



Industrial Ethernet Journal



ETHERNET · WIRELESS · SECURITY

QR-Code: Scannen Sie den QR-Code mit Ihrem Smartphone, um direkt auf die Homepage von Beckhoff zu gelangen.



Steuerungstechnik im Sub-100µs-Bereich

Optimales Zusammenwirken von Steuerungshard- & -software ermöglicht Verlagerung komplexer Regelkreise in zentrale Steuerung

(S.6)



Profinet S.18
Der Weg vom Holz zum Haus mithilfe von Profinet

Marktübersicht S.22
Sie sorgen für Übersetzung: Serielle Adapter für Ethernet

Computerkriminalität S.28
Cyber-Angriffe werden immer aggressiver

Sichere Fabriken S.31
Strategien für den Schutz der Produktions-IT

Safety S.32
SafetyNet p für optimierten Fahrzeug-Recyclingprozess

Interview S.34
Olaf Schilperoord über Bedeutung und Sicherheit von WLAN

Produktübersicht S.36
Kabellose Kommunikation: Wireless-Produkte

Power-over-Ethernet S.42
Daten und Strom über eine Leitung transportieren

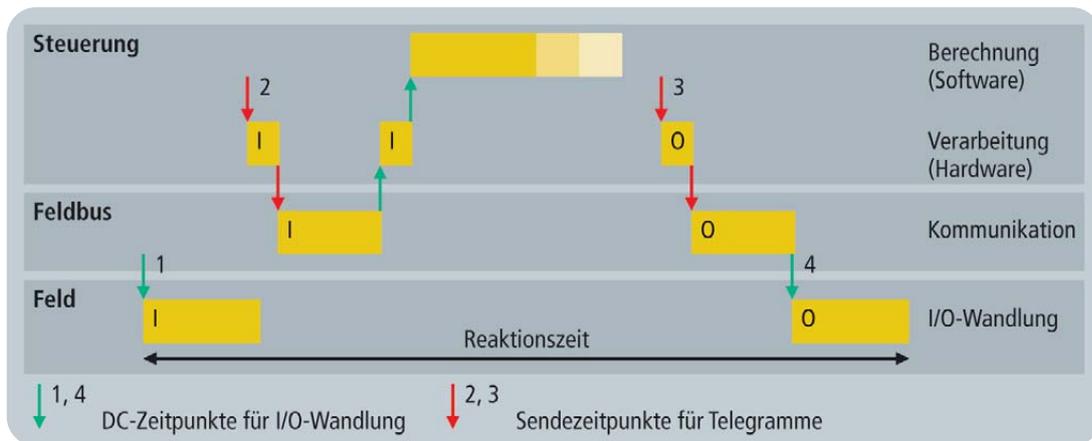


Bild 1: Reaktionszeit und Zeitsteuerung



Halle 9
Stand F06

Steuerungstechnik im Sub-100µs-Bereich

Optimales Zusammenwirken von Steuerungshard- & -software ermöglicht Verlagerung komplexer Regelkreise in zentrale Steuerung

Schnelle Regelkreise mit sehr kurzen Reaktionszeiten in der zentralen Steuerung zu schließen, stellt auch moderne Steuerungen vor eine Herausforderung. Allein die Schnelligkeit des Feldbussystems und die Leistungsfähigkeit der eingesetzten Hardware reichen hierfür nicht aus.

Um echte Reaktionszeiten deutlich unter 100µs erzielen zu können, müssen alle eingesetzten Soft- und Hardwarekomponenten optimal aufeinander abgestimmt sein. Mit seiner eXtreme Fast Control(XFC)-Technologie hat Beckhoff hier frühzeitig die Weichen gestellt. Twincat 3.1, die aktuelle Softwaregeneration von Beckhoff, verfolgt diesen Weg weiter, indem sie die hundertprozentige Ausnutzung der Prozessorleistung ermöglicht. Durch die exklusive Nutzung von Prozessorkernen für Echtzeitanwendungen steht diesen nun mehr Rechenleistung zur Verfügung. Twincat erkennt von Windows nicht genutzte CPUs und kann diese per Konfiguration in das Echtzeitsystem einbinden. Das führt zu einer optimalen Skalierung und Ausschöpfung der Systemressourcen für den jeweiligen Anwendungszweck und ermöglicht die Verlagerung komplexer Regelkreise in die zentrale Steuerung ohne Leistungseinbußen für weitere Echtzeitaufgaben. Moderne,

Ethernet-basierte Feldbussysteme, wie z.B. Ethercat, bieten einer übergeordneten Steuerung die Möglichkeit, im Sub-100µs-Bereich mit den angeschlossenen Komponenten zu kommunizieren. Zykluszeiten von 12,5µs hat Beckhoff bereits auf der Hannover Messe 2012 vorgestellt; eine weitere Reduktion ist theoretisch möglich. Um aber auch echte Reaktionszeiten deutlich unter 100µs zu erzielen, reicht ein schneller Feldbus alleine nicht aus; vielmehr müssen sämtliche Steuerungskomponenten entsprechend leistungsfähig und optimal aufeinander abgestimmt sein, wie in der XFC-Technologie von Beckhoff. Die Reaktionszeit ist definiert als die Zeit zwischen dem Auftreten eines externen Signals und der Ausgabe einer Reaktion an einen Aktor. Sie beinhaltet die Erfassung und Wandlung eines physikalischen Signals im Sensor, die Kommunikationszeit zur Steuerung, die Verarbeitung innerhalb der Steuerung, die Übertragungszeit zum Aktor und die Wandlung in ein physika-

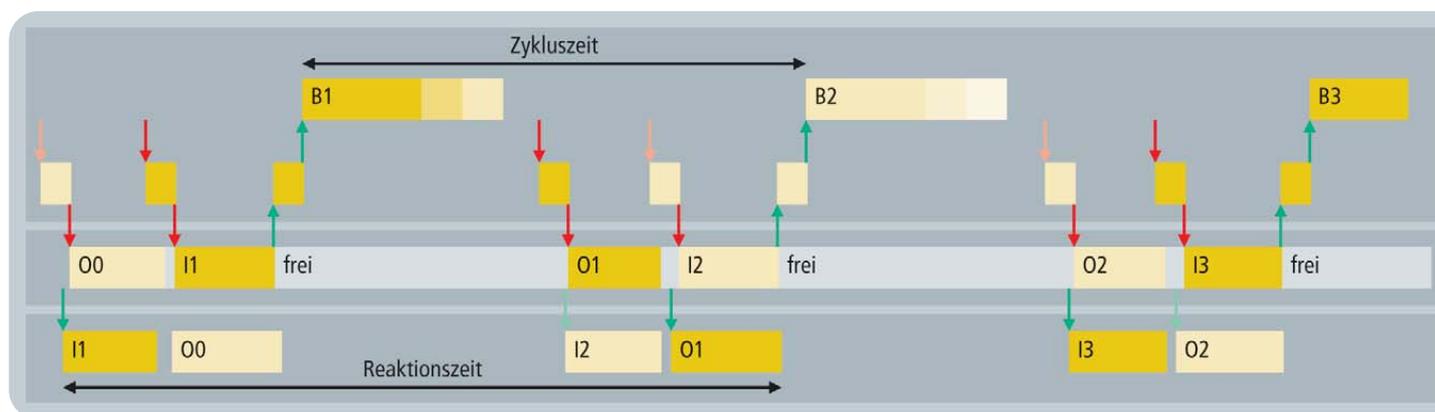


Bild 2: Reaktionszeit versus Zykluszeit

liches Signal. Für viele regelungstechnische Prozesse ist die Reaktionszeit ein entscheidendes Kriterium: je kürzer, desto schneller kann der Regler auf Abweichungen vom Sollzustand reagieren. Regelkreise in der Automatisierungstechnik, die sehr schnelle Reaktionen erfordern, werden daher in speziellen Geräten lokal geschlossen und nicht in der übergeordneten Steuerung. Typische Beispiele sind Antriebsverstärker, die den Stromregelkreis, den Geschwindigkeitsregelkreis und häufig auch den Positionsregelkreis direkt im Antriebsverstärker schließen und nur noch die Sollpositionen von der überlagerten Steuerung erhalten. Für viele Anwendungen ist das ein praktikabler Ansatz. Allerdings besitzen solche dezentralen Regler häufig bis zu 100 Parameter, die zur Anpassung an die konkrete Aufgabenstellung einzustellen sind. Wird die Regelstrecke komplexer und abhängig von vielen Einflussfaktoren, die nicht am lokalen Gerät verfügbar sind, ist eine zentrale Regelung in einer leistungsfähigen, zentralen Steuerung von Vorteil. In ihr sind alle Informationen gleichzeitig vorhanden und Zustände, die eventuell aus anderen Regelkreisen stammen oder durch weitere Sensoren erfasst werden, können berücksichtigt werden. Ein weiterer, wichtiger Vorteil einer zentralen Steuerung ist die Möglichkeit, den Regelalgorithmus speziell an die konkrete Anwendung anpassen zu können – direkt vom Anwen-

der in seiner gewohnten Programmierumgebung und -sprache. Hat die zentrale Steuerung, aufgrund der zusätzlich notwendigen Kommunikation, jedoch eine deutlich langsamere Reaktionszeit zur Folge, gehen die Vorteile der Lösung meist wieder verloren.

Feldbuskommunikation für möglichst kurze Reaktionszeit

Eine wesentliche Voraussetzung für eine möglichst kurze Reaktionszeit eines verteilten Steuerungssystems ist die schnelle und deterministische Kommunikation. Ethercat bietet hierfür mit seiner effizienten Übertragung und durch die Eigenschaft der verteilten Uhren (Distributed Clocks) eine gute Basis. Die Wandlung von Eingangssignalen muss so angestoßen werden, dass das Ergebnis unmittelbar vor dem nächsten Kommunikationszyklus fertig ist und zur Steuerung gelangt. Direkt nach der Berechnung der Sollwerte sollte dann der Ausgangskommunikationszyklus angestoßen werden, sodass die Werte gerade im Aktor angekommen sind, wenn deren Wandlung ansteht. Für die Ausgangsdaten gilt das Kriterium, dass sie 'früh genug' gesendet werden; bei den Eingangsdaten darf aber nicht 'zu früh' kommuniziert werden, da ansonsten deren Wandlung noch nicht abgeschlossen ist. Um den gemeinsamen Kommunikationszyklus für Ein- und Ausgänge am Ende des Steue-

rungszyklus anzustoßen, reicht die Standardbetriebsart eines Ethercat-Masters nicht aus, da wertvolle Zeit ungenutzt verstreichen würde.

Zeitgesteuerter Doppel-Sendebetrieb

Durch die Trennung von Eingangs- und Ausgangskommunikation besteht die Möglichkeit, jeweils den optimalen Zeitpunkt für den entsprechenden Sendepunkt zu berechnen und auszuführen. Eine strikte Zeitsteuerung der Kommunikation ermöglicht das zeitlich optimierte Versenden der Eingangs- und Ausgangstelegramme und die darauf abgestimmte Einstellung der Distributed Clocks in den Sensoren und Aktoren. Eine weitere wichtige Eigenschaft von Ethercat, das parallele Fahren verschiedener Kommunikationszyklen und -zykluszeiten auf einem System, sollte ebenfalls erhalten bleiben. Neben der extrem schnellen Regelung sollen auch noch andere Signale in 'normaleren' Zykluszeiten auf demselben Ethercat-Strang ausgetauscht werden können. Die verbleibende Bandbreite kann daher geschickt genutzt werden, um den Kommunikationsbedarf anderer Tasks in der Steuerung abzudecken. In der Konfigurationsphase berechnet der Twincat-Systemkonfigurator für einen Zeitraum, der sich als kleinstes gemeinsames Vielfaches aller beteiligten Zykluszeiten ergibt, die exak-

ten Sendezeitpunkte aller Ethercat-Telegramme und übergibt diese Tabelle an den Ethercat-Master. Nach diesem Zeitraum wiederholt sich der Ablauf, sodass die Tabelle vom Master immer weiter abgearbeitet werden kann. Verbleibende zeitliche Lücken in der Tabelle kann der Master dann eigenständig mit azyklischen Kommunikationsanforderungen füllen – z.B. mit Parameterdaten über das CAN Application Protocol over Ethercat (CoE). Eine strikte Zeitsteuerung ist aber nicht leicht einzuhalten. Ein Vorteil von Ethercat, der den Master-Betrieb auf handelsüblichen (Commercial Of The Shelf, kurz COTS) Prozessoren und Ethernet-Controllern (MAC) ermöglicht, steht hier ein wenig im Widerspruch zum geforderten Determinismus. Eine reine Softwarelösung reicht nicht aus, da sich die Laufzeit des auszuführenden Codes nicht vollständig vorausberechnen lässt. Neben dem offensichtlichen Einfluss der bedingten Ausführung der vom Anwender erstellten Software unterliegt die Abarbeitungszeit außerdem Schwankungen, die von der Prozessorarchitektur abhängig sind. Da diese nicht zu vernachlässigen sind, wird darauf im weiteren Verlauf noch einmal eingegangen. Entscheidend für die Einhaltung der Telegrammsendezeiten ist daher die Unterstützung der unterlagerten Ethernet-Hardware, die die einzelnen Telegramme zu dem exakt berechneten Zeitpunkt verschickt. Da diese Problema-

Release von TwinCAT 3.1

Mit dem Release der TwinCAT-Version 3.1 zur Hannover Messe 2013 verfügt der Anwender über die neue Eigenschaft der exklusiven Echtzeit-Cores. Der zeitgesteuerte Doppelsendebetrieb wird als Update zu TwinCAT 3.1 Mitte dieses Jahres verfügbar sein. Damit können auf Multiprozessor-Systemen die im Text beschriebenen Zykluszeiten und Leistungswerte erreicht werden. Die ersten Systeme mit QoS-Cache werden gegen Ende 2013 erhältlich sein.

tik auch außerhalb der Automatisierungstechnik eine Rolle spielt, kann weiterhin allgemein verfügbare Hardware eingesetzt werden. Erste MACs mit zeitgesteuerten Sendequellen sind von großen Herstellern bereits auf dem Markt, und diese Eigenschaft scheint in viele neuere MACs einzufließen. Die eingesetzten Ethernet-MACs der Embedded-PC-Serie CX von Beckhoff sowie der externe CU2508 als Ethernet-Portmultiplier unterstützen das zeitgesteuerte Sendebetrieb ebenfalls in Hardware und können für diesen Einsatzzweck herangezogen werden.

Moderne Prozessoren für schnelle Steuerungsaufgaben

Die PC-Steuerungstechnik profitiert vom schnellen technischen Fortschritt der allgemeinen IT-Welt. Allerdings bekommt man diesen Fortschritt nicht in Gänze

geschenkt, sondern muss technologische Erweiterungen in Hard- und Software vornehmen, um ihn für die Automatisierungswelt nutzen zu können. Die Echtzeiterweiterung für Windows in TwinCAT und die Nutzung von Ethernet als Feldbus durch Ethercat sind Beispiele hierfür. Aufgrund physikalischer Grenzen ging die Entwicklung der Prozessoren zur Erhöhung der Rechenleistung von stetig steigenden Taktfrequenzen über zu einer Vervielfachung der Rechenkerne pro Chip. Man spricht hierbei von 'Chip Multiprocessing' (CMP). Aktuell sind über Doppelkern-CPU's hinaus auch Vierfach- bzw. Achtfach-Kerne zu vertretbaren Kosten verfügbar. Diese Entwicklung kommt den softwarebasierten Automatisierungslösungen, wie TwinCAT 3, sehr entgegen, da diese in der Lage sind, Aufgaben je nach Anzahl der verfügbaren CPU-Cores zu verteilen. D.h., funktionale Einheiten, wie HMI, PLC-Control, PLC-Runtime, NC, können mit weniger Aufwand als heute auf dedizierte Cores verteilt werden. TwinCAT 3 erleichtert dem Anwender durch entsprechende Konfigurations- und Diagnosewerkzeuge die Nutzung von Multi-Core-Systemen. So lassen sich im TwinCAT System Manager z.B. die Laufzeiten der Echtzeittasks beobachten und Prioritäten bzw. Ablaufreihenfolgen der Tasks manuell konfigurieren. Über konfigurierbare Core-Affinitäten lassen sich Tasks statisch einem

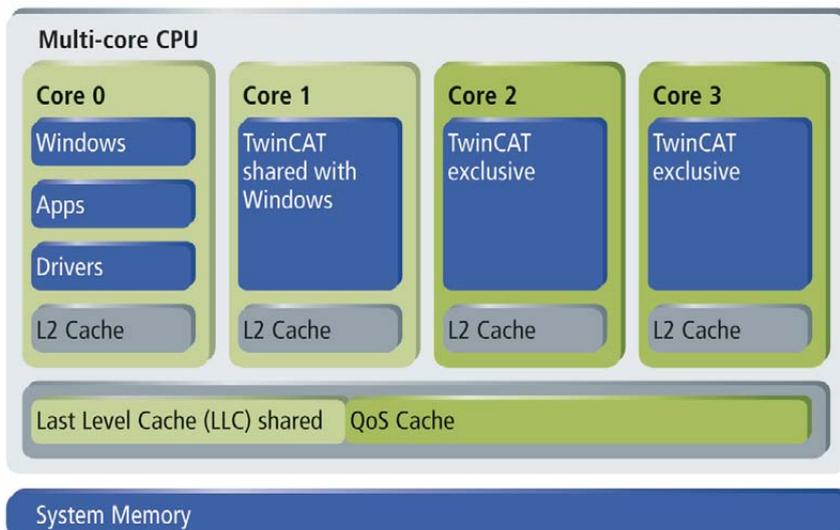


Bild 3: Exklusive CPU-Cores für TwinCAT-Echtzeit-Tasks

Core zuordnen. Allerdings unterliegen heutige Multicore-Systeme aufgrund der gemeinsamen Nutzung von Arbeitsspeicher und Prozessor-Cache (hierbei: Last Level Cache) der Einschränkung, dass sich die Cores gegenseitig beeinflussen können. Das hat unter Umständen auch Einfluss auf die Ablaufzeiten des Codes in der Echtzeitumgebung. Das muss bei der Konfiguration der Zykluszeiten von Echtzeittasks im Sub-100- μ s-Bereich beachtet werden. Für die Beurteilung des jeweiligen Systemverhaltens bringt Twincat die geeigneten Messmethoden mit. Auch in Nicht-Echtzeitanwendungen kann die gemeinsame Nutzung von Systemressourcen durch CPU-Cores nachteilig sein. CMP-Systeme sind ursprünglich für die performante Bearbeitung durch optimale Nutzung aller Systemressourcen von Multithreading-Einzelapplikationen designed. Aufgrund der verfügbaren Rechenleistung können sie aber auch – mithilfe von Virtualisierungstechnologien – mehrere Betriebssysteme auf einem Rechner parallel bearbeiten. Hierbei ist die gegenseitige Beeinflussung unter Umständen ebenfalls störend. Um hier Abhilfe zu schaffen, wurden bzw. werden für Prozessoren und Betriebssysteme neue Hardware- und Softwarelösungen von Prozessor- und Betriebssystemherstellern entwickelt und implementiert. Die für die Leistungsfähigkeit eines Rechners entscheidenden Systemeigenschaften sind, neben der Taktfrequenz des Prozessorkerns, vor allem die Größe des Prozessor-Caches und die Bandbreite des Arbeitsspeicherbusses. Eine Softwarelösung zur Optimierung der Cache-Nutzung ist das sogenannte Cache Coloring bzw. Page Coloring. Hierbei wird über die virtuelle Speicher-verwaltung der CPU ein zusammenhängender Speicher aus Sicht des Caches erzeugt und somit seine verfügbare Größe effizienter genutzt. Diese Funktion ist z.B. in der dynamischen Speicher-verwaltung eines Betriebssystems implementiert oder auch in der virtuellen Speicher-verwaltung eines Hypervisors enthalten. Für die Priorisierung von parallel ablaufenden Anwendungen besonders interessant ist ein Last Level Cache, der eine Quality of Service(QoS)-Funktion bietet. Über diese Hardware-Eigenschaft lassen

sich Cache-Bereiche für hochprioräre Anwendungen reservieren bzw. die Cache-Größe kann für niederprioräre Anwendungen begrenzt werden. Das aktuelle Non-plusultra in Sachen Skalierbarkeit und Ressourcenaufteilung in SMP(Symmetric Multiprocessing)-Systemen ist ein 'echtes' Multiprozessorsystem in Non Uniform Memory Access(NUMA)-Architektur. Hier hat nicht nur jeder Prozessor seinen eigenen Last Level Cache, sondern auch seinen eigenen lokalen Arbeitsspeicher. Dabei haben alle Prozessoren einen gemeinsamen Adressraum und sind über einen Bus miteinander verbunden. Allerdings ist dies auch die aufwendigste und somit teuerste aller Lösungen. Ab der Version 3.1 bietet Twincat die Möglichkeit, Prozessoren bzw. Prozessorkerne eines PC-Systems exklusiv für Echtzeitanwendungen zu nutzen. Twincat erkennt von Windows nicht genutzte CPUs und kann diese per Konfiguration in das Echtzeitsystem einbinden. Für Twincat und dessen Echtzeitanwendungen, wie z.B. PLC und Motion Control, geschieht dies vollkommen transparent, während die betreffende CPU für Windows unsichtbar ist. Twincat konfiguriert die Echtzeit-CPU in der Art, dass der gesamte Adressraum von den Echtzeit-CPU aus sichtbar ist, aber umgekehrt private Speicherbereiche vor Windows 'versteckt' werden können. Für die Implementierung dieser Funktionalität wurde bewusst auf Virtualisierungsfunktionen verzichtet, um eine zusätzliche Softwareschicht zu vermeiden, die eventuell zu Leistungseinbußen bzw. Latenzzeiten führen könnte. Genutzt werden kann diese Funktion in allen Windows-Versionen ab Windows XP SP2. Mit dieser neuen Twincat-Eigenschaft bekommen die Echtzeitanwendungen mehr Rechenleistung, da der betreffende Prozessorkern zu 100% genutzt werden kann. Lokale Ressourcen eines Cores, z.B. der 'Second Level Cache' oder der Speicher einer Numa-Architektur, können dem Echtzeitanteil der Software klar zugeordnet werden. Außerdem wird die statische Konfiguration von QoS-Caches zukünftiger Prozessoren vereinfacht. Basierend auf den zuvor beschriebenen Hardwareeigenschaften erlaubt diese Lösung eine opti-

male Skalierung und Nutzung der Systemressourcen für den jeweiligen Anwendungszweck. Zykluszeiten bis hinunter in den 10 μ s-Bereich zeigen die Möglichkeiten heute verfügbarer Technologien auf, die für die Anwender von Twincat und Ethercat nutzbar sind.

Verlagerung von Regelkreisen eröffnet neue Möglichkeiten

Die Verlagerung schneller Regelkreise mit sehr kurzen Reaktionszeiten in die zentrale Steuerung stellt auch für moderne Kommunikationssysteme und Steuerungshardware eine Herausforderung dar. Twincat, die auf allgemeiner PC-Hardware aufsetzende Steuerungssoftware, und Ethercat, das auf allgemeinen Ethernet-MACs basierende Highspeed-Kommunikationssystem, bilden die Basis der Lösung. Aber erst das optimale Zusammenspiel zwischen Soft- und Hardware und die Ausnutzung aktueller Hardwareeigenschaften durch Twincat 3 schaffen die Voraussetzung für den sicheren, deterministischen Betrieb schneller Regelkreise in der zentralen Steuerung. Für den Anwender ergeben sich dadurch neue Möglichkeiten, bestimmte Regelungen selbst zu realisieren, auf deutlich mehr Informationen zuzugreifen, seine gewohnte Entwicklungsumgebung zu nutzen und – nicht zuletzt – auch Hardwarekosten einzusparen. ■

www.beckhoff.de



Autor: Dr.-Ing. Dirk Janssen, Leiter Software-Entwicklung System, CNC und I/O bei Beckhoff Automation



Autor: Ramon Barth, Leiter Software-Entwicklung System, HMI und Echtzeit bei Beckhoff Automation