



Automation Studio 4:

Im Team schneller entwickeln

Bewährte Funktionen nutzen

Softwareentwickler müssen trotz zunehmender Produktkomplexität innovative Softwarelösungen mit hoher Qualität schaffen und dabei die meist engen Budget- und Terminvorgaben einhalten. Dies gelingt nur, wenn Entwicklungsaufgaben parallelisiert und bereits bewährte Funktionen wiederverwendet werden. Mit Automation Studio 4 stellt B&R Entwicklern von Automatisierungslösungen ein zentrales Werkzeug zur Verfügung, das mit Applikationsmodulen die parallele Entwicklung der Software bestmöglich unterstützt.

Die Anforderungen des Marktes sind so vielfältig wie die Märkte selbst. Verlangt werden individuell zugeschnittene Maschinen mit großer Komplexität, Flexibilität und Optionsvielfalt. Die-

se sollen in Produktionszellen, -linien oder Gesamtanlagen integrierbar, kurzfristig verfügbar, energieeffizient, ergonomisch und schnell einsetzbar sein. Gefordert wird der Preis und die

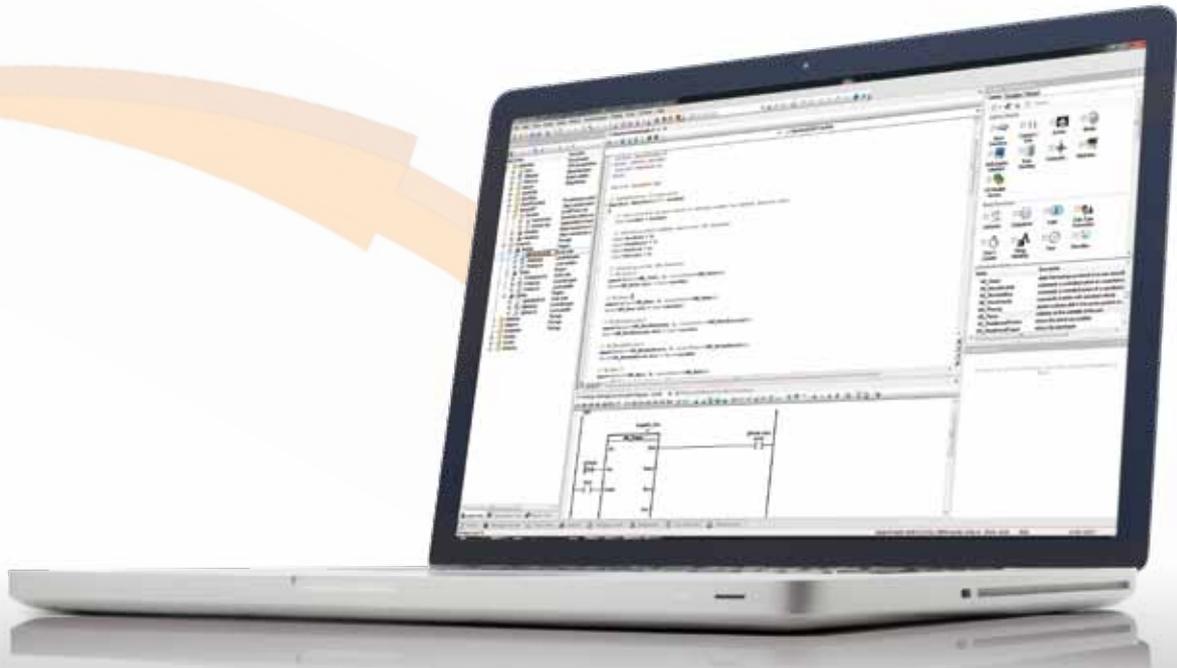
Stabilität eines Großserienproduktes, wobei die Maschinen allen Normen und Dokumentationsvorschriften entsprechen müssen. Zustandsüberwachung, Diagnosetools zur Wartungsunterstützung und vorausschauende, intuitive Benutzerführung sind gefragter denn je. Nicht zuletzt möchten Käufer später durch Nach- oder Umrüsten und eigene Eingriffe auf Bedarfsveränderungen reagieren können.



“Die Möglichkeit, in Automation Studio 4 die Software für komplexe Maschinen und Anlagen aus unabhängig voneinander entwickelten Applikationsmodulen zusammenzustellen, unterstützt Maschinenentwickler aktiv bei der Bewältigung der steigenden Komplexität.

Dr. Hans Egermeier,
Business Manager Automation Software bei B&R

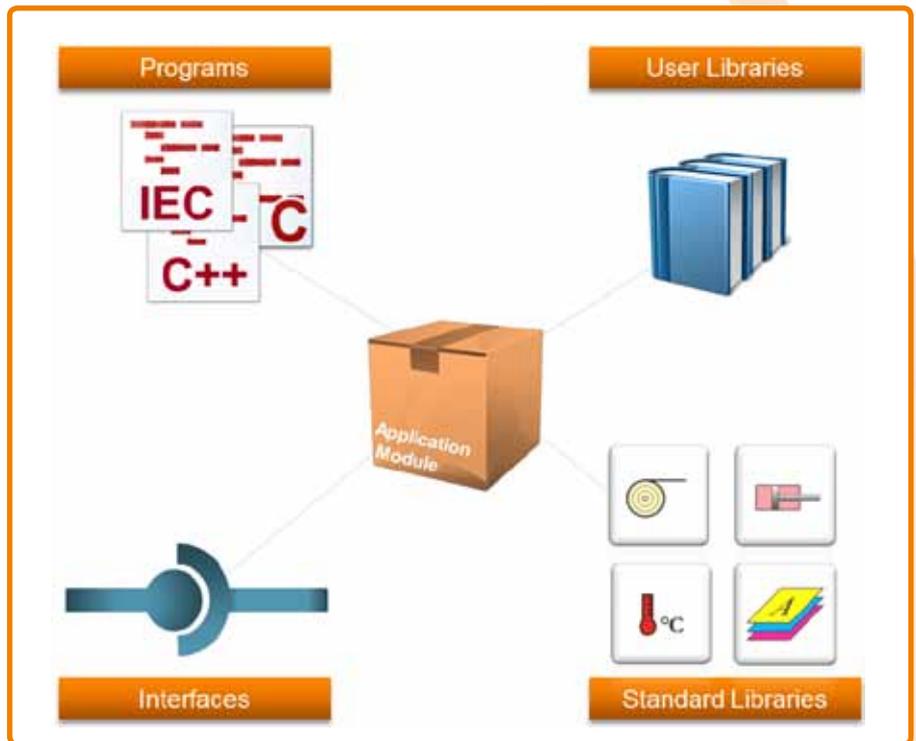




links Unabhängige und wiederverwendbare Module sind ein wesentlicher Bestandteil von Smart Engineering. Der Maschinenbauer profitiert von einer parallelen Modulentwicklung und damit verbundenen verringerten Entwicklungszeiten.

oben Eine wichtige Voraussetzung für die parallele Entwicklung ist ein für alle gültiger Projektrahmen. Innerhalb der Applikationsmodule erfolgt die Programmierung in den gewohnten Programmiersprachen.

rechts Zur Implementierung der Applikationsmodule stehen etablierte Programmiersprachen als auch gewohnte Benutzer- und Standardbibliotheken zur Verfügung. Eine kostspielige Einschulung und Umstellung z. B. auf eine neue Sprache entfällt. Definierte Schnittstellen ermöglichen den späteren Austausch der abgegrenzten Module und erhöhen so die Wartbarkeit der Software.



Produktkomplexität beherrschen

Der Anteil der Software im Maschinen- und Anlagenbau steigt. Funktionalitäten, die früher noch mechanisch, elektrisch oder elektronisch gelöst wurden, werden zunehmend in Software realisiert. Die Erfassung und Auswertung von Bildinformationen sowie die Steuerung und Synchronisierung komplexer, mechanischer Abläufe über Antriebssysteme sind Anforderungen an moderne Maschi-

nen und Anlagen, die es zu erfüllen gilt. Des Weiteren wird die Interaktion der Maschine mit dem Benutzer immer anspruchsvoller. Es werden Regelungen mithilfe komplexer Regelungsalgorithmen verlangt. Mithilfe von Software lassen sich diese Forderungen realisieren. Die stetig steigende Leistungsfähigkeit der Hardware erlaubt eine immer weitreichendere Umsetzung des technisch Möglichen.

Kurvenscheiben und die strikt sequenzielle Programmierung aus der Frühzeit der speicherprogrammierbaren Steuerungen gehören der Vergangenheit an. Mit früheren Methoden sind die Herausforderungen an die Software-Entwicklung nicht mehr zu bewältigen. Automation Studio hilft Maschinen- und Anlagenbauern, sich diesen Herausforderungen zu stellen und zu beherrschen. →



Die modulare Entwicklung der Maschinensoftware kann wahlweise in einem großen Projekt oder aufgeteilt auf mehrere kleinere Projekte erfolgen. Die autark lauffähigen Applikationsmodule können parallel auf einer Steuerung getestet, diagnostiziert und bearbeitet werden – dies spart wertvolle Inbetriebnahmezeit.

Aufgaben sind zu verteilen

In der mechanischen Konstruktion ist es seit Generationen üblich, Aufgaben auf mehrere Köpfe zu verteilen. Dazu müssen zwei Voraussetzungen erfüllt sein: Erstens eine gute interne Projektkoordination inklusive einer sinnvollen Aufteilung der Teilaufgaben und zweitens eine Entwicklungsumgebung, die paralleles Arbeiten an gemeinsamen Projekten bestmöglich unterstützt.

Das Erste ist eine organisatorische Aufgabe. Jeder Softwareentwickler benötigt zunächst einen Rahmen, innerhalb dessen er sich entfalten kann. Dieser entsteht aus dem Konzept der Gesamt-

maschine und enthält eine funktionale Definition der einzelnen Maschinenteile oder -module, die in beliebiger hierarchischer Tiefe untergliedert sein können. Die Schnittstellen zwischen den Modulen werden analog den Anschlüssen bei mechanischen Einheiten festgelegt.

Viele funktionale Teile enthalten mehrere oder alle Automatisierungsaspekte. Entwicklungsleiter müssen die Aufgaben daher äquivalent zu den Einheiten der Gesamtmaschine oder nach automatisierungstechnischen Fachkriterien aufteilen. Es hat sich bewährt, beides zu tun. So ergeben sich Programmfunktionen als wiederverwend-

bare Module für alle Beteiligten und zugleich Teilprogramme für funktionale Einheiten aus einer Hand.

Wiederverwendbarkeit als Voraussetzung

Automation Studio wurde erstmals im Jahr 1992 vorgestellt. Seither werden die Programme für die gesamten B&R-Automatisierungslösungen innerhalb dieser durchgängig einheitlichen Entwicklungsumgebung erstellt. Auch in bisherigen Versionen konnten Entwickler für Funktionen, Abläufe und Regler auf Bibliotheken zurückgreifen und so die zunehmende Komplexität funktionaler Abläufe, das Steuerungsver-



“ Die objektorientierte Programmierung hilft, größere Software-Systeme übersichtlich zu halten und ihre Wartbarkeit zu erhöhen.

Wolfgang Portugaller, Leiter Systemarchitekten bei B&R



ten ganzer Maschinen und Regelungs-algorithmen beherrschbar machen. Häufig verwendete Grundfunktionalitäten sind in diesen Bibliotheken bereits verwendungsfertig enthalten und brauchen nur noch konfiguriert werden. Selbst entwickelte Funktionen können dem Katalog zur weiteren Verwendung hinzugefügt werden. So entsteht analog zur Mechanik ein System von Software-Normteilen.

In der aktuellen Version von Automation Studio kann vollständig objektorientiert in C++ programmiert werden. Die Grundidee dieser Softwareentwicklung besteht darin, die Architektur der Software nach den Grundstrukturen der realen Welt auszurichten. Objektorientierte Software kann nach derselben Gliederungssystematik strukturiert werden. Somit bilden die Bestandteile einer Maschine oder Anlage die Elemente des Software-Modells. Neben den Programmiersprachen der IEC 61131-3 unterstützt Automation Studio auch die Programmierung in Hochsprachen wie C und die modellbasierte Entwicklung mit automatischer Programmableitung aus Simulationssystemen.

Modularisierung auf übergeordneter Ebene

Mit der im Frühjahr 2013 erstmals ausgelieferten vierten Generation der B&R-Entwicklungsumgebung erreichte die parallele Softwareentwicklung im Team ein bislang unerreichtes Niveau. Die bahnbrechende Neuerung ist dabei die Modularisierung auf einer höheren Ebene.

Diese erfolgt durch autonom lauffähige Applikationsmodule, die beinahe beliebig skalierbar sind und daher einzelne Funktionen, wie ganze Maschinenteile oder Teilmaschinen, repräsentieren. Das lässt modulare Maschinenkonzepte leichter abbilden und die Entwicklungsaufgaben auf mehrere Entwickler verteilen, die nicht notwendigerweise im selben Haus sitzen müssen. So können ohne großen Aufwand ganze Maschinenmodule inklusive der zugehörigen Software von Spezialunternehmen zugekauft, externe Automatisierungsdienstleister einbezogen oder gewisse

Teile vom Kunden selbst programmiert werden.

Entwicklungsaufgaben effizient verteilen

Teamarbeit beschleunigt die Softwareentwicklung wesentlich. Die parallele Entwicklung basiert auf dem Projektrahmen. Dieser legt die Konfiguration des Gesamtwerks fest und definiert die anzusprechende Hardware, Task-Klassen und die vereinbarten Schnittstellen. Die hierarchisch aufgeteilten Applikationsmodule können aus einzelnen Funktionsblöcken, ganzen Programmen oder einer beliebigen Mischung bestehen. Deshalb können funktional unterschiedliche Teilprogramme auf die jeweiligen Spezialisten und große Einzelaufgaben auf mehrere Entwickler verteilt werden.

Die Beteiligten haben die Möglichkeit, ihren individuellen Erfahrungsschatz einzubringen. Solange die Schnittstellendefinition eingehalten wird, funktioniert das Gesamtsystem, auch wenn ein Applikationsmodul mit unterschiedlichen Sprachen programmiert wird. So kann bewährte Funktionalität in Form bestehender Softwareteile weiter verwendet werden. Zur Erstellung von Applikationsmodulen stehen etwa für die Ansteuerung von Antriebsachsen vorgefertigte Bibliotheken und Funktionsmodule zur Verfügung.

Um Schnittstellen für den Datenaustausch festzulegen, werden Prozessvariable mit Namen, Semantik und Typ definiert und den logischen Ein- und Ausgängen der jeweiligen Applikationsmodule zugeordnet. Auf Basis dieses Mappings kann die Software entwickelt werden. Dazu benötigen die Entwickler weder genauere Informationen über die interne Funktionsweise anderer Softwareteile, noch müssen diese selbst zur Verfügung stehen. Jedes Applikationsmodul kann, auch wenn angrenzende Softwareteile noch nicht vorliegen, ausführlich getestet werden. So lassen sich viele Fehlerquellen bereits frühzeitig ausschließen und abschließende Tests der Zusammenstellung stark abkürzen. Der Zertifizierungsaufwand verringert sich, wenn Applikationsmodule, die bereits

einmal zertifiziert wurden, wiederverwendet werden.

Einzelne Funktionsteile sukzessive weiterentwickeln

Die einzelnen kompilierten Applikationsmodule können nach und nach in die Zielhardware geladen werden. Dadurch erfolgen Testläufe und die Inbetriebnahme der Gesamtmaschine sukzessive, was die Fehlersuche und -behebung stark erleichtert und beschleunigt. Um die Maschinen laufend zu verbessern und weiterzuentwickeln, lassen sich einzelne Module erneuern, ohne den bewährten Rest des Gesamtsystems anzutasten. So können z. B. funktionale Einheiten wie Handling-Einrichtungen ausgetauscht werden. Dies kann als komplettes mechatronisches Gebilde mit maschinenbaulicher und elektronischer Komponente sowie einem nachladbaren Applikationsmodul für die Steuerung der Hauptmaschine erfolgen. „Tendenziell sinkt dadurch der Termindruck und die Softwarequalität steigt“, ist Wolfgang Portugaller, Leiter Systemarchitektur bei B&R, überzeugt. „Bereits fertig entwickelte Applikationsmodule können in unterschiedlichen Maschinenprojekten wieder verwendet werden. Die eingesparte Zeit kann wiederum in umfangreichere Tests investiert werden und verkürzt die Inbetriebnahme.“ Eine gute Projektplanung und Koordination der Entwicklungsanstrengungen ist nach wie vor unverzichtbar. Allerdings müssen sich Entwickler nur mehr bei nötigen oder vorteilhaften Veränderungen der Definition und bei Zusammenstellungstests abstimmen. Durch die Integration einer Versionsverwaltung können Kollaborationswerkzeuge wie der Microsoft Team Foundation Server vollständig mit ihren Workflow-Mechanismen angebunden werden. Das reduziert unproduktive Abstimmungssitzungen auf ein Minimum und steigert nachhaltig die Software-Entwicklungseffizienz.

**Bernecker + Rainer
Industrie-Elektronik GmbH**

B&R Straße 1, A-5142 Eggelsberg
Tel. +43 7748-6586-0
www.br-automation.com