

Roland Wagner

IEC 61131-3 – Robotik inklusive

Lange Jahre waren SPS-Systeme mit Logikapplikationen voll ausgelastet. Günstige und gleichzeitig hoch performante Prozessorplattformen machen es nun möglich, selbst anspruchsvolle Bewegungssteuerungen auf einem einzigen Automatisierungsgerät komfortabel abzuarbeiten. Speziell die Integration der Robotik-Funktion vereinfacht dabei die Projektierung von komplexen Achsgruppen noch einmal ganz erheblich.

Viele Anwender erinnern sich noch an die Zeit, als sie für sündteures Geld den SPS-Speicher zum Beispiel um 16 kByte (!) erweitern konnten, nur um ein noch aufwendigeres ‚Upgrade‘ auf die nächstgrößere Steuerung zu vermeiden. Für die Ansteuerung koordinierter Verfahrbewegungen in einer Maschine gab es nur spezialisierte Motion-Controller oder CNC-Steuerungen – eine eigene Liga in puncto Engineering und Kosten. Heute ist praktisch jedes Schülerhandy mit einem Gigahertz-Mehrkern-Prozessor ausgestattet – während viele ihrer Besitzer mit der Angabe ‚kByte‘ kaum noch etwas anfangen können. Entsprechend sind intelligente Automatisierungsgeräte mit hoch performanten CPUs und viel Spei-

cher längst Stand der Technik. So haben Gerätehersteller die Möglichkeit, zum Beispiel in einer Panelsteuerung klassische Steuerungsaufgaben und die Maschinenbedienung in einem Gerät zu vereinen. Und auch die Integration von Motion-Aufgaben in einer SPS hat sich mittlerweile etabliert. Welche Entwicklungsschritte haben dazu geführt?

Direkte Ansteuerung von Antrieben

Eine SPS kann per se Bewegungen kommandieren: Einen 24-VDC-Motor an einen analogen oder digitalen Ausgang angeschlossen, den Ausgang gesetzt und schon bewegt sich der Motor – entweder mit fester oder variabler Drehzahl. Der Ausgang benötigt ledig-

lich genügend Leistung beziehungsweise einen Leistungsverstärker. Mit einem Frequenzumrichter kann sogar die Ansteuerung von AC-Motoren erfolgen. Antriebssysteme mit einer analogen ± 10 -VDC- beziehungsweise einer Feldbus-Schnittstelle lassen sich noch komfortabler von der Steuerung bedienen. Verbindet man einen Encoder mit der Motorachse, so ist die Impulsrückführung auf die Steuerung zur Ermittlung und Auswertung von Position oder Geschwindigkeit und somit zur Achsregelung verwendbar – wenn der Antrieb nicht ohnehin mit einer Servo-Funktion ausgestattet ist. Roboterhersteller bieten vereinzelt sogar spezielle SPS-Schnittstellen für ihre proprietären Robotersteuerungen an:

(Bild: Veitru AG / iStockphoto.com lovegulli)

Achs-Konfiguration im Projektbaum sowie deren Parametrierung: Der Anwender definiert Parameter, wie zum Beispiel Achstyp, Dynamikgrenzen oder Bewegungsskalierung.

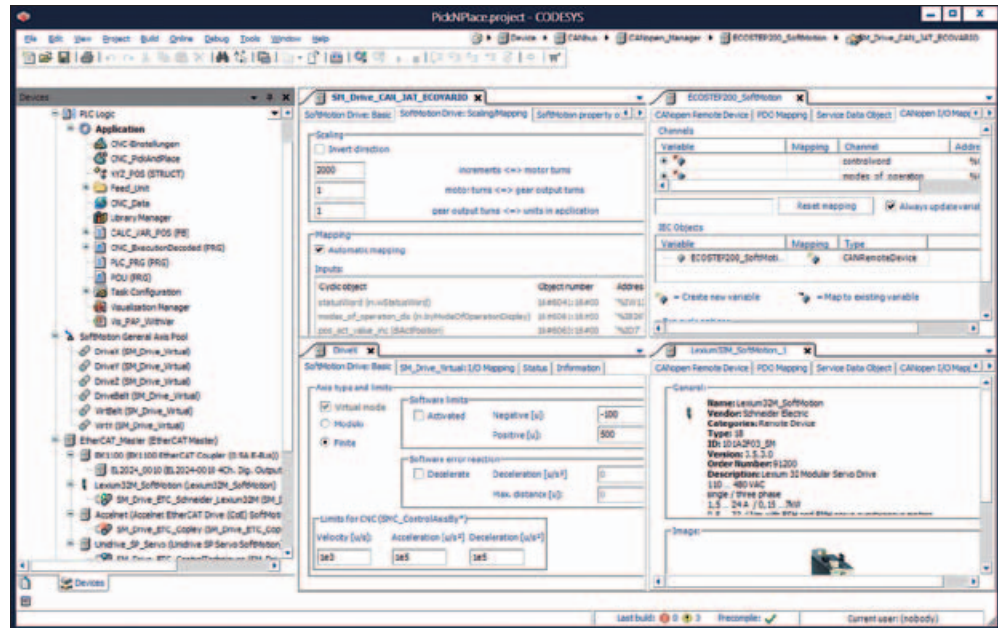
Damit kann die komplette Roboterapplikation anhand von spezifischen Programmbausteinen aus der Logikapplikation heraus kommandiert werden.

So einfach solch eine direkte Ansteuerung auch ist, sie hat doch einen entscheidenden Nachteil: Die Bewegungssteuerung ist sehr spezifisch für das eingesetzte Antriebssystem respektive die verfügbare Schnittstelle. Bei jeder Hardware-Änderung ist eine Anpassung von SPS-Applikation und Ansteuerung erforderlich. Die Kommandierung ist dem Gutdünken des Applikationsentwicklers überlassen und somit für andere nicht immer sofort nachvollziehbar.

Nicht nur deswegen hat sich innerhalb der PLCopen, der Nutzerorganisation der Anbieter von IEC-61131-3-Systemen, bereits um die Jahrtausendwende ein ‚Technical Committee Motion Control‘ zur Standardisierung gebildet. Eine einheitliche Definition von Funktionsbausteinen für klassische Bewegungsaufgaben innerhalb von Steuerungsprogrammen wurde bereits 2001 in einer ersten Spezifikation vorgelegt. Darin lautet die Empfehlung, die definierten Funktionsbausteine in einer IEC-61131-3-Bibliothek zu hinterlegen. Die Verfügbarkeit eines Bibliothekskonzepts innerhalb der Entwicklungsumgebung ist somit bereits eine wichtige Voraussetzung für solch eine standardisierte Bewegungskommandierung. Die Baustein-Definition selbst umfasst zahlreiche typische einachsige Funktionen, etwa

- zum Einschalten sowie Starten der Achse mit vorgegebenen Werten für Position, Beschleunigung und Geschwindigkeit
- oder auch zum Lesen von Diagnose- und Statusinformationen, Stoppen oder Abbrechen von Bewegungen.

Diese Bewegungsfunktionen übergeben der jeweiligen Achse sämtliche



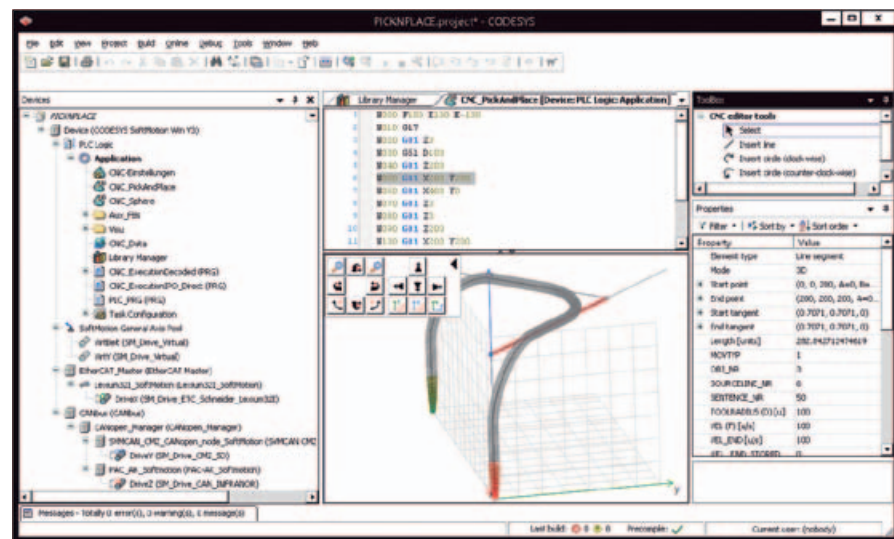
berechneten Daten als abstrahierte IEC-61131-3-Datenstruktur. Das Steuerungssystem muss also solch eine Datenstruktur generieren können. Eine allgemeine Lösung dafür vorzuschreiben ist, angesichts vieler konkreter Implementierungen der PLCopen-Motion-Funktionen in ganz unterschiedlichen Systemen, kaum möglich.

Antriebskonfiguration am Beispiel Codesys

Das Beispiel von Codesys, dem Hersteller-unabhängigen IEC-61131-3-System, zeigt, wie sich die Vorgaben der PLCopen konkret implementieren lassen. Die Datenstruktur wird dabei mit den integrierten Konfiguratoren für Feldbus- und I/O-Systeme erzeugt. Der Anwender bindet dazu den Antrieb

anhand von generischen oder spezifischen Motion-Treibern als Knoten seines Geräts oder Feldbus-Systems ein. Dieser Knoten bringt die Achse mit, deren Name gleichzeitig die entsprechende Datenstruktur repräsentiert. Der Motion-Treiber setzt seinerseits die Datenstruktur auf das jeweilige Protokoll beziehungsweise die Physik um, so dass sich der Anwender darüber keine Gedanken machen muss. In der Praxis bedeutet das: In einer hochprioritären Task der SPS-Applikation werden die Werte für Position, Geschwindigkeit und Beschleunigung zur Laufzeit der Steuerung dynamisch berechnet, dem Treiber übergeben und an den Antrieb geschickt.

Daraus erschließt sich der große Vorteil dieser Standardisierung: Eine er-



Planung einer CNC-Bewegung direkt in der IEC-61131-3-Entwicklungsumgebung: Der CNC-Editor ermöglicht die Planung per G-Code oder durch grafische Elemente in 3D.

(Bilder: 3S-Smart Software Solutions)



THE OFFICIAL DAILY ZUR SPS IPC Drives 2016



„Heute lesen, welches die Innovationen von morgen sind“

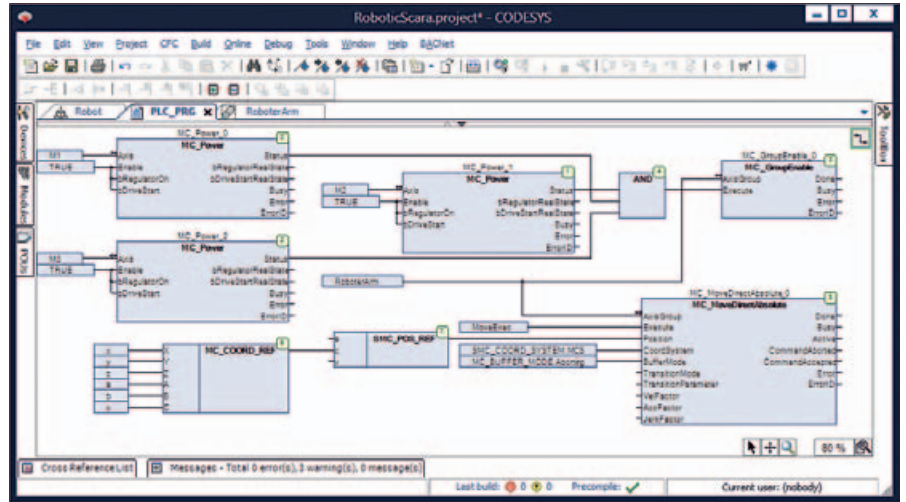
Martin Gugelmann
Vertrieb
exceed electronics AG



Offizieller Medienpartner
sps ipc drives
2016

Erscheinungstermine: 22. – 24.11.2016
Anzeigenschluss: 02.11.2016

Kontakt:
Mediaberatung der OFFICIAL DAILY
zur SPS IPC Drives 2016
Tel.: +49 89 25556-1308
Fax: +49 89 25556-1670
E-Mail: media@computer-automation.de
www.computer-automation.de



Beispiel der Projektierung einer einfachen Robotik-Funktion in einem freigrafi-schen FUP-Editor (CFC).

stellte Motion-Applikation innerhalb des SPS-Programms ist mit unterschiedlichen Antriebssystemen verwendbar. Das heißt: Der Anwendungsprogrammierer kann die Applikation bereits entwickeln und funktional testen, selbst wenn er noch gar keine Antriebe zur Verfügung hat. In Codesys lassen sich dazu beliebig viele virtuelle Achsen einfügen und unabhängig vom verwendeten Antriebssystem in Echtzeit auf der Steuerung ‚rechnen‘. Andererseits kann eine bereits in Betrieb genommene Logikapplikation ohne Applikationsänderung an neue Maschinenanforderungen angepasst werden – nur durch Wechseln der Antriebstreiber beziehungsweise des Antriebssystems.

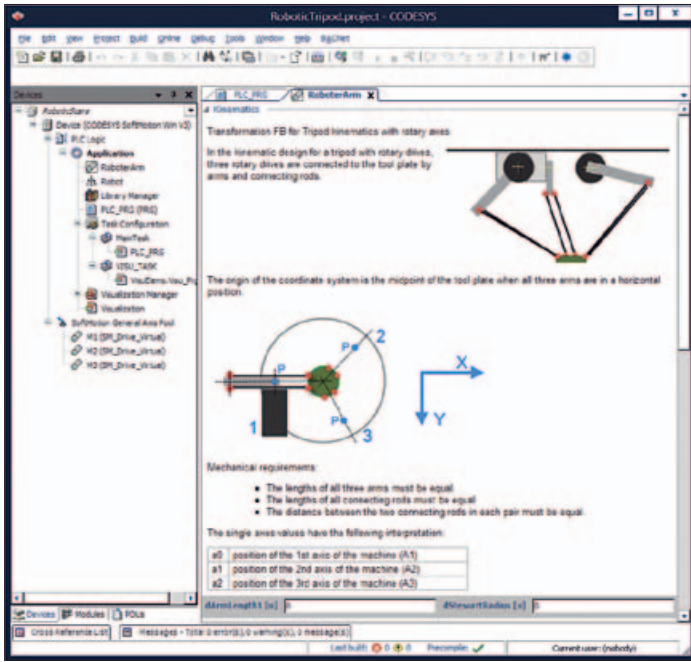
Ein weiteres Plus: Mit der integrierten Visualisierung der genannten IEC-61131-3-Umgebung ist die Inbetriebnahme der Achsen sehr einfach – unabhängig davon, ob es sich um virtuelle oder reale Achsen handelt. Visualisierungsvorlagen für alle Funktionsbausteine können mit der jeweiligen FB-Instanz verknüpft werden und ermöglichen damit einen direkten Zugriff auf alle Eingabe- und Ausgabeparameter des Bausteins. Darüber hinaus lassen sich die Achsen unabhängig von der Applikation in einem speziellen Konfigurationsmodus bedienen und initial einrichten.

Weitaus anspruchsvoller – koordinierte Bewegungen

Richtig anspruchsvoll wird es, wenn die Achsen koordinierte Bewegungen ausführen sollen, etwa um Kurvenscheiben oder CNC-Bewegungen abzuarbeiten.

Solche Bewegungen plant der Anwender in Codesys mittels in die IEC-61131-Software integrierter Editoren. Ein Kurvenscheiben-Editor ermöglicht die Konfiguration der Stützpunkte für Haupt- und Nebenachse sowie der Nockenschaltpunkte in grafischer oder tabellarischer Ansicht. Für die Abarbeitung verschaltet er standardisierte Funktionsbausteine gemäß PLCopen Motion Control Teil 2 aus der entsprechenden Bibliothek. Die Planung von CNC-Verfahrensbewegungen erfolgt in einem integrierten DIN-66025-Editor im G-Code. Parallel dazu wird die Bewegung in einem grafischen 3D-Editor angezeigt. Für eine schnelle Realisierung kann der Anwender die Reihenfolge auch umdrehen: Er skizziert die Bewegung komfortabel im 3D-Editor mit Linien, Kreissegmenten oder Splines und optimiert sie dann im G-Code.

Ist die Bewegung bereits elektronisch beschrieben, so kann G-Code auch aus einer externen Tabelle abgearbeitet werden oder der Anwender importiert DXF-Dateien von CAD-Tools in das Projekt. Default-Werte für Parameter wie zum Beispiel Geschwindigkeit, Beschleunigung und Abbremsung sowie Ursprung können dabei global oder pro CNC-Objekt festgelegt werden. Außerdem stehen Funktionen zur Geometriedaten-Manipulation wie etwa Werkzeugradius-Korrektur oder Verschleifung von Ecken zur Verfügung. Die Abarbeitung der Bewegung wird wiederum in den Sprachen der IEC 61131-3 auf Basis bereitgestellter Funktionsbausteine programmiert. Wie bei den Einachs-Bewegungen werden die erforderlichen Achsen von Bausteinen verarbeitet, die die zyklisch be-



Achsgruppen-Editor mit Parametrierung einer Tripod-Kinematik: Der Editor beschreibt die Roboterparameter und ermöglicht deren Eingabe.

rechneten Sollwerte an die Datenstruktur der Achse übergeben. In einer Funktionsbibliothek bietet die SPS-Programmierungsumgebung Transformationsbausteine für die wichtigsten Anwendungsfälle von koordinierten Bewegungen.

Im Teil 4 der PLCopen-Motion-Control-Standardisierung sind Funktionsbausteine zur Abarbeitung von Achsgruppen für komplexe Kinematiken definiert – darunter MC_GroupEnable, MC_GroupHalt, MC_MoveDirectRelative, MC_MoveDirectAbsolute, MC_TrackConveyorBelt oder MC_TrackRotaryTable. Mit diesen Bausteinen ist die Projektierung der Abarbeitungsfunktion noch einfacher, da nicht einzelne Achsen, sondern eben komplette Achsgruppen inklusive Transformationsinformationen übergeben werden. In ‚Codesys SoftMotion CNC+Robotics‘, dem neuen Codesys-Add-on, stehen diese jetzt ebenfalls in Form einer Bausteinbibliothek zur Verfügung.

Baustein-Bibliothek für Robotik-Funktionen

Die Konfiguration der Achsgruppen erfolgt dabei in einem neuen Editor, der zahlreiche Komfortfunktionen mitbringt: So kann der Anwender die erforderliche Robotertransformation aus einer Liste an Roboter-Kinematiken auswählen, unter anderem für diverse Portal- (2/3/5 Achsen), Bipod-/Tripod- oder Scara-Roboter. Der Editor erläutert die Roboterparameter teilweise anhand von Grafiken, wie etwa Arm- oder Werkzeuglängen, die Position oder den

maximalen Bewegungsradius einzelner Achsen und ermöglicht die Eingabe der realen Werte. Anhand einer dokumentierten Schnittstelle lassen sich sogar anwenderspezifische Kinematiken in den Sprachen der IEC 61131-3 erstellen. Daneben stehen Werkzeug-Kinematiken zur Verfügung, die mit einigen der genannten Kinematiken kombinierbar sind, um Werkzeuge mit einer oder mehreren Drehachsen und Werkzeug-Offsets festzulegen. Der Anwender verknüpft in diesem Editor auch die Achsen der Achsgruppe mit den im Projekt konfigurierten Antriebsachsen. Die integrierte Robotik-Funktion bringt damit genau den geplanten Effekt: Mit der Achsgruppendefinition und wenigen Baustein-Aufrufen kann die gesamte Projektierung eines Roboters erfolgen – unabhängig von der Kinematik, die ja in der Achsgruppe selbst festgelegt wird.

Auf den Punkt gebracht: Die in eine IEC-61131-3-Programmierungsumgebung integrierte Robotik-Funktion vereinfacht die Projektierung von komplexen Achsgruppen noch einmal ganz erheblich. Anwender profitieren von Einsparungen bei der Hardware sowie vom vereinfachten Engineering in einer einzigen Oberfläche. gh



Roland Wagner
leitet das Produktmarketing bei 3S-Smart Software Solutions.



KEEP CALM AND



TAKE A SYSTEM

Make your life easier!

Persistent CompactPCI Serial design

Reliable industrial solutions

Highly adaptable modular system, suitable for a wide range of applications, e.g.:

- measurement systems
- multi node wireless networks
- radar surveillance systems
- image processing
- 2oo3 interlocking systems

