

Skripte machen Umrichter effizienter



Bild: Gis/Shutterstock, alle anderen: International Energy Agency

Fast die Hälfte der elektrischen Energie fließt in elektrische Antriebe. Um den Ausstoß von klimaschädlichem Kohlendioxid einzuschränken, besteht demzufolge ein wesentlicher Hebel darin, solche Antriebe durch Umrichter effizienter zu machen. Das verlangt aber nach einfachen Umrichterlösungen.

Ingo Skuras
Product Marketing Manager für iMotion bei Infineon Technologies

Mit einer kontinuierlich wachsenden Bevölkerung steigt zwangsläufig auch der weltweite Bedarf an elektrischer Energie. Zwei Trends sind hierbei feststellbar: Zum einen legen die erneuerbaren Energien im Bereich der Energieerzeugung zu, zum anderen wächst die Forderung nach dem effizienten Einsatz von Energie.

Dies spiegelt sich sowohl in strengeren Regulierungen als auch im wachsenden Umweltbewusstsein der Konsumenten wider.

Ein Ansatz für Energieeinsparungen sind Elektromotoren. Diese und die davon angetriebenen Systeme stehen beim weltweiten Strombedarf an erster Stelle. Stu-

dien gehen davon aus, dass beim Stromverbrauch zwischen 43 % und 46 % auf Elektromotoren entfallen (Bild 1, [1]). Erst auf dem zweiten Platz folgt mit deutlichem Abstand die Beleuchtung.

Es ist also wenig verwunderlich, wenn der Trend im Bereich der Haushaltsgeräte ebenso wie im industriellen Segment eindeutig hin zu geregelten Antrieben (Variable Speed Drive, VSD) geht. Denn sowohl für Privathaushalte als auch für Industrieunternehmen ist der Stromverbrauch mit tendenziell steigenden Kosten verbunden. Obwohl immer mehr Motoren

elektronisch geregelt sind, stellen diese für die meisten Gerätehersteller kein Alleinstellungsmerkmal dar. Vielmehr wird die Motorregelung als funktionaler Baustein gesehen, insbesondere dann, wenn es sich um Hilfsfunktionen wie Abwasserpumpen in Spülmaschinen oder Lüfter in Mikrowellen handelt. Punkten können Anbieter von Leistungshalbleitern deshalb vor allem dann, wenn sich mit integrierten Lösungen der Entwicklungsaufwand reduzieren lässt und der Einsatz eines Umrichters kostengünstiger wird. Für solche Anwendungen bietet Infineon mit iMotion eine integrierte Lösung (Bild 2). Diese Produkte kombinieren die erforderliche Hardware mit einem einsatzbereiten Algorithmus, um Motor und PFC (Power Factor Correction) anzu-

steuern. Die Anforderungen der Gerätehersteller im privaten wie auch im industriellen Bereich sind stark von der jeweiligen Applikation abhängig. Für den breiten Einsatz in einer Vielzahl von Anwendungen müssen integrierte Produkte daher eine Flexibilität anbieten, die über die reine Konfiguration von Motor und Leistungsstufe hinausgeht.

Sowohl die vorherige iMotion-Generation wie die IRMCF100-Serie als auch

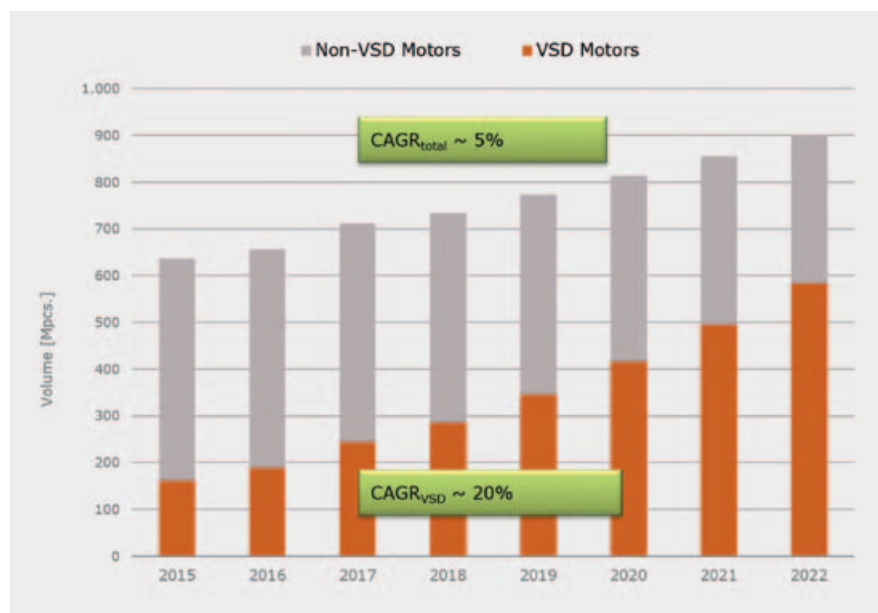


Bild 1: Die Wachstumsraten geregelter Antriebe liegen deutlich über dem allgemeinen Marktwachstum.

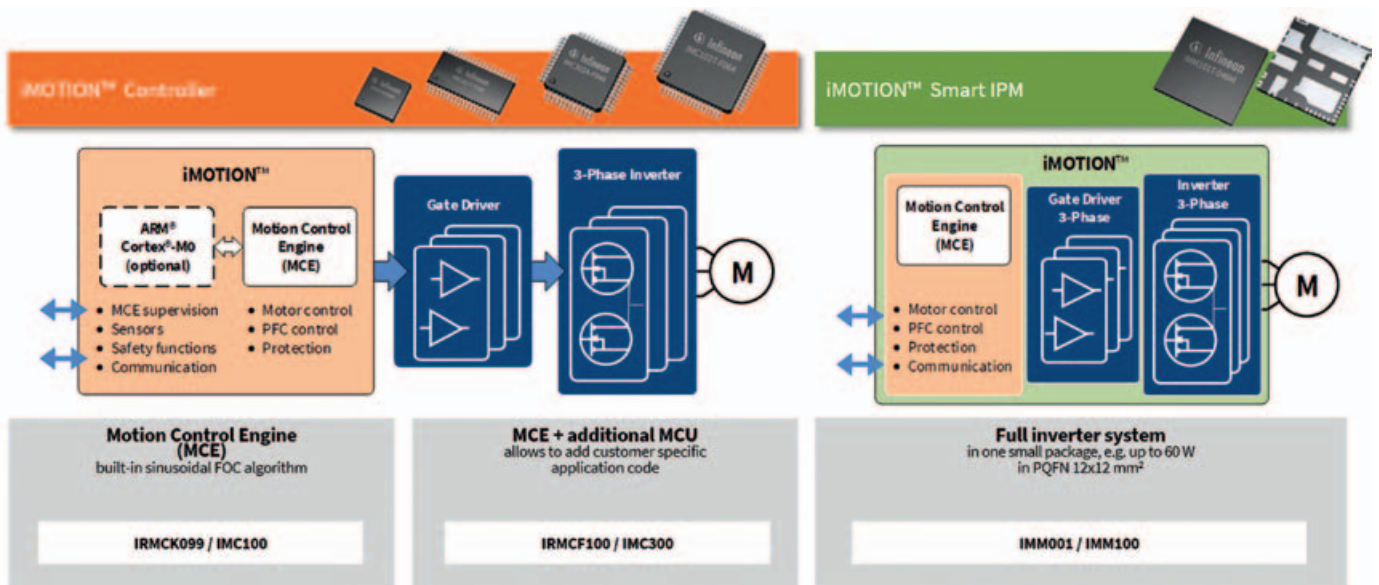


Bild 2: Die Kombination von Hard- und Software eröffnet den Geräteentwicklern neue Freiheitsgrade.

die kommende IMC300-Familie sollen dieser Forderung durch Kombinieren des Motor-Controllers mit einem zusätzlichen Mikrocontroller gerecht werden. Diese Dual-Core-Architektur bietet dem Geräteentwickler die volle Flexibilität, erhöht aber die Hardwarekosten und verlangt von ihm, den vollen Software-Entwicklungszyklus bis zur Freigabe oder Zertifizierung zu durchlaufen. Überwiegend kommen Anwendungen allerdings ohne diese volle Flexibilität aus. Hier wäre es wünschenswert, eine einfachere und kostengünstigere Möglichkeit zu haben, kundenspezifische Anpassungen vorzunehmen.

Erst vor kurzem hat Infineon die iMotion-Controller-Familie IMC100 eingeführt. Diese basiert auf einer neuen Hardware-Architektur, die moderne analoge und digitale Peripheriemodule mit einem 32-Bit-Rechenkern verbindet. Mit dem Fokus auf kostensensitiven Anwendungen kommt bei den IMC100 nur ein einzelner Kern zum Einsatz.

Flexibilität ohne Zusatzkosten

Die neueste Version der Motion-Control-Engine (MCE) von iMotion beinhaltet einen Skript-Interpreter, der bereits für einen einzelnen CPU-Kern recht flexible Möglichkeiten für kunden- und applikationsspezifische Anpassungen bietet – der zusätzliche MCU-Kern kann entfallen. Die Skript-Sprache unterstützt das Einlesen analoger Sensordaten ebenso wie das Ansteuern digitaler Ein- und Ausgänge. Auch lässt sich der Motoralgorithmus zur Laufzeit steuern, um zum Beispiel einen speziellen Motoranlauf zu implementieren. Der Skript-Interpreter läuft als Hintergrundprozess der MCE und ähnelt damit einer virtuellen Maschine. Dadurch ist sichergestellt, dass das Skript den eigentlichen Motor- und PFC-Algorithmus nicht beeinträchtigt.

Im Bereich eingebetteter Applikationen wie zum Beispiel Motorsteuerungen

unterliegen Algorithmen einem typischen Lebenszyklus. Die Erstimplementierung erfolgt in der Regel vollständig in Software, um das System weiter anpassen und optimieren zu können. Ist ein gewisser Reifegrad erreicht, erfolgt die Abbildung teilweise oder komplett in Hardware, um die Performance und Effizienz zu verbessern. Im Fall eines Motoralgorithmus übernehmen inzwischen »intelligente« A/D-Wandler und Timer große Teile der Regelschleife, denn das reduziert die Rechenlast für den CPU-Kern deutlich. Da heutzutage auch die günstigsten Controller auf modernen 32-Bit-Architekturen aufsetzen, ergibt sich daraus die Möglichkeit, ungenutzte Rechenkapazität dem Anwendungsentwickler zur Verfügung zu stellen.

Controller der iMotion-Serie implementieren einen geprüften und freigegebenen Algorithmus und sind für die Anforderungen der funktionalen Sicherheit für Haushaltsgeräte nach IEC/UL 60335

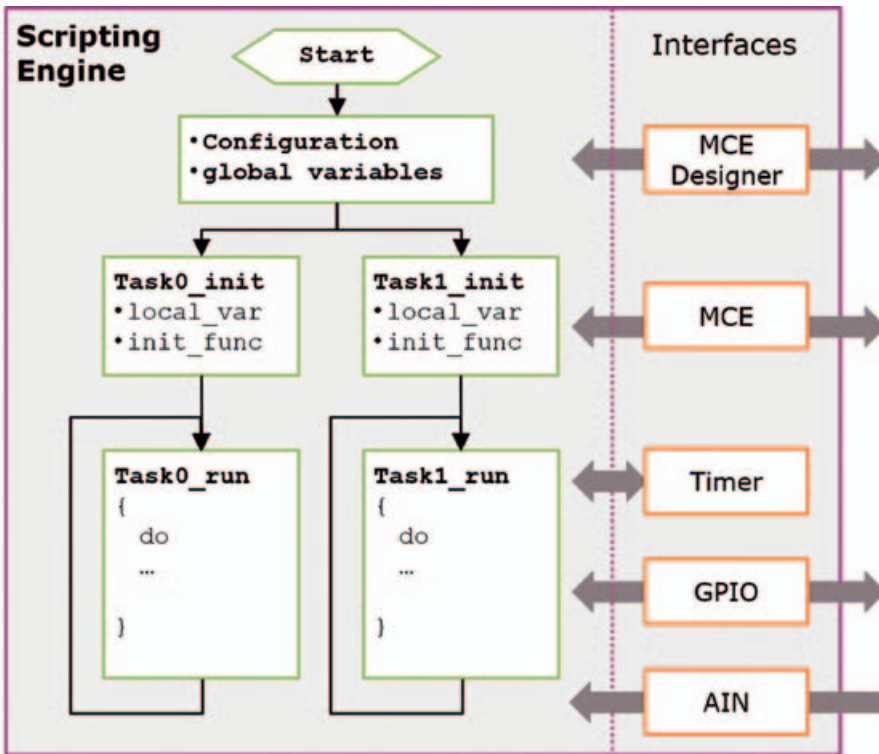


Bild 3: Die Implementierung und die Schnittstellen des Skript-Interpreters sind darauf ausgerichtet, dass Geräteentwickler den iMotion-Controller recht schnell und vielfältig einsetzen können.

zertifiziert. Dementsprechend hat die sichere Funktion des Motors (und der PFC) Vorrang. Die neue Version der MCE führt die Kundenskripte in der abgeschlossenen Umgebung des Skript-Interpreters aus, der an das Konzept einer virtuellen Maschine angelehnt ist. Diese Implementierung bietet einige Vorzüge, hierzu gehören vor allem:

- keine Beeinträchtigung des Motor- und PFC Algorithmus,

- klar definiertes Interface zwischen Skript und Kontroll-Algorithmus,
- einfach zu verstehende und komplette Skriptsprache und
- Schutz von Skripten vor dem Auslesen, um das geistige Eigentum des Kunden zu wahren.

Der Skript-Interpreter führt zwei Tasks parallel aus (Bild 3): Task 0 läuft mit einer Wiederholrate von 1 ms, Task 1 mit einer Wiederholrate von 10 ms. Beide Tasks wer-

den beim Systemstart über ihre Initialisierungsfunktionen konfiguriert und laufen anschließend in einer Endlosschleife. Über den Skript-Interpreter kann der Geräteentwickler Ausführungszeit und Wiederholrate der Tasks konfigurieren. Zusätzlich unterstützt ein freilaufender 32-Bit-Timer die Ausführung von Aktionen zu genau definierten Zeitpunkten. Um das Skript zu debuggen und zu visualisieren kann der Geräteentwickler über die Benutzeroberfläche zum MCEDesigner am PC auf die globalen Skriptvariablen zugreifen.

Skripte für iMotion verwenden eine an die Programmiersprache C angelehnte Syntax. Sie unterstützen arithmetische und logische Operatoren ebenso wie Entscheidungen (if...else) und Schleifen (for...). Variablen lassen sich global oder lokal in einem der Tasks definieren und verwenden. Der Zugriff auf die Motor- und PFC-Regelung erfolgt über die definierten Register der MCE [3].

Um den Skriptcode ausführen zu können, wird dieser in einen kompakten Bytecode übersetzt und im Zuge dessen auf korrekte Syntax geprüft. Nach dem Ablegen des Bytecodes im Flash-Speicher des iMotion-Produkts, kann der Skript-Interpreter diesen ausführen. Um das jeweilige geistige Eigentum zu schützen, lassen sich die gespeicherten Skripte gegen Auslesen schützen. Eine integrierte Prüfsumme stellt dabei die Identifikation und Integrität des jeweiligen Skriptes sicher.

Um den Programmieraufwand zu reduzieren, ist das Schreiben und Verwalten von Skripten in die gleichen Werkzeuge

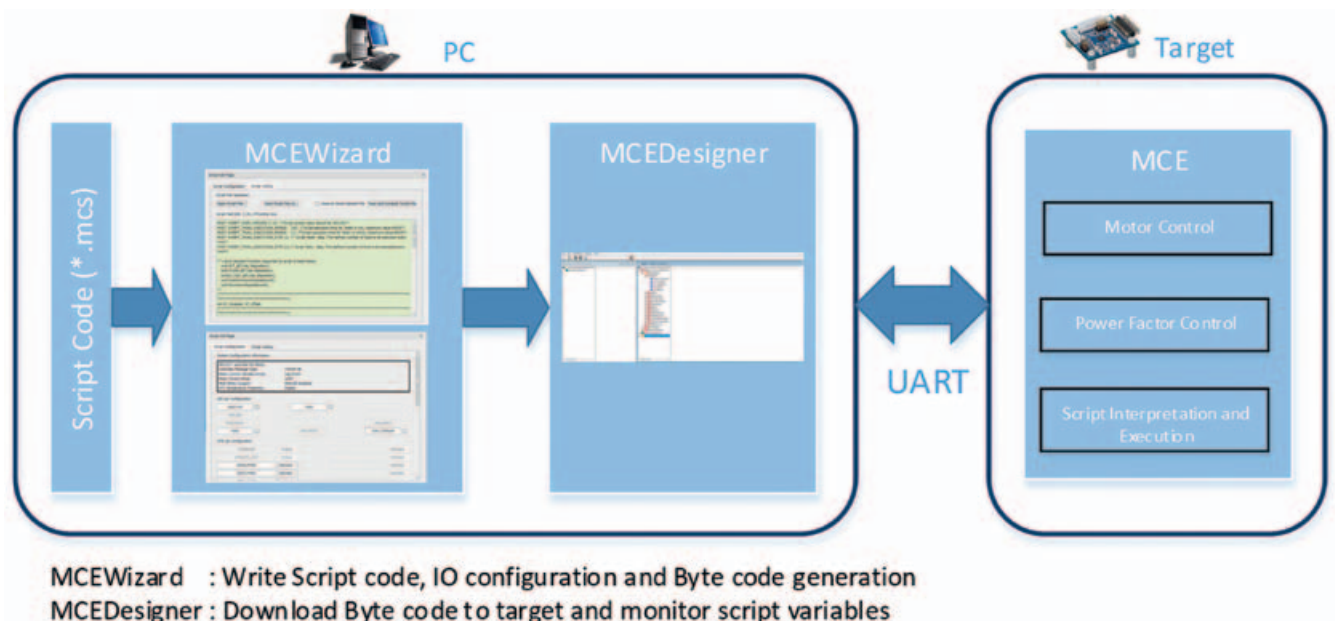


Bild 4: Integrierte Tools für das Schreiben und Verwalten der Skripte.

integriert, die auch zur Konfiguration des iMotion-Controllers für den jeweiligen Antrieb verwendet werden (*Bild 4*). Mithilfe des MCEWizard kann der Geräteentwickler die analogen und digitalen IOs für die Verwendung im Skript konfigurieren. Die Skripte selbst schreibt er im integrierten Editor, wo sie dann das Toll auch in Bytecode transferiert.

Der MCEDesigner behandelt die Skrip-te analog zu den Konfigurationsparametern:

- Unterstützung für Laden und Löschen im Flash sowie
- Lesen, Schreiben und Visualisieren der globalen Skriptvariablen.

In der Serienfertigung werden Skripte und Parameter in einem Zug zusammen mit dem Image der MCE programmiert.

■ *Beispiel Haartrockner und Fazit*

Die Details des Skript-Interpreters und der entsprechenden Sprache sind in [3] beschrieben. Darüber hinaus ist eine Application Note [4] verfügbar, die den konkreten Anwendungsfall eines Haartrockners mit bürstenlosem Gleichstrommotor beschreibt und den kompletten Skriptcode enthält. Der eingesetzte Controller vom Typ IMC100 steuert den Motor und implementiert folgende Zusatzfunktionen per Skript:

- Auswahl der Geschwindigkeit,
- drehzahlabhängiges Begrenzen des Motorstroms,
- Tiefpassfilter für die Zwischenkreisspannung und
- dynamische Anpassung der Drehzahl an die Zwischenkreisspannung.

Weil der IMC100 eine fertige Motorsteuerung mit der Flexibilität des Skript-Interpreters kombiniert, profitieren Hersteller von niedrigsten Stückkosten bei gleichzeitig höchster Energieeffizienz.

Durch den neu eingeführten Skript-Interpreter in der neuen Version der Motion-Control-Engine (MCE) bei iMotion lassen sich die vorhandenen Hardware-Ressourcen bestmöglich nutzen.

Er bietet das notwendige Maß an Flexibilität, welches für den Großteil der adressierten Anwendungen passend ist. Die gewählte Implementierung in Anlehnung an eine virtuelle Maschine sorgt für eine klare Trennung von Motor- und PFC-Regelung vom Kundenskript. Diese Trennung stellt sicher, dass die eigentliche Regelung nicht beeinträchtigt wird und damit ein funktional sicherer Betrieb gewährleistet ist. Mit dem Einsatz der betriebsbereiten Regelung und der Adaption unterschiedlicher Funktionalitäten durch Skripte kann sich der Geräteentwickler auf höherwertige Funktionen konzentrieren und die Schnittstelle zum Gerät optimieren. (rh)

REFERENZEN

- [1] Paul Waide and Conrad U. Brunner, Energy-Efficiency Policy Opportunities for Electric Motor-Driven Systems, International Energy Agency (IEA), 2011, <https://tinyurl.com/ospa7n9>
- [2] Dinesh Kithany, Home Appliance Database: All Devices and Associated Electronics, IHS Markit, Mai 2018, <https://tinyurl.com/yc7a8os2>
- [3] iMOTION Motion Control Engine, Software Reference Manual V 1.1, Infineon Technologies, August 2018, <https://tinyurl.com/y74rssvu>
- [4] How to Use iMOTION Script Language (AN2018-27), September 2018, <https://tinyurl.com/yc6kgdny>