

Roboter-Automatisierung: 3 Tipps für eine zeit- und kosteneffiziente Umsetzung

Warum No-Code- & Low-Code-Tools aus der Robotik nicht mehr wegzudenken sind

Der Einsatz von Robotern lohnt sich für Unternehmen fast immer. Sie können Arbeitskosten senken, die Mitarbeiter entlasten und die Produktion flexibler gestalten – denn die geforderten Losgrößen werden immer kleiner und die Produktionsprozesse immer individueller. Zudem bietet ein Roboter oder Cobot einen weiteren großen Vorteil: Er kann prinzipiell ohne Pausen und Ermüdungserscheinungen arbeiten, wodurch die Produktqualität gesteigert und Ausschuss verringert wird.

Mit modernen Robotersystemen lassen sich mittlerweile nahezu alle Prozesse im Unternehmen automatisieren. Meist übernehmen Roboter einfache, schmutzige, monotone, körperlich anstrengende oder auch gefährliche Tätigkeiten. Mit der richtigen Hard- und vor allem Software lassen sich aber auch sehr komplexe oder besonders anspruchsvolle Aufgaben, so genannte „Advanced Robotics“ Anwendungen, lösen. Beispiele hierfür sind z.B. die Montage flexibler und biegsamer Bauteile wie Kabel, Drähte oder Schläuche oder die kraftgeregelte Oberflächenbearbeitung.

Dem No-Code-/Low-Code-Trend folgend gibt es unterschiedliche Software-Lösungen am Markt, die eine grafische und damit vereinfachte und schnellere Programmierung ermöglichen. Der Vorteil ist, dass keine speziellen Programmierkenntnisse notwendig sind. Das Angebot reicht von herstellerspezifischen Lösungen bis zu unabhängigen Angeboten, mit welchen Roboter verschiedener Hersteller mit ein und derselben Software programmiert werden können. Insbesondere bei letztgenannten empfehlen Experten auf Tools zurückzugreifen, die automatisch den nativen Robotercode für die offizielle Robotersteuerung generieren - und nicht etwa über einen eigenen IPC den Roboterarm selbst kontrollieren.

Im ersten Fall bleiben Anwender im laufenden Betrieb bei Anpassungen oder Optimierungen während der Wartung flexibel und vermeiden einen Lock-in-Effekt, da sie den Roboter weiterhin auch ohne das Verwenden der Software klassisch per Zeilencode programmieren können.

Egal ob mit externen Engineering-Tools oder per Zeilencode gibt es Faktoren auf dem Weg zur roboterbasierten Automatisierung, denen Anwender häufig eine zu geringe Bedeutung und damit Beachtung schenken. Wie sich drei der wichtigsten Stolpersteine vermeiden lassen, wird im Folgenden kurz erläutert.

PUNKT 1: PROGRAMMIERAUFWAND

Der Zeitaufwand für die Programmierung der Anwendung per se wird zwar meist nicht unterschätzt, doch es lauern noch andere Fallstricke in dieser Phase: Etwa nicht beachtete Prozesstoleranzen und -varianzen oder die erhöhte Komplexität, wenn Sensorik eingebunden oder eine SPS-Roboter-Kommunikation hergestellt werden muss.

Dazu kommt, dass die Programmierung einer Anlage oft eine maßgeschneiderte und komplexe Lösung ist, die sich nur schwer adaptieren lässt. Zudem haben Programmierer oft ihren eigenen Stil. Damit können die entstandenen Codezeilen bzw. das Programm für andere Programmierer schwer nachvollziehbar und bearbeitbar sein.

An dieser Stelle unterstützt der Markt den Anwender mit den bereits genannten No-Code-/Low-Code Lösungen. Denn mit den vorgefertigten Funktionsbausteinen zum Beispiel lassen sich Programme übersichtlich und auch für andere nachvollziehbar aufbauen und strukturieren. Auch Prozesstoleranzen und -varianzen können mit der richtigen Software automatisch ausgeglichen, analysiert und optimiert werden. Wenn die entsprechenden Schnittstellen bereits integriert sind, wird der Aufwand beim Einbinden von Sensorik oder SPS-Kommunikation ebenfalls immens reduziert.

PUNKT 2: AUFWAND WÄHREND DES ANLAUFS

Eine Faustregel besagt, dass die Kosten für den Roboter selbst nur rund ein Drittel der Anschaffungskosten der Zelle ausmachen, und die Erfahrung zeigt, dass rund 45 Prozent der typischen Kosten während des Anlaufs anfallen. Und das hängt damit zusammen, dass Anwender häufig den Zeitaufwand für die Inbetriebnahme unterschätzen. Die Anlage wird zwar im Vorfeld durchgängig simuliert und offline programmiert, doch oft zeigen sich bei der Inbetriebnahme Unterschiede zwischen Theorie und Praxis.

So kann der Prozess trotz guter Vorbereitung deutlich länger als geplant dauern und Anpassungen und Änderungen können schnell teuer werden. Hinzu kommen Faktoren, die bei der digitalen Vorarbeit gar nicht ersichtlich waren und nun während der Inbetriebnahme vor Ort gelöst werden müssen. Das macht diese Phase schwer zu kalkulieren. Mit durchgängigen Tools lässt sich der Anlauf kontrolliert und ohne großen Zeitverlust umsetzen, so dass diese Phase eben nicht zum Kostentreiber wird. Wichtig hierbei ist, Simulation, Programmierung, Sensorik und Datenanalyse in einem Software-Paket zusammenzuführen. So wird die Engineering-Kette durchgängig bis zur Instandhaltung, ohne dass der Anwender Abstriche bei der Funktionalität hinnehmen muss.

Dadurch sind Änderungen und Anpassungen schneller, flexibler und einfacher möglich. Durch bestenfalls automatisch erzeugten Robotercode sowie die Möglichkeit, Teach-Punkte auch vom realen Roboter in die Software zurückspielen zu können, integriert sich eine solche Lösung nahtlos und optimal in die bestehenden Inbetriebnahme- und Instandhaltungsprozesse. Dadurch bietet sie auch die größte Flexibilität hinsichtlich Online- und Offline-Programmierung, um für die jeweilige Aufgabe die beste Variante und den einfachsten Weg zu wählen.

PUNKT 3: VERÄNDERUNGEN WÄHREND DER ANLAGENLAUFZEIT

Auch wenn der Roboter läuft, lauert noch immer eine Gefahr, die oft nicht bedacht wird: Über die Anlagenlaufzeit kann es zu zahlreichen Veränderungen der Rahmenbedingungen kommen, die eine Anpassung der Programmierung notwendig machen.

Das können Erschütterungen und Vibrationen ausgelöst beispielsweise von Gabelstaplern oder anderer Maschinen sein, der Verschleiß von Werkzeugen, Ersatzteile, die anders reagieren als die bisherigen Komponenten, Änderungen bei Werkstückchargen und das Platzangebot in der Halle oder der Wechsel des Bedienpersonals.

Auch veränderte Lichtverhältnisse und Temperaturbedingungen oder der Unterschied zwischen kalt gestartetem und warmgelaufenem Roboter können sich auswirken.

Software, die die Programmierung standardisiert und vereinfacht, ermöglicht dem Anwender, einfach, schnell und flexibel auf diese und viele andere Änderungen zu reagieren und notwendige Anpassungen im Programm selbst vorzunehmen. Nutzt er ein Tool, mit dem sich Veränderungen bzw. die daraus folgenden Konsequenzen bei Kräften, Zykluszeiten oder Ausschuss und Qualität auch

noch frühzeitig sichtbar machen, und dadurch leichter analysieren und mögliche Optimierungen ableiten lassen, ist der Anwender bestmöglich vorbereitet.

www.artiminds.com

Autor:

Silke Glasstetter
Head of Marketing bei der ArtiMinds Robotics GmbH

Bildmaterial:



Bild 1: No-Code-/Low-Code-Tools vereinfachen und beschleunigen die Roboterprogrammierung dank ihres Template-basierten Ansatzes; Quelle: ArtiMinds Robotics GmbH

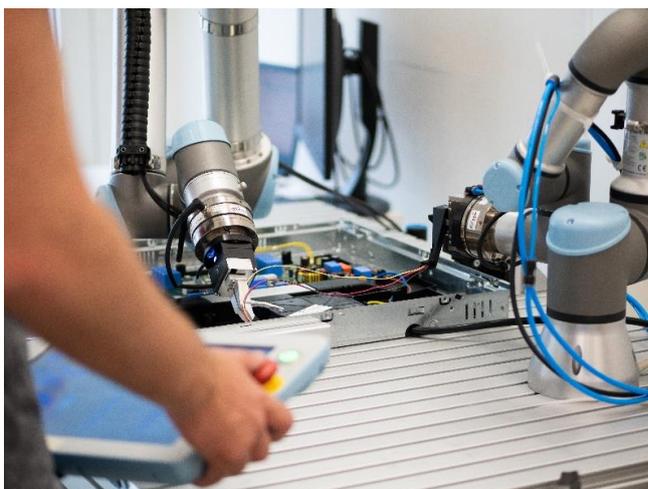


Bild 2: Advanced Robotics Anwendungen wie die Montage biegsamer Kabel stellen eine große Herausforderung für die klassische Roboterprogrammierung dar; Quelle: ArtiMinds Robotics GmbH

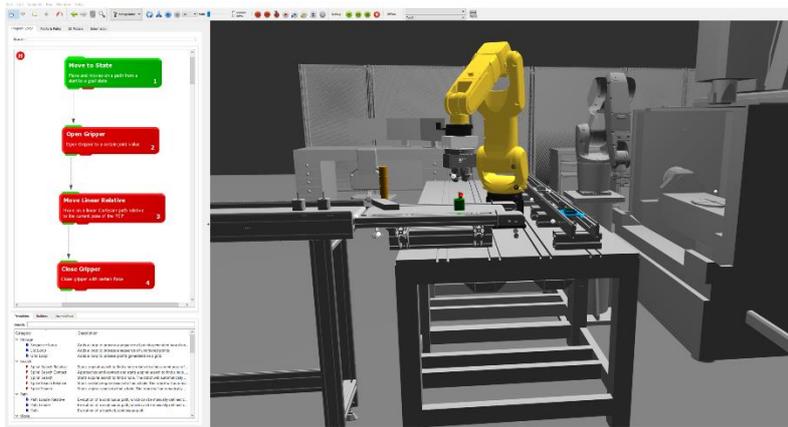


Bild 3: Mittels vorgefertigter Funktionsbausteine lassen sich Programme strukturiert und nachvollziehbar aufbauen. Vor der Inbetriebnahme kann der Prozess in der 3D-Simulationsumgebung möglichst real abgebildet und getestet werden; Quelle: ArtiMinds Robotics GmbH



Bild 4: Wenn Teach-Punkte auch vom realen Roboter in die Programmiersoftware zurückgespielt werden können, integriert sich eine solche Lösung nahtlos in den Inbetriebnahmeprozess und erspart unerwarteten Aufwand; Quelle: Ridvan – stock.adobe.com