



Bild: Sybera GmbH

Anpassbar mittels PC-Technik

Interview zu einer PC-basierenden Technik, die sich durch Flexibilität und Anpassbarkeit auszeichnet

Steuerungssysteme von Maschinen sind aufgrund der Vernetzung und der Vielzahl unterschiedlicher Komponenten und Funktionen höchst komplex. Dies wirkt sich sowohl auf die Dauer der Inbetriebnahme als auch auf Servicezeiten aus. Sybera bietet mit der PC-basierenden Lösung X-GO Flex eine interessante Alternative zu Hardware-basierenden Controllersystemen an. Wir haben uns mit Sybera-Geschäftsführer Jürgen Rall über Technik, Anwendungsbereiche und Nutzen unterhalten.

SPS Ihr Unternehmen Sybera bietet hardwarenahe Software an. Was verstehen Sie darunter?

Rall: Hardware-nahe Software ist heute sehr differenziert zu sehen. Ich kann es wahrscheinlich am besten im Bereich des Windows-Betriebssystems erklären, weil das meinem Gebiet am nächsten kommt. Hardware-nahe Software ist der Bereich, wo Hardware und Software miteinander verschmelzen, also die Ansteuerung der Hardware, z. B. eines Ethernet-Adapters. Der Übergang von einem Programm hin zu einem Ethernet-basierenden Feldbus umfasst dabei mehrere Ebenen – von der Applikationsebene über die Protokollebene bis hin zur Transportebene (Ethernet). Wenn man das jetzt für PC-basierenden Feldbus-Controller sieht: Dort ist es so, dass wir typischerweise im Kernel agieren und das abzuarbeitende SPS-Programm bis hin zur Ansteuerung des Ethernet-

Adapters letztendlich in einer Art Ablaufsteuerung realisiert werden muss.

bauen. Angefangen mit einem Echtzeit-System, das die Basis für alle übergeord-

„Hardwarenahe Software ist der Bereich, wo Hardware und Software miteinander verschmelzen.“

Ganz im Unterschied zur applikationsnahen Software-Entwicklung, wo wir ein Thread-basiertes Umfeld haben. Zum Beispiel einen Sleep-Modus, also einen Wartemodus einzuführen, das geht typischerweise in der Ablaufsteuerung auf der Kernel-Ebene so nicht mehr.

SPS Welche Zielgruppe sprechen Sie mit Ihren Produkten an?

Rall: Die Firma Sybera bietet eine ganze Reihe von verschiedenen Systemen, z.B. Bibliotheken, an, die aufeinander auf-

neten Ebenen liefert. Beispielsweise die Transport-Bibliothek für Ethernet oder die Protokoll-Bibliothek für EtherCAT. Wir sprechen damit viele Zielgruppen an, sei es die Automobil-Industrie oder die Messtechnik oder irgendwelche gezielten Einsatzgebiete für Echtzeit. Wenn wir die verschiedenen Echtzeit-Anforderungen des Kundenumfelds betrachten, ist das sehr umfangreich. Dieses Umfeld des Determinismus ist immer dann gesucht, wenn Abläufe zeitbestimmt programmiert werden müssen.

SPS Können Sie uns ein konkretes Beispiel nennen?

Rall: Beispiel Messtechnik – wir haben ein Messtechnik-System, das bestimmte Instanzen aufnimmt, also beispielsweise Größen von Körpern. Diese Größen müssen nun in einem gewissen Zeitraster abgetastet werden, über einen Tastkopf, der mit einem Motor gesteuert wird. Man kann sich schnell und leicht vorstellen, dass natürlich die Äquidistanz dieser Abtastung sehr wichtig ist. Würden wir das unter einem nicht-deterministischen Betriebssystem realisieren wollen, dann wären sehr große Sprünge in der Abtastung. Da ist nun typischerweise solch ein Echtzeit-System gefordert. Speziell in diesem Beispiel ist der Kunde ein Unternehmer, der Messautomaten für die Automobilindustrie in Serie baut.

SPS Ich möchte kurz auf die Software X-Go Flex eingehen – dort haben Sie den Anspruch, dass sie für jede Anwendung flexibel und adaptierbar, also anpassungsfähig

ist. Jetzt stellt sich für mich die Frage: Muss nicht jedes Steuerungssystem anpassungsfähig sein? Das ist der Charakter einer SPS – dass die Software die Hardware auf die Applikation anpassbar macht.

ferenz-System besteht typischerweise aus einer entsprechenden Hardware, dem Controller und der dazugehörigen Software. Interessanterweise ist bei fast allen Software-Systemen, die wir heute auf dem Markt haben, die Einarbei-

„Interessanterweise ist bei fast allen Software-Systemen, die wir heute auf dem Markt haben, die Einarbeitungszeit, damit man überhaupt erstmals nur ein einzelnes Byte auslesen kann, relativ hoch.“

Rall: Dazu muss ich vielleicht ein bisschen aus meiner eigenen Erfahrung ausholen. Es ist ja immer so, wenn man ein neues Protokoll-System zu realisieren versucht, muss man sich zwangsläufig mit einem Vergleichssystem beschäftigen. Beispiel: Wir haben eine Profinet-Steuerung und bevor wir einen eigenen Profinet-Stack auf den Markt bringen können, müssen wir natürlich ein Referenz-System aufbauen. Ein Re-

tungszeit, damit man überhaupt erstmals nur ein einzelnes Byte auslesen kann, relativ hoch. Das wird natürlich von jedem einzelnen Hersteller bestritten werden. Aber es ist meiner Erfahrung nach tatsächlich so und gar nicht spezifisch auf ein System bezogen. Wir beschäftigen uns derzeit z. B. mit Ethernet/IP und benötigten mit der Referenz-Hardware eine Woche Einarbeitungszeit, bis wir das System soweit am Lau-

„Der Anspruch, den wir bei der Entwicklung hatten, war sehr hoch: Wir wollten ein System schaffen, das einerseits möglichst einfach zu bedienen ist ... und das andererseits sehr flexibel ist, also den gesamten Funktionsumfang bietet.“

fen hatten, dass wir Daten austauschen konnten. Das ist natürlich für ein System, mit dem man Kleinsteuerungen realisiert oder auch für den Service-Betrieb kontraproduktiv. X-Go-Flex bietet hier einen deutlichen Vorteil. Der Anspruch, den wir bei der Entwicklung hatten, war sehr hoch: Wir wollten ein System schaffen, das einerseits möglichst einfach zu bedienen ist, sich also mit wenigen Schritten an unterschiedliche Umgebungen anpassen kann, und das andererseits sehr flexibel ist, also den gesamten Funktionsumfang bietet, der in gewissen Automations-Bereichen erforderlich ist. In gewisser Weise war das die Quadratur des Kreises. Unsere Lösung war eine mehrstufige Vorgehensweise: 1. Stufe: Schnell, durch einfache Parametrierung den Datenaustausch ermöglichen; 2. Stufe: Programmierung mit einem gewohnten Umfeld; 3. Stufe: Erweiterbarkeit des Systems. Damit haben wir bei X-Go-Flex das Ziel, adaptiv zu sein, weitgehend erreicht.

SPS **Wie sieht ein typischer Anwendungsfall für X-Go-Flex ganz praktisch aus?**

Rall: Auch hier bietet der Bereich Profinet wieder ein sehr schönes Beispiel. Typischerweise haben wir es hier im Service-Bereich mit sehr vielen Kunden zu tun, die die Aufgabe haben, als Service-Techniker zu überprüfen, ob ein jeweiliges Profinet-Gerät funktioniert. Bisher muss der Service-Techniker hierfür die Steuerung aufwendig parametrieren, sich an das System ankoppeln und dann einen indirekten Vergleich machen - indirekt, weil er ja die SPS dazwischen hat. Dann sieht er das Ergebnis 'Drive funktioniert' und kann sagen, es

ist vermutlich die Hauptsteuerung, die defekt ist. Unser Ansatz basiert dagegen auf einem Standard-PC. Wir haben also keine Hardware dazwischen, auf der wir nochmal aufsetzen und programmieren müssen, sondern wir können uns direkt über den Ethernet-Port, der sowieso am Notebook verfügbar ist, an den Antrieb ankoppeln und dann durch eine sehr einfache Parametrier-Vorgehensweise den Antrieb ansteuern. Die Zwischenschritte, die ja auch Unsicherheiten entstehen lassen können, fallen weg. Der Service-Bereich ist also ein sehr guter Einsatzbereich für unsere X-Go-Flex-Software.

„Die Zwischenschritte, die ja auch Unsicherheiten entstehen lassen können, fallen weg.“

SPS **Wenn Sie ein bisschen in die Zukunft schauen: Wie können Sie sich vorstellen, dass sich das System weiterentwickelt? Ist solch eine Lösung als komplette PC-basierte Steuerungslösung nutzbar?**

Rall: Man muss hier mehrere Faktoren betrachten. Das eine ist die Größe der Anlage, das ist meines Erachtens ganz entscheidend. Handelt es sich zum Beispiel um eine Insellösung der Steuerung, also ein geschlossenes System? Gerade im Mittelstand sind solche Anlagen sehr verbreitet, da geht es nicht immer um die riesige Automobil-Fabrik, die automatisiert werden muss, sondern oft sind es eben solche Insel-Anlagen. Dafür sind Software-Lösungen wie unsere ideal. Das erste Argument hierfür ist wiederum der Zeitfaktor: Wir können davon ausgehen, dass die Programmie-

rung einer Automations-Insel auf herkömmlichen Weg ungefähr die dreifache Zeit in Anspruch nimmt als mit einer solchen flexiblen adaptiven Lösung. Das zweite Argument ist die Vertrauensbasis einer PC-Lösung. Hätten wir uns vor zehn Jahren unterhalten, wäre das Urteil hierzu sicherlich ganz anders ausgefallen, aber mittlerweile haben sich PC-Systeme etabliert. Und sie haben sich deshalb etabliert, weil man heute in der Lage ist, auf einem Multi-Prozessor-Rechner ein oder mehrere Prozessoren exklusiv für die Arbeit eines Controllers bzw. als Simulator einzusetzen. Das bedeutet, dass wir im Prinzip nichts anderes haben als eine SPS. Der große Unterschied liegt darin, dass wir eine direkte Verbindung zu Windows haben, worauf ja auch die Visualisierung und die Programmierung aufsetzen. Wenn Sie mich fragen, ob sich das Ganze letztendlich ausweitet, hängt das sehr stark mit der Akzeptanz und der Vertrauensbasis eines solchen PC-basierten Systems zusammen. Rein

technisch gibt es meines Erachtens heute keinen Grund, warum ein PC-basierender Controller nicht die gleiche Ausfallsicherheit leisten können soll wie ein autarkes Controller-System. Es gibt natürlich gewisse Bindungen an das Hauptbetriebssystem, das ist ganz klar, aber wir sind heute in der Lage, die Mechanismen so zu kontrollieren, dass sie für einen Dauerbetrieb geeignet sind.

SPS **Und das trifft auf Ihre Lösung X-Go Flex genau so zu?**

Rall: Wir behaupten, dass X-Go Flex genau diesen Weg geht. Wir sind hier auch immer bereit, unseren Kunden gegenüber Überzeugungsarbeit zu leisten, das heißt, die Kunden auch schon in der Testphase zu betreuen. Wir sagen: „Schaut euch diese Systeme als Alternative einmal an, wir begleiten Euch auch dabei.“ Es ist sehr wichtig, den Kunden am Anfang an die Hand zu nehmen und ihm damit eine gewisse Sicherheit für die Entscheidung, ein solches System einzusetzen, zu bieten. ■

SPS **Vielen Dank für das Interview.**

www.sybera.de

„Rein technisch gibt es meines Erachtens heute keinen Grund, warum ein PC-basierender Controller nicht die gleiche Ausfallsicherheit leisten können soll wie ein autarkes Controller-System.“