



**Rada Rodriguez**  
Spezialisten für  
Energiemanagement



**Zeitsparend**  
Standardisierte  
Systemverkabelung

**Marktübersicht**  
Dezentrale  
IP20-IO-Systeme

**VDE**  
VERLAG

Sonderausgabe S5/2011

# etz

Elektrotechnik Information

**SPS/IPC/Drives**  
**Sonderausgabe**  
**Highlights**



**TwinCAT® 3**

**BECKHOFF**



# Steuerungsprogrammierung nutzt Vorteile der SPS- und IT-Welt

**Softwareprogrammierung geht heute über Kontaktplan hinaus und der Softwareentwicklungsprozess muss dem Rechnung tragen. Standard-IT-Umgebung, Engineering mit Modulen, Runtime auf Multicore-Prozessoren – mit TwinCAT 3 werden Steuerungsprogrammierer neue Wege beschreiten.**

**Ronald Heinze**

kleinen ARM-Chips bis hin zu modernen X86-Prozessoren mit Multicore genutzt werden.

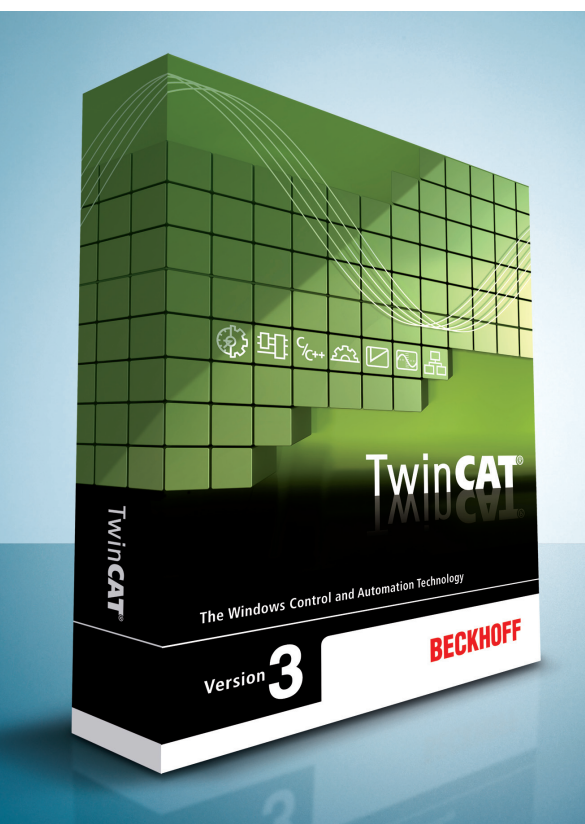
Twincat 3 unterstützt Echtzeit-C++ auf Basis des marktführenden Compilers von Visual Studio. „Die Integration der Twincat-3-Entwicklungswerkzeuge in eine Standardentwicklungsumgebung ermöglicht es Anwendern, auf alle Errungenschaften der IT-Welt zuzugreifen“, so R. Barth. „Programme können modularer geschrieben werden. Hinzu kommen die Wiederverwendbarkeit von Code sowie die objektorientierten Mechanismen. Darüber hinaus laufen die Matlab/Simulink-Algorithmen direkt in der Echtzeit-Umgebung ab.“ (Bild 3)

## Für SPS- und Hochsprachen-Programmierer geeignet

Twincat 3 ist, laut R. Barth, für zwei Anwendergruppen ideal ausgelegt: Zum einen sind dies die SPS-Programmierer, die sich in der Umgebung entsprechend der IEC-61131-3 [2] zu Hause fühlen. Zum anderen zählen

61131-3-Software-SPS, Software-NC und Software-CNC in Windows-Betriebssysteme.

„Die neue Version unserer Automatisierungsssoftware, Twincat 3, ist die Softwareplattform für die Automatisierungstechnik des nächsten Jahrzehnts“, betont Ramon Barth, Leiter der Softwareentwicklung System, HMI und Echtzeit bei Beckhoff (Bild 1 und 2). „Mit ihr ist es nun möglich, moderne PC-Architekturen optimal zu nutzen. Bei der Entwicklung stand, nicht der Portalgedanke, sondern ein komplett offenes System mit einer vollständigen Produktpalette, auf der Agenda“, so der Softwarespezialist. Die neue Version bietet moderne SPS-Programmierung mit objektorientierten Erweiterungen, C++ und Matlab/Simulink-Integration und das Ganze in einer modularen Runtime, die von

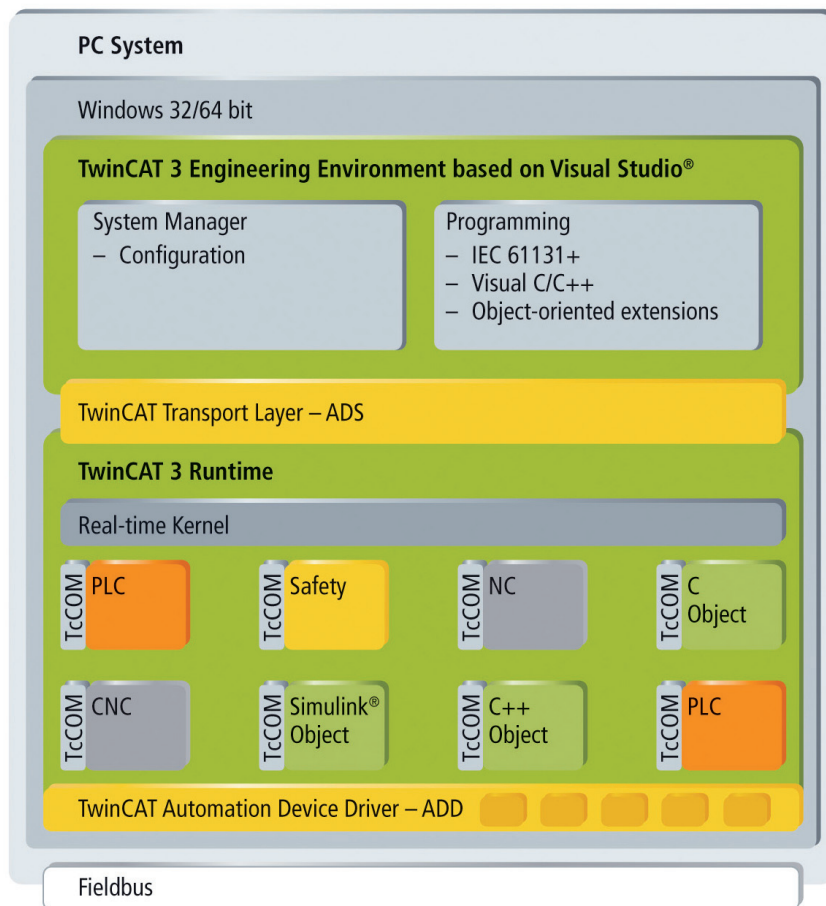


**Bild 1.** Twincat 3 wird zur SPS/IPC/Drives 2011 releast

In der Automatisierungstechnik gewinnen integrierte Entwicklungsumgebungen immer mehr an Bedeutung. Zunehmend werden sie zum ausschlaggebenden Kriterium, aufgrund dessen Anwender ihre Systementscheidungen treffen. Die Erhöhung der Effizienz des Engineerings spielt dabei eine wichtige Rolle. Ein Softwaresystem, wie Twincat von Beckhoff Automation [1], verwandelt nahezu jeden kompatiblen PC in eine Echtzeitsteuerung mit Multi-SPS-System, NC-Achsregelung, Programmierungsumgebung und Bedienstation. Es ermöglicht die Einbettung von IEC-



**Bild 2.** Dr. Dirk Janssen (links) und Ramon Barth, beide Leiter Softwareentwicklung bei Beckhoff Automation, im Gespräch mit der etz



**Bild 3.** Die Engineering-Plattform Twincat 3 basiert auf Visual Studio

Anwender dazu, die mit C++ ihre Aufgaben lösen wollen. „Beides ist in TwinCAT 3 vorgesehen: objektorientierte Programmierung mit den Mechanismen der Vererbung sowie die Kontaktplanprogrammierung“, stellt R. Barth heraus.

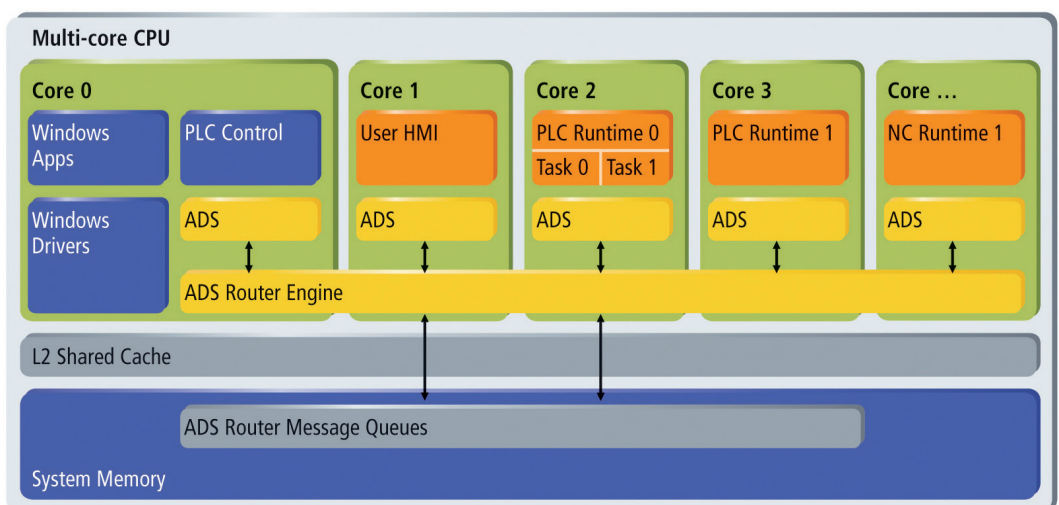
Die Einbindung in Visual Studio ermöglicht es, einfach auf weitere Tools zuzugreifen. „Visual Studio stellt alle modernen Hilfsmittel für die Programmierung zur Verfügung“, erläutert Dr. Dirk Janssen, Leiter der Softwareentwicklung System, CNC und IO. Dazu zählt R. Barth zum Beispiel die UML-Integration und die Versionskontrolle. „Automatisierungsprojekte lassen sich nun

so planen, wie es im IT-Bereich üblich ist, nämlich entsprechend dem Softwareentwicklungsprozess“, setzt er fort. Dies sei Stand der Technik und würde derzeit an den Hochschulen gelehrt. „Visual Studio ist ein Standard, nach dem sich alle richten.“ Alternativen gebe es nur in der Open-Source-Welt, die aber nicht die Viel-

seitigkeit, die Verbreitung und die Performance mitbringt. Dr. D. Janssen ergänzt: „Wir stellen unseren Anwendern die gleichen Tools zur Verfügung, die wir selber für die Programmierung nutzen.“ So sei es nun auch möglich, dass Anwender das komplette Automatisierungsprojekt in Visual Studio erstellen und es anschließend in TwinCAT 3 integrieren.

## Wiederverwendbarkeit von Code

TwinCAT 3 bietet dem Softwareentwickler eine Infrastruktur, um wiederverwendbare Softwaremodule zu generieren. „Dies lebt die neue TwinCAT-Version durch die eigene Modularisierung vor“, bemerkt R. Barth. Das Prinzip beruht auf dem Austausch von Interfaces, deren Beschreibung und der Unterstützung von Techniken zur objektorientierten Programmierung (OOP). Diese Interfaces stellen strikte Verträge zwischen den Modulen dar und gewährleisten die Kompatibilität zu zukünftigen Entwicklungen. „TwinCAT liefert sozusagen den Rahmen und Teile der Implementation, in den der Anwender seine Module einbetten kann“, teilt R. Barth mit. Er gibt zu bedenken, dass heutige Projekte oft von jungen Ingenieuren realisiert werden. „Diese wollen neue Ideen einbringen und fordern moderne Tools ein. Die jüngeren Programmierer sind mit den neuen Technologien vertraut und wollen diese auch nutzen. Zudem kann man auch langsam in diese neue Welt hineinwachsen, indem man beispielsweise zuerst



**Bild 4.** Die Multicore-Unterstützung durch TwinCAT 3 ermöglicht es, die Leistungsfähigkeit der Cores vollständig zu nutzen



nur einfache Eigenschaften, wie die Vererbung nutzt.“ Trotzdem sind, laut Dr. D. Janssen, auch die bisherigen SPS-Programmierer genügend berücksichtigt worden, da die übliche IEC-61131-3-Programmierung in Twincat 3 weiter vorgesehen ist. „Die objektorientierten Erweiterungen in der SPS von Twincat 3 sind nahtlos in die bestehende IEC-61131-3-Programmiersprache integriert“, unterstreicht Dr. D. Janssen. „Kein Anwender wird gezwungen, neue Programmierpara-

führungen eingesetzt wird, soll möglichst einheitlich von der Steuerung angesprochen werden. Daher definiert der Anwender die gemeinsamen Eigenschaften in einem Basisfunktionsbaustein und leitet davon die verschiedenen Varianten ab, sodass nur noch die Unterschiede neu entwickelt werden müssen.“

Dr. D. Janssen betont, dass Twincat 3 auch die Inbetriebnahme weiter vereinfacht, da die Module fertig ausgetestet werden können. „Inbetriebneh-



**Bild 5.** Dr. D. Janssen: „Die Multicore-Unterstützung durch Twincat 3 ermöglicht es, die Leistungsfähigkeit der Cores vollständig zu nutzen.“

digmen zu lernen und einzusetzen.“ Darüber hinaus ist die Software skalierbar, sodass auch „abgespeckte“ Versionen zur Verfügung stehen.

Dank der objektorientierten Programmierung lässt sich der Code leicht wiederverwenden. „Mit der Wiederverwendbarkeit von Code lassen sich viele Programmteile automatisch erstellen“, äußert Dr. D. Janssen. Dies war zwar auch schon in der Vorgängerversion möglich, jedoch unter Nutzung einer darüber liegenden Compiler-Schicht, was sich ungünstig bei nachträglichen Änderungen auswirkte. R. Barth gibt ein Beispiel für den praktischen Nutzen der Wiederverwendbarkeit von Code: „Ein Bohr-aggreat, das in verschiedenen Aus-

mer arbeiten heute oft nur mit der Anpassungsschicht“, setzt er fort. „Vor Ort sind dann nur noch kleinere Veränderungen erforderlich, was sich einfach realisieren lässt. Die objektorientierten Erweiterungen bieten komfortable Möglichkeiten, um die vermehrte Modularisierung von Maschinen und deren Aggregaten auch im Steuerungsprogramm entsprechend abzubilden.“

### **Unterstützung von Multicore**

„Die Multicore-Unterstützung durch Twincat 3 ermöglicht es, die Leistungsfähigkeit der Cores vollständig zu nutzen“, schließt Dr. D. Janssen an (Bild 4 und 5). Dies habe signifikante Auswirkungen, da sich nun das Ma-

## Fragen aus der Praxis

Ramon Barth und Dr. Dirk Janssen, verantwortlich für die Softwareentwicklung bei Beckhoff, beantworten Fragen aus der Praxis (Bild 6).

**etz:** Inwieweit ist Twincat 3 auch für andere Steuerungshersteller offen?

**R. Barth:** Wir sehen durchaus das Interesse kleinerer Steuerungshersteller, ihren Code in Twincat ablaufen zu lassen. Damit können sie die vielfältigen Möglichkeiten nutzen, die unsere Softwareumgebung bietet: Darunter fallen auch die enge Integration in die PC-Welt und die Nutzung der Ethercat-Kommunikation. Technisch ist das überhaupt kein Problem. Dies trifft nicht nur auf Steuerungshersteller zu. Auch Maschinenbauer haben ihr Know-how oft in die Programmiersprache C „gegossen“ und sehen nun die Möglichkeit, dieses in einer leistungsfähigen und offenen Umgebung zu nutzen.

**etz:** Wie kann der Quellcode geschützt werden?

**R. Barth:** Unser Twincat-Lizenzierungsmodell sieht vor, dass das Zielsystem identifiziert werden muss. Als Zusatzoption lässt sich der Anwender-Code verschlüsseln, sodass dieser nur auf einem Zielsystem bzw. nur auf einem geschützten System läuft. Übrigens spricht die zunehmende Bedeutung eines geschützten Quellcodes nicht gerade für Open-Source-Modelle.

**etz:** Was sind die Pläne für die strategische Weiterentwicklung von Twincat 3?

**R. Barth:** Die weitere Unterstützung zur Erstellung und Nutzung von bereits vordefinierten Softwaremodulen steht bei der Weiterentwicklung im Vordergrund. Wir haben in Twincat 3 ein einheitliches Beschreibungsformat für Softwaremodule eingeführt, das weiter ausgebaut und um entsprechende Tools ergänzt wird.

**Dr. D. Janssen:** Ende 2012 steht eine erweiterte Version von Twincat 3 an.

Die Visualisierung, deren Erstellung wir in der Vergangenheit mit leistungsfähigen Datenschnittstellen unterstützt haben, wird hier eine wichtige Rolle spielen. Weiterhin bringen wir im nächsten Jahr ein Modul mit integrierter PC-Safety-PLC heraus. Darüber hinaus wird eine Authentifizierung bei der Kommunikation mit Steuerungen oder von Steuerungen untereinander integriert, das heißt, ein zusätzlicher Security-Level schützt noch besser vor Angriffen.

**R. Barth:** Das Laufzeitsystem wird 64-bit-fähig. Dies ist bisher nur die Entwicklungsumgebung, da im Runtime-Bereich kaum eine Nachfrage bestand.



**Bild 6.** Dr. Dirk Janssen und Ramon Barth: „Wir bringen im nächsten Jahr ein Modul mit integrierter PC-Safety-PLC heraus.“

**etz:** Inwieweit sind Sie mit dem Plan vorangekommen, Softwaremodule von Drittanbietern für Twincat mit anzubieten?

**Dr. D. Janssen:** Von der technischen Seite haben wir dieses Thema bereits abgeschlossen. Die notwendigen Schnittstellen und Beschreibungsformate liegen vor und entsprechen exakt denen, die wir intern verwenden. Auf der Roadmap steht u. a. noch die Frage der Kommerzialisierung. Es bleibt unsere feste Zielsetzung, Softwaremodule von Drittanbietern ins Produktportfolio zu nehmen.

**etz:** Ist bei Twincat eine Eplan-Unterstützung vorgesehen?

**Dr. D. Janssen:** Bereits mit Twincat 2 ist es möglich, über Supplements, aus der Eplan-Welt Variablen zu übernehmen. Selbstverständlich ist dies auch in der

neuen Version 3 möglich. Wir sehen dies als Stand der Technik. Mithilfe eines Importwerkzeugs können E-CAD-Systeme angebunden werden.

**etz:** Können alte Twincat-Projekte mit Twincat 3 gewartet werden?

**Dr. D. Janssen:** Es ist möglich, beide Versionen auf einem Rechner laufen zu lassen, sodass Servicetechniker die Tools auswählen können, die zur jeweiligen Runtime passen. Wir denken, dass Anwender ihre Zielsysteme mit der Version bearbeiten, in der die Programme auch erstellt wurden. Nur dies stellt sicher, dass nur das geändert wird, was

der Anwender auch ändern will. Allerdings können bestehenden Twincat-Projekte in Twincat 3 importiert werden, um sie dann auf einer Twincat-3-Runtime unverändert und erweitert laufen zu lassen.

**etz:** Wie sehen die Lizenzmodelle für Twincat 3 aus?

**Dr. D. Janssen:** Es werden weiterhin die Laufzeitsysteme lizenziert und nicht

die Programmierungsumgebungen. Für die Laufzeitsysteme haben wir ein gerechteres Lizenzmodell eingeführt, welches deutlich feingranularer ist. Basis-Run-time-Komponenten (Echtzeit, IO, PLC-Laufzeit,...) werden als Module betrachtet. Anwendermodule können sich in das Lizenzsystem integrieren und die Infrastruktur über Beckhoff mit nutzen. Die Lizenzierung erfolgt grundsätzlich pro Zielsystem.

**R. Barth:** Unsere Module sind übrigens auch nicht versionsgebunden.

**etz:** Ist kostenloses Testen noch vorgesehen?

**Dr. D. Janssen:** Selbstverständlich. Alle Twincat-Module unterstützen einen zeitlimitierten Testbetrieb. Es gibt keine funktionalen Einschränkungen.

schinenprogramm einfach auf verschiedene Cores aufteilen lässt. „Die Aufteilung wurde in unserem Haus bereits mit 16 Cores erfolgreich getestet“, berichtet *R. Barth* an. „Dabei ermöglicht es die Hyperthreading-Funktion, virtuelle Cores zu bilden, die nur unwesentlich langsamer als echte Cores sind. Auf einem Serversystem mit zwei physikalischen Cores lassen sich problemlos 24 virtuelle Cores platzieren.“

„Wir haben mit allen Prozessoren Benchmarks durchgeführt“, betont der Softwareentwickler weiter. „Die Ergebnisse haben gezeigt, dass die Twincat-Multicore-Unterstützung eine Performance-Steigerung linear zur Anzahl der CPU-Cores ermöglicht.“ [3] „Dies beruht darauf, dass das Steuerungsprogramm sofort aus dem Speicher verfügbar ist und keine Wartezeiten erforderlich sind“, begründet Dr. *D. Janssen*. „Bei einem Core fallen viele Unterbrechungen an“, ergänzt *R. Barth*. „Zuerst werden hochprioritäre Tasks bearbeitet. Die Unterbrechungen kosten Rechenleistung. Bei Multicore werden Leistungsverluste, die durch die Synchronisierung der Cores entstehen, durch den Wegfall von Task-Kontextwechseln nahezu ausgeglichen.“ Die beiden Spezialisten sind sich einig: „Je komplexer die Anwendung, desto größer ist der Nutzen.“

„Die Modularisierung und die Multicore-Technologie passen ideal zusammen“, betont Dr. *D. Janssen*. „Denn Multicore-Anwendungen ermöglichen es, dass die Module autark laufen und sich nicht gegenseitig beeinflussen.“ „Dies spart Rechenleistung“, ist *R. Barth* überzeugt. Hinzu komme, als weiterer Vorteil, dass die Programme bei der Modularisierung leichter getestet werden können.

## Komplexität reduzieren

„Die Beta-Phase von Twincat 3 wurde sehr intensiv genutzt, um die Software weiter zu verbessern“, stellt Dr. *D. Janssen* fest. „Dazu haben wir mit sehr unterschiedlichen Anwendern zusammen gearbeitet. *R. Barth* ergänzt hierzu: „Vertreter der klassischen Automatisierungstechnik haben viel Wert auf die objektorientierten Erweiterungen gelegt und ihre Modularisierungsvorstellungen entsprechend umgesetzt. Aber auch für die Matlab/Simulink-Unterstützung kam

eine Reihe von Anregungen, zum Beispiel für den Austausch von Matlab-Elementen.“ Gezeigt habe sich aber auch die einfache Handhabung und Installation. Zum Beispiel war ein Tester aus dem Bereich komplexer Sensoren sehr zufrieden: Dieser hatte einen eigenen Hardwaretreiber geschrieben. „Bei den komplexen Sensorvorverarbeitungen kommt die Leistungsfähigkeit eines modernen PC-Systems mit Multicore-Unterstützung zum Tragen, da dieses spezielle und teure externe Messsysteme ersetzt. Neu für uns waren Anwender, die komplett auf C/C++ als Automatisierungssprache setzen und keine SPS im Einsatz haben“, so *R. Barth* weiter. „Positives Feedback haben wir auch für die von uns aus der SPS-Welt übertragenen Debug-Möglichkeiten erhalten.“

*R. Barth* stellt heraus, dass Twincat 3 die Komplexität vereinfacht: „Es sind nur drei bis vier Maus-Clicks erforderlich, um die einzelnen Tasks auf verschiedene Cores zu verteilen.“ Anwender müssen nur ihren Code implementieren. Die Software sorgt für die komplette Infrastruktur und die Kommunikation über ADS von selbst. „Dies gilt ebenso für die Integration von Matlab/Simulink“, räumt Dr. *D. Janssen* ein. Diese laufe wie bei C++. „Um mit Matlab ein Twincat-Systemmodul zu erstellen, wird kein spezielles Zielmodul benötigt – dies ist der PC“, so *R. Barth*. „Bisher kamen fertig definierte Regler zum Einsatz, die nicht mehr beeinflussbar waren“, teilt Dr. *D. Janssen* mit. „Nun lassen sich individuelle Regler erzeugen und einbinden.“

Zur Nürnberger Fachmesse SPS/IPC/Drive 2011 wird die Version Twincat 3 nun releast. Die Auslieferung ist für Anfang 2012 vorgesehen. In der Zwischenzeit wird die Infrastruktur für die Lizenzierung freigeschaltet.

## Literatur

- [1] Beckhoff Automation GmbH, Verl: [www.beckhoff.de/Twincat3](http://www.beckhoff.de/Twincat3)
- [2] IEC 61131-3:2003-01 Programmable controllers – Part 3: Programming languages. Genf/ Schweiz: Bureau Central de la Commission Electrotechnique Internationale
- [3] *Barth, R.*: Optimale Ausnutzung der Multi-Core-Technologie. etz Elektrotechnik + Automation 131 (2010) H.2 S. 8–10 ■



**Halle 7, Stand 406**