

Virtuelle Steuerungen für Motion und Robotik

Die Echtzeithürde überwinden



Jetzt anhören!



Gegenüber herkömmlicher Steuerungstechnik verspricht die Automatisierung mit virtuellen Steuerungen erhebliche Einsparpotenziale bei Beschaffung, Wartung und Betrieb. Unter bestimmten Voraussetzungen eignen sie sich sogar bei Anwendungen in denen z.B. koordinierte Verfahrbewegungen hohe Anforderungen an das Echtzeitverhalten haben. Welche dies sind und wie sich diese Herausforderungen bewältigen lassen, erläutert dieser Beitrag.

Ein großer Vorteil von virtuellen Steuerungen ist ihre deutlich einfachere Wartbarkeit. So können Security-Updates viel schneller und umfassender ausgerollt werden, um Produktionssysteme gegen gezielte oder zufällige Attacken zu härten. Und weil mit der Zertifizierung nach IEC 61508 für SIL3-Anwendungen jetzt auch virtuelle Sicherheitssteuerungen wie Codesys Virtual Safe Control SL zur Verfügung stehen, gibt es keine Einsatzbegrenzungen mehr. Das zeigen auch die ersten Anwendungen mit virtuellen Steuerungen im Produktiv-

betrieb: Gehostet auf IT-Servern ersetzen sie beispielsweise in Produktionslinien der Audi AG physikalische SPSen.

Für Echtzeitanforderungen geeignet

Und dennoch bleibt ein Zweifel: Wenn virtuelle Steuerungen auf abgesetzten IPCs in der Nähe der Maschine oder sogar auf entfernten Servern im Rechenzentrum betrieben werden, lassen sich dann weiterhin Echtzeitanwendungen mit hohen Anforderungen an Performance und Jitter

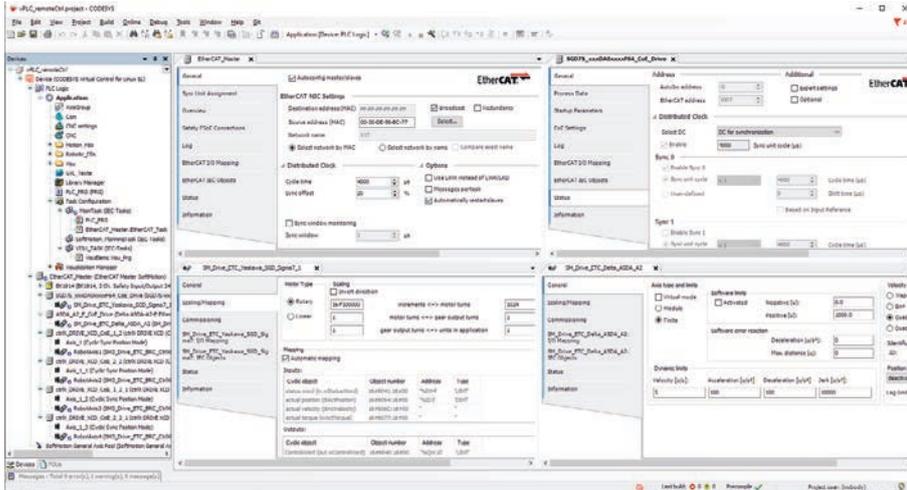
realisieren? Vor allem dann, wenn koordinierte Verfahrbewegungen benötigt werden, wie das bei Motion-Controllern, CNC- und Robotiksteuerungen der Fall ist. Wie sieht es aus, wenn größere Distanzen zwischen Steuerung und Antrieben bzw. E/As zu überbrücken sind?

Zur Realisierung von koordinierten Verfahrbewegungen für Motion-, CNC- oder Robotikaufgaben benötigt man geeignete Servoantriebe mit digitalen Feldbuschnittstellen. Aufgrund ihrer Echtzeitfähigkeit haben sich CANopen und zunehmend Ethercat als Bussysteme etabliert. Allein die in Codesys integrierte Motion-Lösung bietet spezifische Treiber für CANopen- und Ethercat-Antriebe von mehr als zwei Dutzend Herstellern sowie generische Treiber für CiA402/CoE- und SoE-kompatible Systeme.

Motion, CNC und Robotik auf SPSen

Wurden früher dedizierte Controller eingesetzt, um koordinierte Verfahrbewegungen zu steuern, so lassen sich die Anfahrpunkte der Antriebe mit ihren Dynamiken heute auch in der SPS berechnen und per Feldbus übertragen. Das reduziert den Hardwareaufwand und die Kosten. Voraussetzung dafür sind eine geeignete CPU-Leistung sowie verfügbare Schnittstellen an der SPS. Ist die Leistungsfähigkeit der eingesetzten Steuerung unzureichend, hat man verloren und muss auf ein leistungsfähigeres Modell wechseln – mit Konsequenzen, die teuer und aufwendig sein können. Oder andersherum: Bei der Abschätzung des Leistungsbedarfs seiner Applikation sollte man vorab sehr genau prüfen, ob Motion Control-, CNC- oder Robotik-Funktionen benötigt werden und die Auswahl geeigneter Steuerungen nach der verfügbaren Leistungsfähigkeit filtern. Dabei fällt auf: Mit SoftSPSen oder virtuellen Steuerungen ist man deutlich flexibler.

Natürlich sind auch die Leistungsreserven von IPCs oder Servern mit mehreren CPU-Kernen irgendwann aufgebraucht, wenn darauf mehrere virtuelle Steuerungen ausgeführt werden. Aber im Vergleich zu dedizierten SPSen sind die Reserven um ein Vielfaches größer, und sie können bei Bedarf abgerufen werden. Wenn zum Beispiel bei der Projektierung klar wird, dass zu-



► Konfiguration einer Motion- und Robotik-Anwendung im Codesys Development System

sätzlich zur Logiksteuerung noch ein Motion Controller nötig wird, kann man durch eine Lizenz einfach aufrüsten. Damit können komplexen Achsgruppen genutzt und die zugehörigen aufgerufenen Bibliotheksbausteine freigeschaltet werden. Bei physikalischen Geräten ist dies in vielen Fällen schlicht undenkbar - ein klarer Pluspunkt für virtuelle Steuerungssysteme.

Unterschiede der Architekturen

Grundsätzlich gibt es für Motion Control zwei Architekturansätze, die die Anforderungen an die Echtzeit der Kommunikation stark beeinflussen. Da ist zum einen der gerade beschriebene zentrale Ansatz, bei dem die Trajektorie in der Steuerung gerechnet wird, wobei die einzelnen Achsen dieser Vorgabe folgen. Dafür ist eine ständige, jitterfreie Kommunikation erforderlich. Dem gegenüber steht der dezentrale Ansatz, bei dem intelligente Antriebe oder untergeordnete Steuerungen diese Motion-Funktion implementieren und von der übergeordneten Steuerung nur Befehle erhalten. Hierfür ist keine zyklische, synchronisierte Kommunikation erforderlich.

Insbesondere Industrieroboter können somit ohne spezielle Programmiersprachen wie Karel, Inform oder Rapid auskommen. Von der SPS aus werden stattdessen Kommandos über Standard-Feldbussysteme wie Profinet an die Robotersteuerung geschickt, die für die gewünschte Bewegung sorgen. Wurden in der Vergangenheit proprietäre Bibliotheken wie mxAutomation dafür angeboten, so etabliert sich mit dem Standard Robot Command Interface (SRCI) gerade ein

übergreifender Standard mit einem Satz aus 115 Befehlen; z. B. für lineare Bewegungen bis hin zu komplexeren Befehlen wie Kraftsteuerung. Die virtuelle SPS kann in solchen Anwendungsfällen problemlos räumlich distanziert ausgeführt werden und ihre Kommandos per Feldbus schicken, weil die eigentliche Robotersteuerung weiterhin vor Ort am Roboter erfolgt.

Abgesetzte industrielle Antriebssysteme

Ob virtuelle Steuerungen für zentrale gerechnete Motion-Aufgaben geeignet sind, entscheidet sich also vor allem bei der Kommunikation zwischen Steuerung und Antrieb. In hardwareunabhängigen Systemen wie Codesys werden die Befehle und Services des Protokollstacks in Bibliotheksbausteinen implementiert, abstrahiert von der tatsächlich eingesetzten Hardware. Dieser Code wird mithilfe integrierter Compiler der Entwicklungsumgebung zusammen mit der SPS-Applikation implizit in nativen Maschinencode für die CPU der eingesetzten Steuerung übersetzt. Lediglich bei der Übersetzungsdauer und Codegröße bemerkt man, dass jetzt zusätzlich zum selbst erstellten Logikprogramm auch noch das Feldbusprotokoll im Kompilat enthalten ist.

Damit dieser Code zusätzlich zykluskonsistent zur SPS-Applikation ausgeführt werden kann, muss zum einen das Antriebssystem im Programmierool richtig konfiguriert sein. Das beinhaltet abhängig vom eingesetzten System Parameter wie Netzwerkadressen, busspezifische Einstellungen sowie technische Eigenschaften der eingesetzten Achsen und Motoren,

wie Getriebeübersetzung, Achstypen und -grenzen sowie deren Dynamik. Zum anderen muss eine Motion-Task im Projekt mit passender Zykluszeit und Priorität angelegt sein, damit Antriebe und Motoren ihre Sollwerte „pünktlich“ bekommen und rund laufen. Per Definition dürfen dabei die Kabellängen z. B. von Ethercat eine Länge von maximal 100m aufweisen. Wenn gleichzeitig die Anzahl der angesteuerten Antriebe eine Menge von 10 bis 20 nicht übersteigt, sind in der Regel keine weitere Maßnahmen bzw. Geräte wie zusätzliche Switches oder Verstärker erforderlich. Neben den harten Echtzeiteigenschaften von Ethercat ist dies ein Grund für die hohe Verbreitung des Systems.

Motion-Kommunikation über große Distanzen

Was aber, wenn virtuelle Steuerungen in einem nahegelegenen Rechenzentrum betrieben werden, bei dem die benötigten Kabellängen über das Limit gehen? Oder wenn sehr viele Antriebe angesteuert oder zusätzlich zur Ethercat-Kommunikation noch weitere Daten im Bus ausgetauscht werden müssen, z. B. für ein Vision-System? Dafür gibt es spezielle Hardware von Ethercat-Anbietern, welche die Daten über Lichtwellenleiter (LWL) übertragen und damit große Distanzen bis zu mehreren Kilometern überbrücken. So lassen sich Motion-Anwendungen mit virtuellen Steuerungen realisieren, die z. B. im abgesetzten Serverraum ausgeführt werden. Kompliziert wird es, wenn vor Ort an der Maschine gleichzeitig ältere Bustechnologien mit Sensoren und Aktoren verbaut sind, deren Daten man nicht ohne weiteres in den Motion-Bus einbetten kann. Es ist unerlässlich, diese Daten ebenfalls über den LWL zu übertragen. Technisch machbar ist es, diese Daten über spezielle Protokolle parallel zu Ethercat zu verpacken und via LWL zu verschicken.

Bei der Einbettung der Daten in ASICs sind diese Daten verlustfrei in nahezu beliebigen Frequenzen gleichzeitig über den LWL übertragbar. Genau das verspricht ein Ansatz mit dem Arbeitstitel „Robo/TSN“ der Firma Missing Link Electronics. Das System fasst Daten unterschiedlicher Bussysteme per FPGAs zusammen und überträgt diese deterministisch. Abhängig vom Medium, typi-



schersweise LWL, können damit mehrere hundert Gigabit an Daten im ns-Bereich übertragen werden. In einem patentierten Verfahren werden die Daten derart getunnelt, dass alle Eigenschaften und Informationen des ursprünglichen Systems erhalten bleiben: Funktional sichere Protokolle wie FSoE (Fail Safe over Ethercat) oder Profisafe können also problemlos genutzt werden. Aufgrund der Kapselung lassen sich die Feldbusinformation sogar in offenen IT-Netzen sicher nach IEC 62443 übertragen.

Wichtig für User von Motion- und SPS-Systemen wie Codesys: Im Gegensatz zu anderen LWL-Systemen ändern sich Bedienung und Konfiguration durch diese Art des Datentunnels nicht. In ersten Tests wurde z. B. die Codesys-Motion-Applikation genau wie bisher konfiguriert: Ein verfügbarer Ethernet-Port wird zum Ethercat-Master samt angeschlossenen E/As und Antrieben – bei der Konfiguration merkt man nicht, dass dieser Port durch Robo/TSN getunnelt wird. Auch die o. g. weitere Parametrierung und Projektierung ändert sich nicht im Vergleich zur Nutzung an einem Standard-Ethernet-Port. Was sich ändert, ist die „Netzwerkkarte“: Sie beinhaltet den entsprechenden ASIC und stellt Ethernet- und andere Kommunikationsports zur Verfügung. Intern tunnelt sie die konfigurierten Daten und überträgt sie über das angeschlossene Medium, um dann an der Gegenstelle durch eine PCIe-Netzwerkkarte wieder entpackt und wie ursprünglich bereitgestellt zu werden. Sowohl am Hostrechner, auf dem virtuelle Steuerungen betrieben werden, als auch im abgesetzten Schaltschrank oder in den Maschinenteilen, in denen die entsprechenden Endgeräte betrieben werden, sind die mit dem ASIC ausgestatteten Karten erforderlich.

Fazit

Die Herausforderungen die sich bei virtuellen Steuerungen aus der räumlichen Trennung von den zu Grunde liegenden Rechnersystemen und den erforderlichen Antrieben für Bewegungssteuerung und Robotik ergeben, sind lösbar – abhängig vom Anwendungsfall und den genutzten Komponenten – und zwar durch den Einsatz von Stan-

dards wie Ethercat und SRCI oder von Lichtwellenleitern als Übertragungsmedium. Durch moderne, integrierte Schaltkreise ändert sich die Projektierung gegenüber lokalen Systemen nicht. Damit ist klar: Virtualisierte Steuerungstechnik ist für alle Anwendungsbereiche einsetzbar. ■

Codesys GmbH
www.codesys.com

- Anzeige -



Akzeptiert noch härtere Arbeitsbedingungen als die Generation X

Hitze, Kälte, Staub oder Nässe – die JSmart707 lebt, um zu arbeiten

Egal, wie hart das Umfeld ist, die JSmart707 von EXOR macht einfach weiter. Dank des vollständigen IP67-Umgebungsschutzes können Sie diese High End HMI genau da platzieren, wo der Bediener sie benötigt – ohne kostspielige Schrankinstallation. Und dort arbeitet sie dann jahrelang superzuverlässig. Top Technologie für die Digitalisierung – in bewährter Qualität.

EXOR:
Der Werkzeugkasten für die Digitalisierung



Panels & Displays



Gateways



Visualisierungs-Software JMobile



CORVINA Cloud



SPS nach 61131



Handheld-Geräte