. Mit Rapid Control Prototyping werden neue Regelungskonzepte entwickelt und am realen System optimiert. Mit graphischen Spezifikationen, kurzen Iterationsschritten und sofort verfügbaren Testergebnissen erreicht man ein optimales Entwicklungsergebnis.*

Elektrische Antriebe haben eine große Bedeutung in nahezu allen Produktionsprozessen. In den letzten 20 Jahren hat der Bereich geregelter elektrischer Antriebe ein schnelles Wachstum erfahren, hauptsächlich durch die Fortschritte der Halbleitertechnologie und vor allem in der Leistungselektronik. Gleiches gilt für digitale Signalverarbeitung und analoge Elektronik in Mikroprozessoren. Elektronisch geschaltete Umrichter haben die Regelung von Drehstrommotoren revolutioniert und so zur fast vollständigen Verdrängung der Gleichstromantriebe geführt.

### **dSPACE Prototyper – eine Lösung für den Entwurf von Antriebsregelungen**

Auch die Entwicklungswerkzeuge haben große Fortschritte erfahren: Design-tools wie MATLAB/Simulink ermöglichen heutzutage den Reglerentwurf direkt im

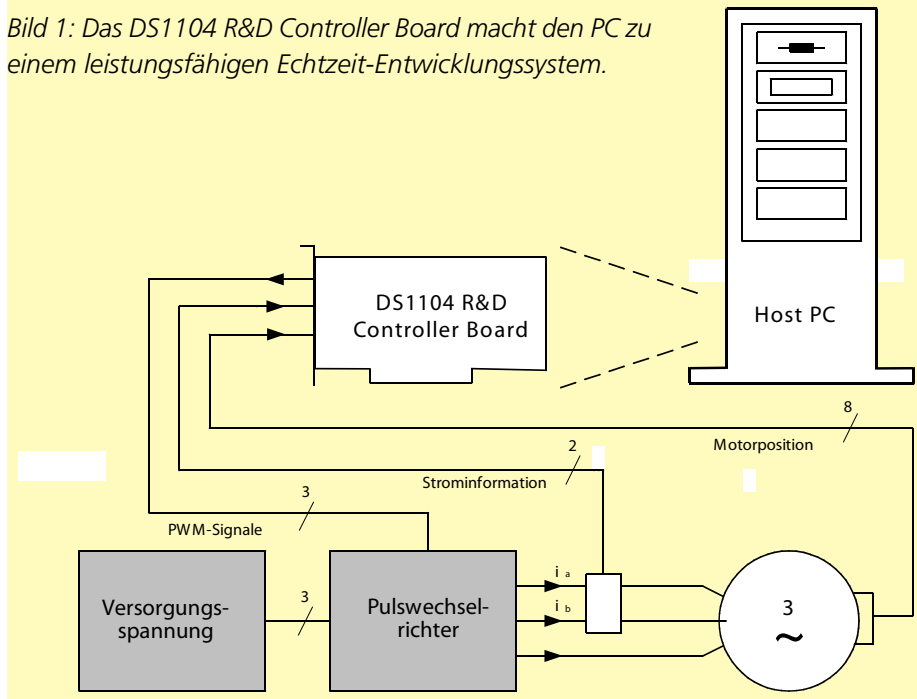
Blockdiagramm und erlauben Offline-Simulationen, die zwar nicht unbedingt in Echtzeit stattfinden, aber trotzdem ein gutes Bild vom zu erwartenden Ergebnis geben. Der nächste Schritt allerdings, der Test an der realen Strecke in Echtzeit, ist oft nur durch Entwicklung eines Hardware-Prototypen und aufwendige manuelle Programmierung möglich.

Genau hier setzt Rapid Control Prototyping an: diese Technik erlaubt es, den Reglerentwurf an der realen Strecke zu testen. Fehlerträchtige manuelle Programmierung und mühsamer Prototypenbau werden auf diese Weise vermieden, was zu enormen Einsparungen an Zeit und Kosten sowie einer massiven Verbesserung der Softwarequalität führt. Rapid Control Prototyping ist in anderen Bereichen, zum Beispiel der Automobiltechnik, schon seit Jahren etabliert und sehr erfolgreich im Einsatz.

Mit „dSPACE Prototyper“, basierend auf dem neuen „DS1104 R&D Controller Board“, bietet dSPACE ein sehr kompaktes, kostengünstiges Entwicklungssystem für Rapid Control Prototyping. Das Entwicklungssystem wurde speziell für Antriebstechnik, Automatisierungstechnik und Robotik entwickelt, aber durch seine Vielseitigkeit ist es auch für viele andere Bereiche ein wertvolles Entwicklungswerkzeug.

Der neue Regelalgorithmus kann in Echtzeit an der realen Strecke getestet werden, ohne dass man sich Gedanken bezüglich Codeoptimierung machen muss. Zur Anbindung an die Strecke bietet das DS1104 R&D Controller Board neben 8 A/D-Kanälen, 8 D/A-Kanälen und 20 digitalen I/O-Kanälen auch Schnittstellen zu Drehgebern, serielle Schnittstellen sowie komfortable Generierung von dreiphasigen, mittensym-

Bild 1: Das DS1104 R&D Controller Board macht den PC zu einem leistungsfähigen Echtzeit-Entwicklungssystem.



DSP-basiertes Subsystem auf Basis eines TMS320F240 von Texas Instruments. Dieser DSP wurde speziell für die Regelung elektrischer Antriebe entwickelt und ist auf dem DS1104 R&D Controller Board als eigenständiges, frei programmierbares Subsystem integriert. Vielseitige Anwendungen

Bild 2 zeigt eine typische Entwicklungsanordnung mit dSPACE Prototypen. Die Echtzeithardware, in diesem Fall das DS1104 R&D Controller Board, wird in einem Standard-PC mit PCI-Interface installiert. Das DS1104 R&D Controller Board übernimmt die Echtzeit-Rechenleistung, die notwendigen I/O-Funktionen sowie die Signalgenerierung. Der Pulswechselrichter wird von dSPACE Prototypen, gesteuert.

metrischen PWM-Signalen.

Das Reglermodell läuft dabei auf dem Controller Board, das neben strengen Echtzeitanforderungen auch hohe Anforderungen bezüglich I/O erfüllen muss. Der Echtzeitcode wird automatisch aus dem Blockdiagramm generiert und auf dem Controller Board implementiert. Für die notwendige Echtzeit-Rechenleistung sorgt der Hauptprozessor der Karte, ein MPC8240 mit PowerPC-603e-Kern und einer Taktfrequenz von 250 MHz. Außerdem ist ein Daten- und Programmspeicher von 32 MByte integriert. Im geschlossenen Regelkreis können Parameter online variiert, Zeitschriebe in Echtzeit aufgenommen und Automatisierungsskripte abgefahren werden – und das in einer Phase, in der die Kosten für Korrekturen noch minimal sind.

### Komfortable Programmierung

Der Hauptvorteil gegenüber anderen Lösungen ist die komfortable Programmierung und Bedienung. Durch die Anbindung an MATLAB®/Simulink®/Stateflow® von The MathWorks können neue Regelalgorithmen grafisch im Blockdiagramm programmiert werden und laufen dann in Echtzeit auf der dSPACE-Prototypen-Hardware. Im Gegensatz zu herkömmlichen Controllern dauert die Umsetzung eines Simulink-Modells in echtzeitfähigen Code nur

wenige Sekunden. In Kombination mit der dSPACE-Software für Design, Analyse und Optimierung von Regelalgorithmen bietet das DS1104 R&D Controller Board somit eine hervorragende Entwicklungsumgebung, mit der der Aufwand für die Entwicklung neuer Regelalgorithmen drastisch reduziert werden kann.

Für den Einsatz in der Antriebstechnik verfügt das Board über umfassende Funktionen zur vereinfachten Entwicklung komplexer Antriebsregler, zum Beispiel eine modifizierbare, erweiterbare Raumzeigermodulation. Für die Generierung der dreiphasigen PWM sorgt ein

Auf dem PC selbst läuft lediglich das Modellierungswerkzeug MATLAB/Simulink/Stateflow für das Reglerdesign sowie dSPACE-Software zur Implementierung des Modells auf der Echtzeithardware und zur Visualisierung, Verwaltung und Automatisierung des Versuchs.

In diesem Beispiel messen zwei auf dem Kompensationsprinzip basierende Stromsensoren die für die Regelung benötigten Ströme. Zur Steuerung der eingesetzten Asynchronmaschine benötigt dSPACE Prototypen also „nur“ zwei 16-bit A/D-Wandler, drei PWM-Ausgänge sowie eine Drehgeber-Schnittstelle zur Ermittlung

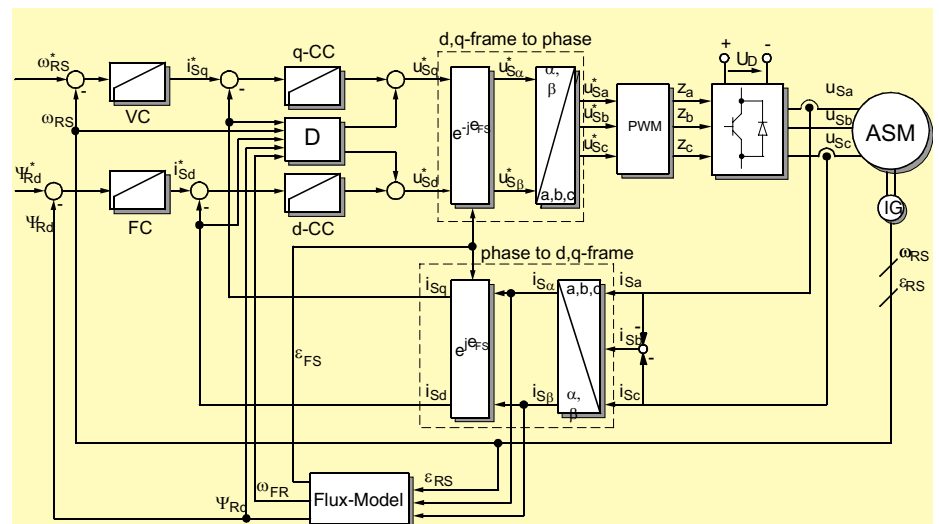


Bild 2: Beispiel für eine feldorientierte Regelung einer Asynchronmaschine.

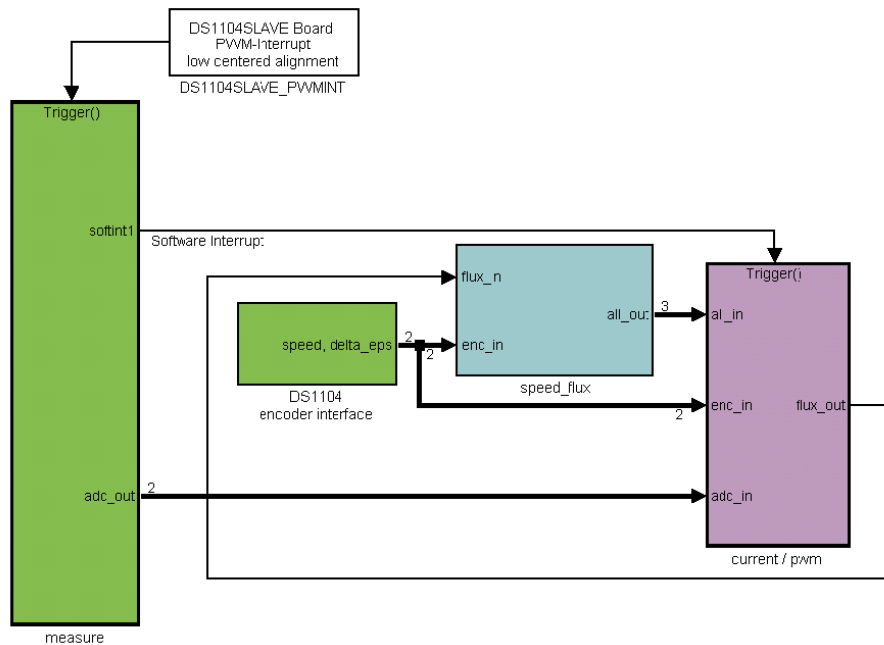


Bild 3: Generierung von dSPACE-Echtzeitcode über Real-Time Workshop.

der Motorposition über den Inkrementalgeber.

### Umsetzung der Regelung / Implementierung

Ein typisches Beispiel ist die Entwicklung einer feldorientierten Regelung. Bild 3 zeigt einen möglichen Regelalgorithmus. Eine solche Regelung kann sehr schnell in Simulink umgesetzt und auf der Echtzeithardware implementiert werden. Dies ist möglich durch ein enges Zusammenspiel von Real-Time Workshop (RTW) von The MathWorks und der dSPACE-Software „Real-Time Interface“ (RTI), die den Regelalgorithmus gemeinsam in Echtzeitcode umsetzen.

RTI übernimmt alle Implementierungsaufgaben: die Anbindung des Simulink-Modells über Blockbibliotheken, die Konfiguration des Echtzeitschedulers, das Compilieren und Linken über Real-Time Workshop sowie das Downloaden auf die Echtzeithardware. Im Gegensatz zu herkömmlichen Controllern, die meist manuell programmiert werden müssen, dauert die eigentliche Implementierung daher nur wenige Sekunden.

Zur Anbindung der I/O und der Einbettung der I/O-Funktionen in den Echtzeitcode bietet Real-Time Interface umfassende Simulink-Blockbibliotheken. Die Blöcke müssen lediglich per Drag&Drop

mit dem Simulink-Modell verbunden werden. Mit diesen Bibliotheken ist es daher möglich, selbst komplexe I/O-Kanäle wie Drehgeber im MATLAB/Simulink-Blockschaltbild anzuschließen und zu konfigurieren.

### Anbindung an die Regelstrecke

Für das Beispiel der Asynchronmaschine wird die Regelung in drei Teilen implementiert, dem Ein-/Ausgabeteil und zwei Regelkreisen: einer unterlagerten Stromregelung und einer überlagerten Drehzahlregelung.

Die für die Regelung benötigten Größen werden in einer separaten Task gemessen. Zur Messung der Strangströme werden mittels A/D-Kanälen des Boards Spannungen gemessen, die den erforderlichen Strangströmen proportional sind. Um sicherzustellen, dass die Ströme nicht an den Flanken der PWM gemessen werden, wird der Messvorgang direkt vom PWM-Interrupt des Slave-DSPs getriggert, und zwar durch einen mittensymmetrischen Interrupt. Der Start der A/D-Wandler auf dem DS1104 ist hardwaremäßig an diesen Trigger angeschlossen, wodurch eine jitterfreie A/D-Wandlung ermöglicht wird.

Zur Ermittlung von Geschwindigkeit und Position des Motors wird ausserdem auch eine Drehwinkelgeber-Schnittstelle innerhalb des Mess-Subsystems benötigt.

### Dreiphasige PWM-Generierung

Der Slave-DSP bietet die Möglichkeit der Generierung von dreiphasigen PWM-Signalen, entweder durch Vorgabe der drei Tastverhältnisse oder durch Vorgabe des Raumzeigers, wie in Bild 4.

Der Raumzeiger gibt den Sektor und die Werte der entsprechenden rechten ( $T_1$ ) und linken ( $T_2$ ) Vektoren vor.  $T_1/T_p$  kennzeichnet das Tastverhältnis des rechten Vektors im entsprechenden Sektor, während  $T_2/T_p$  das Tastverhältnis des linken Vektors vorgibt.

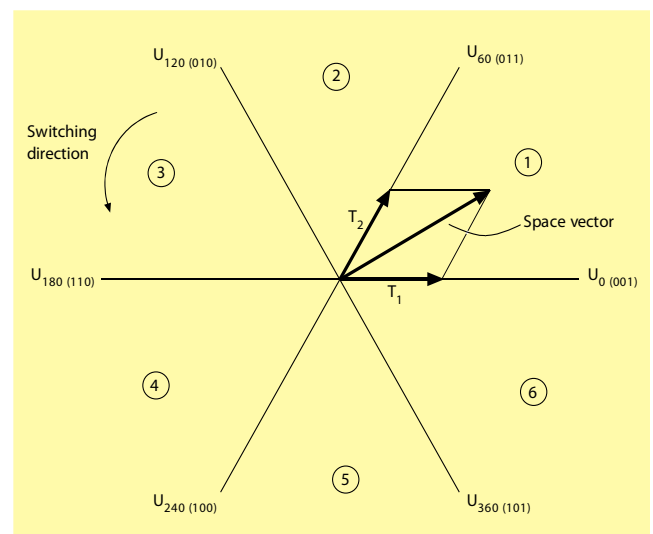


Bild 4: Raumzeigerbasierte PWM-Signale werden vor allem für die Regelung elektrischer Antriebe benötigt.



Für einen solchen Raumzeiger in Sektor 1 sieht das Pulsmuster der PWM-Signale, die vom Slave-DSP des DS1104 R&D Controller Board erzeugt werden, dann so aus. Bild 5: Pulsmuster der generierten PWM-Signale. Die Signale SPWM1, SPWM3, SPWM5 werden vom Board erzeugt und steuern den Pulswechselrichter. Die Programmierung dieser Signale ist mit dSPACE Prototyper denkbar einfach: Die Software RTI erlaubt die Programmierung solcher Funktionen in komfortablen Dialogen.

## Experimentierumgebung ControlDesk

Die Ergebnisse aller Echtzeit-Experimente stehen sofort zur Verfügung und werden auf dem PC online ausgewertet. Die Software „ControlDesk“ ist dabei die zentrale Experimentier-Umgebung innerhalb dSPACE Prototyper.

ControlDesk ermöglicht die Verwaltung, Instrumentierung und Automatisierung aller Experimente – von der interaktiven Steuerung von Simulink- und Echtzeit-Anwendungen bis hin zur komplexen Testautomatisierung.

Dafür stehen zum Beispiel virtuelle Instrumentenpanels zur Verfügung, die via Drag & Drop zusammengestellt und mit den entsprechenden Variablen verbunden werden. Annähernd alle ControlDesk-Funktionen können auch automatisiert werden.

Bild 6 zeigt ein ControlDesk-Layout für die implementierte Motor-Regelung. In

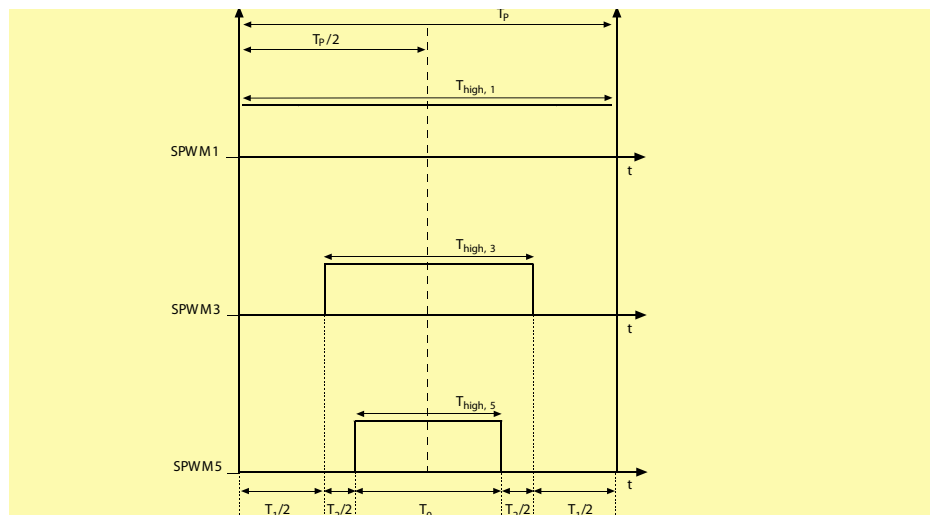


Bild 5: Pulsmuster der generierten PWM-Signale. Die Signale SPWM1, SPWM3, SPWM5 werden vom Board erzeugt und steuern den Pulswechselrichter

verschiedenen Layouts werden alle wichtigen Motordaten angezeigt, ausserdem bietet ControlDesk Eingabemöglichkeiten zur Einstellung der Sollwerte. In diesem Beispiel zeigt ein Zeitschrieb die gemessenen Werte von Geschwindigkeit, Strömen und der Trajektorie des Spannungsraumzeigers.

Mit dem neu integrierten Simulink Interface können Reglermodelle auch schon bei der Reglerentwicklung von

ControlDesk aus verwaltet, überwacht und automatisiert werden. Also in einer Phase, in der Echtzeitfähigkeit noch gar nicht gefordert ist. Damit ist mit ControlDesk der Schritt von Simulink zur Echtzeit-Welt denkbar einfach: man benutzt immer dieselben virtuellen Instrumente, Parametersets und automatisierten Testschleifen.

Autor: Thomas Pöhlmann  
dSPACE GmbH, Paderborn

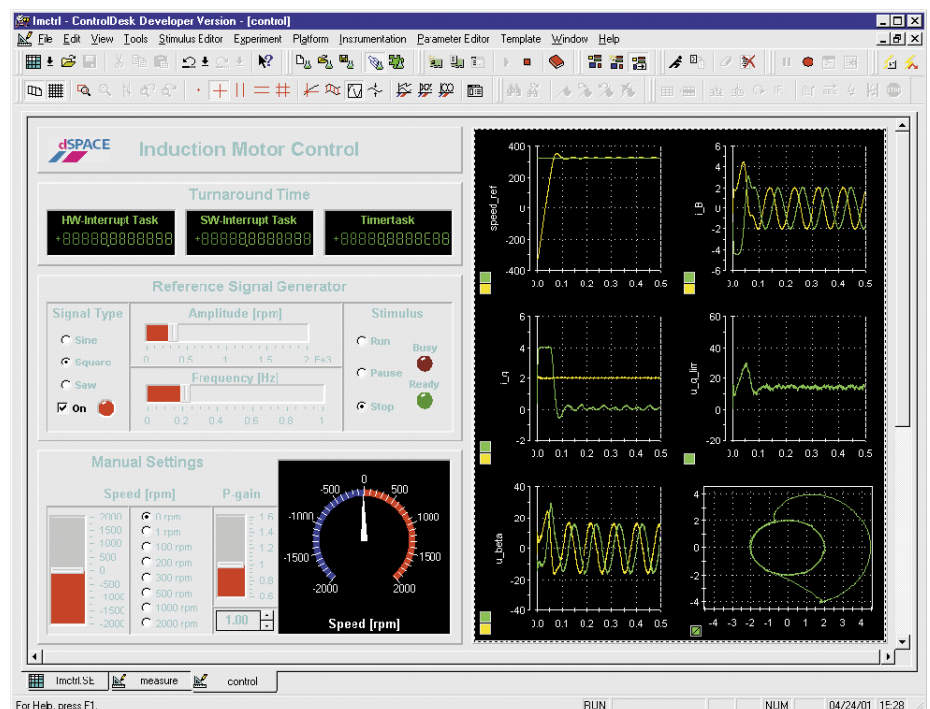


Bild 6: ControlDesk ist die zentrale Experimentierumgebung in dSPACE Prototyper.