

In LASAL längst Stand der Technik:

Funktionalität vererben



Engineering-Zeit ist Geld wie wir alle wissen. Dementsprechend wichtig sind wiederverwendbare modulare Bausteine für den modernen Maschinenbau. In der Mechanik ist es längst Usus, dass erprobte Konstruktionen immer wieder zum Einsatz kommen. Eine objektorientierte Programmierung macht dies auch in der Software möglich. Die Engineering Suite LASAL von SIGMATEK ist das beste Beispiel dafür: Sie bietet die perfekten Rahmenbedingungen für eine Modularisierung von Maschinenfunktionen in der Software und für das »Vererben« bewährter Funktionalitäten.

Modularität und Wiederverwendbarkeit zählen zu den wichtigsten Grundsteinen für eine effiziente und flexible Applikations-Entwicklung. Denn diese beiden Eigenschaften ermöglichen es, kundenspezifische Lösungen schnell bzw. in relativ kurzer Zeit umzusetzen. Einfach anpassbare Baukastensysteme stehen deshalb sowohl bei Hardware- als auch bei Software-Themen hoch im Kurs. Eine Tendenz, die SIGMATEK bereits im Jahr 2000 herannahen sah: Und so zählt das Unternehmen seit damals zu den ab-

soluten Vorreitern in Sachen objektorientierter Programmierung. „Der größte Vorteil einer objektorientierten Programmierung ist eine durchgängige Modularität von der untersten Ebene der einzelnen Funktionen bis nach oben zum Gesamtprojekt. Das bedeutet: Maschinenfunktionen können einzeln oder in Gruppen entwickelt, getestet, hinzugefügt, ausgetauscht oder bei Nichtverwendung einer Option ausgeblendet werden“, beschreibt Bernhard Gangl Abteilungsleiter Software bei SIGMATEK.



„Der größte Vorteil einer objektorientierten Programmierung ist eine durchgängige Modularität von der untersten Ebene der einzelnen Funktionen bis nach oben zum Gesamtprojekt“, erklärt Bernhard Gangl Abteilungsleiter Software bei SIGMATEK.

Die Engineering Suite LASAL vereint alle Automatisierungsaufgaben auf einer Software-Plattform, sorgt somit für ein Maximum an Übersicht und für ein einfacheres Handling modularer Maschinen- und Anlagenkonzepte. Ein Faktor der gerade im Zeitalter eines »Internet der Dinge« immer wichtiger wird. Die Kommunikation mit übergeordneten Systemen kann beispielsweise über eine OPC UA- oder TCP/IP-Schnittstelle in der Steuerung realisiert werden.

Ein Datenfluss-Manager

Automatisierungssysteme von SIGMATEK, die Ablauf-, Bewegungssteuerung, Visualisierung und sicherheitsgerichtete Steuerungstechnik in Komplettlösungen vereinen, weisen seit Jahren eine dezentrale Architektur auf: Maschinen- oder Anlagenfunktionen sind in logische, mechatronische Einheiten gegliedert – ausgestattet mit dezentralen Kompakt-CPU's. Jede CPU-Funktionseinheit erledigt die ihr zugedachte Aufgabe. Um diese intelligente Modularität auch in der Software abbilden zu können, hat SIGMATEK den »Machine Manager« entwickelt. Dieses Tool ist Teil der Entwicklungsumgebung LASAL und liegt als Software-Schicht über der eigentlichen Maschinenprogrammierung. Es weist die modularen Softwareprojekte den vorhandenen Hardware-Steuerungseinheiten zu. Maschinensoftware besteht zumeist aus Teilprojekten, die oftmals in weitere Untergruppen wie Ablauf, Visualisierung und Safety aufgegliedert sind. Mit Hilfe des Maschine Managers wird aus diesen komplexen Zusammenhängen eine stimmige »Solution«. Dazu kanalisiert und übersetzt er die Kommunikation zwischen den einzelnen Maschinenteilen. Projekte, die Daten zur Verfügung stellen, werden mit den Projekten verbunden, die diese Daten verwenden sollen bzw. dürfen. Zur Anzeige des Verbindungsstatus kann ein vorgefertigter Funktionsbaustein eingesetzt werden. Da sich auch der Datenaustausch mit externen Anlagenteilen und übergeordneter Software mit dem Machine Manager sehr übersichtlich gestaltet, erleichtert das nicht nur die Inbetriebnahme, Handhabung sowie Wartung komplexer Maschinen und Anlagen, sondern auch die Umsetzung adaptiver Produktionsstrategien.

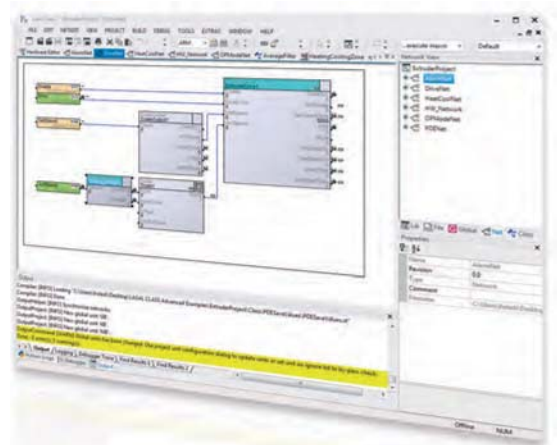


Der LASAL Machine Manager hat den Datenfluss bei Mehr-CPU-Konzepten fest im Griff.

Modulares Baukastenprinzip

Beim objektorientierten Programmieren werden die verschiedenen Teile einer Maschine oder Anlage in Form von »Objekten« repräsentiert. Hinter einem Objekt steht jeweils eine »Klasse«. Sie ist der Bauplan des Objekts und enthält den Programmcode und die dazugehörigen Datenelemente. Jede Klasse kann eine bestimmte Aufgabe übernehmen, wie beispielsweise die Messung und Auswertung einer Temperatur, die Regelung eines Ventils oder die Ansteuerung eines Förderbandes. Die vom Programmierer definierten Klassen werden in übersichtlichen Bibliotheken abgelegt. LASAL ermöglicht die Modularisierung von Maschinenfunktionen in der Software und somit ein mechatronisches Engineering. Die einzelnen Funktionen (= Objekte) lassen sich in der Software im Baukastensystem zusammenstellen und einfach »verdrahten«. Denn die anwenderfreundliche Devise lautet: Weniger Programmieren – einfach Konfigurieren. Demzufolge stehen in den umfangreichen LASAL-Bibliotheken fix fertige Applikationsmodule für einen raschen Einsatz bereit. Diese getesteten Funktionsbausteine decken ein breites Maschinenspektrum ab und umfassen komplexe Filter- und Regelungsalgorithmen, verschiedenste Bewegungsmodule und Roboter-Kinematiken, Kommunikationsprotokolle wie Modbus TCP und vieles andere mehr. Zusätzlich stehen Applikationstemplats bereit, die neben dem fer-

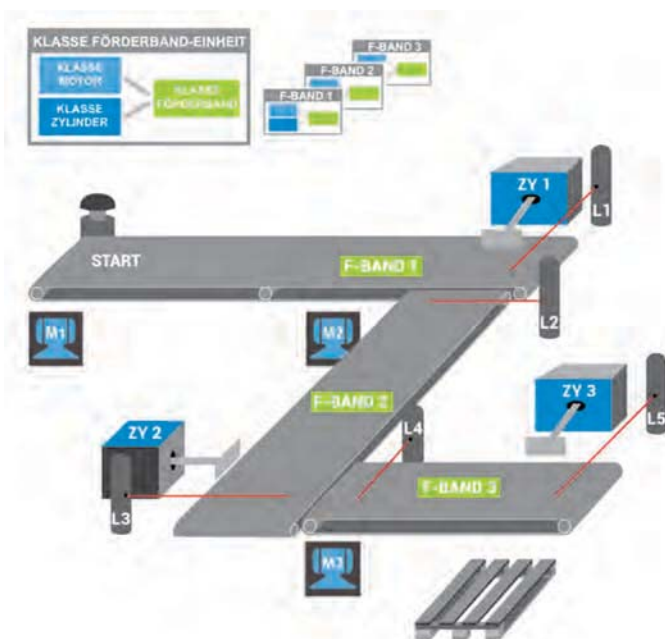
tigen Projekt der Ablaufsteuerung gleich auch die passende Visualisierung beinhalten. Beispiele dafür sind u. a. die Anmeldefunktion und Zugriffsberechtigungen, das Ereignisjournal oder das Oszilloskop zur Aufzeichnung mehrerer Kanäle. Alle diese Funktionalitäten sind ready-to-use und können modular im Baukastenprinzip eingesetzt werden.



Durch die grafische Darstellung wird eine übersichtliche Softwarestruktur möglich. Dargestellt werden die Funktionalitäten, die Beziehungen zwischen den Objekten, der Datenverkehr und die Schnittstellen.

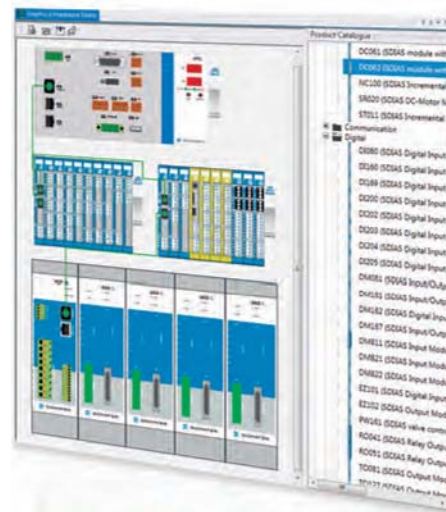
Grafische Darstellung: Alles im Überblick

Für Bedienerergonomie bei der objektorientierten Programmierung mit LASAL sorgt die grafische Darstellung. Per Drag & Drop wird eine Klasse aus dem Projektbaum in ein »Objektnetzwerk« eingebunden und somit zu einem instanziierten, realen Objekt. Die Objekte müssen lediglich miteinander verbunden werden und schon entsteht eine Applikation. Durch die grafische Darstellung erhält der Anwender eine Gesamtübersicht über sein Projekt: Er sieht Funktionalitäten, die Beziehungen zwischen den Objekten und die Schnittstellen. Komplexe Zusammenhänge sind so viel transparenter und einfacher zu kontrollieren bzw. zu ändern.



Beziehungen und Eigenschaften einfach »vererben«

Mit den Techniken »Vererbung«, »Ableitung« und »Aggregation« ist es möglich, neue Ausprägungen von Maschinenteilen mit minimalem Programmieraufwand umzusetzen. So kann zum Beispiel mittels Vererbung eine Objektklasse zuerst dupliziert und dann verfeinert bzw. spezialisiert werden. Eine abgeleitete Klasse erbt die Eigenschaften der Basisklasse, kann aber verändert oder um zusätzliche Informationen, wie Attribute oder Operationen, erweitert werden. Durch Aggregation können mehrere einzelne Klassen zu einer komplexen Klasse zusammengefasst werden. Einmal getestete Klassen lassen sich so übersichtlich in Bibliotheken ablegen und nach dem Baukastenprinzip zu komplexen Programmstrukturen zusammenfügen. „Bei einer Transportstrecke für Stückgut mit drei aneinandergereihten Förderbändern beispielsweise wird jedes Förderband von einem Motor angetrieben, verfügt jedes über spezifische Start- und Stopp-Bedingungen sowie über einen Zylinder am Ende, der das Stückgut weiterschiebt. Demnach werden drei Klassen modelliert: Motor- und Zylinderansteuerung sowie Start-/Stopp-Bedingungen für das Förderband. Durch Aggregation, sprich einer Gruppierung, lässt sich aus diesem Klassenverbund dann eine komplexe Klasse namens »Förderband-Einheit« erstellen“, erklärt Bernhard Gangl.



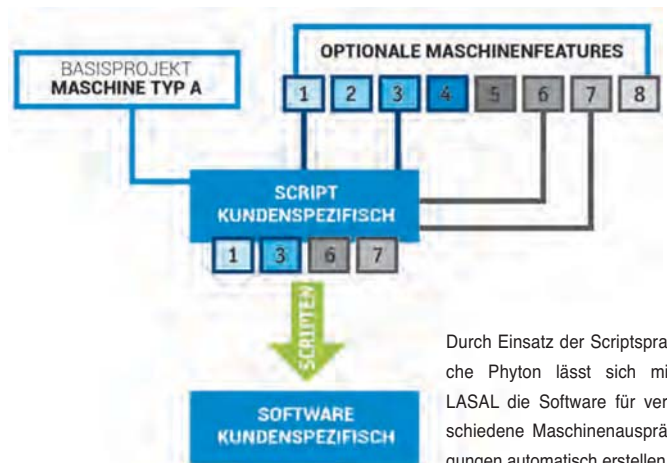
Der grafische Hardware-Editor sorgt für große Übersichtlichkeit bei der Projektierung. Aus dem integrierten Produktkatalog lassen sich die Komponenten einfach einfügen.

Detailgetreue Abbildung der Realität

Mit dem grafischen Hardware Editor bietet sich dem Anwender eine neue Perspektive in der Automatisierungssoftware LASAL: Er kann sein Projekt bildlich auslegen und mögliche Varianten realitätsgetreu durchspielen. Aus einem Hardware-Katalog können alle zur Verfügung stehenden Produkte in die Arbeitsfläche gezogen werden – benutzerfreundlich per Drag & Drop. Wie in der Realität lassen sich die Komponenten aneinanderreihen und miteinander verbinden. Auf nicht erlaubte Verbindungen wird per Tooltip bzw. mit einem veränderten Mauszeiger hingewiesen. Um rückseitige Anschlüsse sichtbar zu machen, sind die Produkte drehbar. Interaktive Schnittstellen ermöglichen ein direktes Hinzufügen eines neuen Hardware-Elements – auch ohne Produkt-Katalog. Das ermöglicht ein einfaches und effizientes Projektieren, Parametrieren und Diagnostizieren der an der Steuerung angeschlossenen Hardware-Elemente wie beispielsweise I/Os, Schnittstellen oder Drives. Neben der Hardware lässt sich mit dem grafischen Editor auch die Topologie der Busstrukturen gut darstellen. Dieses komfortable Tool vereinfacht und verkürzt die Projektierungszeiten von Applikationen enorm.

Programme automatisch erstellen

Basierend auf den gekapselten Software-Bausteinen der objektorientierten Programmierung lassen sich durch Scripting aus einem Basisprojekt auch sehr schnell unterschiedliche Maschinenausprägungen anfertigen.



Durch Einsatz der Scriptsprache Python lässt sich mit LASAL die Software für verschiedene Maschinenausprägungen automatisch erstellen.

„Wenn ein Kunde beispielsweise den Basistyp A einer Maschine mit der Option 1, 3, 6 und 7 bestellt, kann die Applikationssoftware für diese spezifische Maschine sogar vollautomatisch erstellt werden, ohne dass der Softwareingenieur dazu auch nur eine einzige Programmzeile manuell ändern muss. Dazu wird auf die Bibliothek mit vorgefertigten Klassen zugegriffen, die alle verschiedenen Module enthält. Da heißt es dann nur noch die für Typ A spezifischen Module auswählen, die gewünschten Sonderfunktionalitäten der Optionen hinzufügen und auf Knopfdruck das passende Applikationspro-

gramm erzeugen“, streicht der Abteilungsleiter Software bei SIGMATEK die Userfreundlichkeit von LASAL hervor.

Anwenderfreundlich bis ins Detail

Im Ansatz von Industrie 4.0 und Smart Factories werden kleinere, mechatronische Einheiten mit eigener Verarbeitungsentelligenz zu einem flexiblen Ganzen vernetzt. Die grafische Darstellung der einzelnen Software-Bausteine in LASAL kapselt nun die Komplexität solch eines Projekts und ermöglicht eine übersichtliche Strukturierung der Software. Denn Anwender-

freundlichkeit stand bei SIGMATEK schon immer im Fokus. „Die User sollen die Vorteile der objektorientierten Programmierung wie Modularität und einfache Wiederverwendbarkeit nutzen können, ohne dass sie dabei mit der dahinterliegenden durchaus komplexen Syntax in Berührung kommen. Der Code liest sich demnach wie ein gewöhnlicher Structured Text, das hilft Zeit und Nerven sparen“, weiß Software-Abteilungsleiter Bernhard Gangl aus eigener Erfahrung.

www.sigmatek-automation.com

Mehr-CPU-Lösungen im Vormarsch

Mit konsequenter Modularisierung ist einiges an (Entwicklungs-)Zeit und Geld zu sparen. Das wissen auch die Maschinenbauer. Lässt sich doch mittels eines ausgeklügelten Baukastensystems weitaus flexibler und vor allem schneller auf sich verändernde Marktanforderungen reagieren. Inwiefern die Engineering-Plattform LASAL zu mehr Flexibilität verhelfen kann, verrät Franz Aschl von SIGMATEK in einem Kurzinterview.

Herr Aschl, Sie zeichnen ja bei Sigmatek für das Innovationsmanagement verantwortlich, deshalb meine Frage an Sie: Wonach verlangt der Markt?

Franz Aschl: Nach Lösungen, die bei der Realisierung von Maschinenkonzepten bestmöglich unterstützen. Unser objektorientiertes Engineering-Tool LASAL beispielsweise verbindet modernste Programmertechnologien mit hoher Modularität und Effizienz. Alle Funktionen zur Lösung von Automatisierungsaufgaben wie Ablaufsteuerung, Visualisierung, Motion Control, Safety, Service und Fernwartung wurden auf einer Plattform gebündelt und sind komfortabel nach IEC 61131-3 realisierbar. Die Methoden der objektorientierten Programmierung stehen als durchgängige Erweiterung der vertrauten Sprache zur Verfügung. Der eigentliche Programmcode eines Objekts wird in strukturiertem Text, Kontaktplan, grafischer Ablaufsprache, Interpreter oder ANSI-C erstellt.

Warum setzt SIGMATEK seit vielen Jahren konsequent auf einen objektorientierten Lösungsansatz?

Aschl: LASAL arbeitet mit gekapselten Objekten, die Maschinen- oder Anlagenfunktionen repräsentieren. Mit solchen vorgefertigten Software-Funktionsbausteinen und Templates, die einfach übernommen werden können, reduzieren sich Programmieraufwand und Fehler.

Gleichzeitig erhöht sich die Wiederverwendbarkeit des Codes. Zudem sorgt die Kapselung der Objekte (Maschinenfunktionen) für die nötige Flexibilität in einem intelligenten Maschinenverbund.

Orten Sie derzeit einen Trend hin zu einer Mehr-CPU-Strategie?

Aschl: Ja. Die Komplexität der Anwendungen steigt permanent. Und bei zentralen Ein-CPU-Lösungen, wie sie vielfach im Einsatz sind, besteht die Gefahr, dass diese im Laufe des Lebenszyklus einer Maschine irgendwann überlastet werden. Es wird also zu einer Verteilung der Aufgaben kommen: Kleinere Einheiten mit eigener Verarbeitungsentelligenz für die Erfüllung einer ganz bestimmten Funktion werden in einem gut vernetzten Gesamtsystem zusammenarbeiten. Mit mehr-CPU-Lösungen lässt sich auch weitaus flexibler auf sich verändernde Produktionserfordernisse reagieren. Somit ist eine Mehr-CPU-Strategie im Grunde genommen nicht anderes als die konsequente Übertragung der Modularisierung, die bereits vor geraumer Zeit im Maschinenbau Einzug gehalten hat, auf die Entwicklung der Automatisierungssoftware.

Heißt das, dass sich Maschinen bereits heute für morgen automatisieren lassen?

Aschl: Ja, mit einer modularen, objektorientierten und offenen Herangehensweise auf



Interview

jeden Fall. Wir laden deshalb am 21. September zu einem Innovationspraxis-Tag »Maschinen für die Zukunft automatisieren« in die Brauerei Schloss Eggenberg – gerne auch die AUTlook-Leser. Experten aus verschiedenen Disziplinen zeigen dort, dass es bereits heute Tools und Lösungen gibt, die die Anforderungen von morgen mitberücksichtigen. Als Beispiele seien der MagicShoe als drahtlose Maschinenbedienungsalternative zu einem kabelgebundenen Fußtaster oder drahtlose Safety genannt. Wir wollen den 21. September vor allem auch dazu nutzen, um die Veranstaltungsteilnehmer zu einem gemeinsamen Querdenken anzuregen und ihren Mut zu wahren Innovationen zu wecken.