

Skillbasierter Robotereinsatz für Industriearbeiten

Studiengang: Master of Science in Engineering
Vertiefung: Mechatronics and Automation
Betreuer: Melchior Borer
Experte: Simon Stucki

Roboter sind in der Industrie unverzichtbar, doch hohe Beschaffungs- und Inbetriebnahmekosten sowie komplexe Programmierung erschweren ihre Integration. Monotone Aufgaben werden oft weiterhin manuell ausgeführt, da die Umstellung auf einen automatisierten Prozess mit hohem Aufwand verbunden ist. Diese Thesis untersucht, wie Roboter durch eine standardisierte Softwarestruktur einfacher und schneller programmiert werden können. Der Fokus liegt auf einem skill-basierten Ansatz.

Motivation

Um die Wettbewerbsfähigkeit der Schweiz auf dem internationalen Markt zu sichern, gewinnen automatisierte Prozesse immer mehr an Bedeutung. Ein zentraler Bestandteil dieser Entwicklung sind Themen wie Flexibilität, Modularität und die einfache Einrichtung sowie Bedienung von Industrieanlagen. Diese Anforderungen betreffen nicht nur die Hardware, sondern stellen auch spezifische Ansprüche an die Software. Basierend auf TwinCAT wurde eine Software-Struktur konzipiert, entwickelt und getestet, die eine einfachere und schnellere Programmierung von Industrieanlagen ermöglicht. Im Mittelpunkt steht dabei ein skill-basierter Ansatz, bei dem die Fähigkeiten der Anlagenkomponenten in übersichtliche und klar definierte Skills unterteilt werden.

Methode

Die Software-Struktur basiert auf drei Aspekten:

Aufteilung von Prozess und Anlage

Die Software gliedert sich in ein Prozessmodell und ein Anlagenmodell. Das Prozessmodell steuert die Abläufe und definiert die Prozessparameter, während das Anlagenmodell die Funktionalität der Systemkomponenten abbildet. Beide Modelle sind unabhängig voneinander, arbeiten jedoch über definierte Schnittstellen zusammen.

Die SPS als zentrales Element

Die Steuerung sämtlicher Anlagenkomponenten erfolgt über die SPS, die damit das zentrale Element der Anlage darstellt. Jede Komponente wird im Anlagenmodell durch eine spezifische Objektklasse repräsentiert. Für die Entwicklung dieser Objektklassen wurde eine standardisierte Struktur implementiert, welche Konsistenz und Effizienz gewährleistet.

Aufteilung der Funktionalitäten

Die Funktionalitäten der Anlage werden in sogenannte Skills unterteilt. Diese Skills bilden die Grundlage

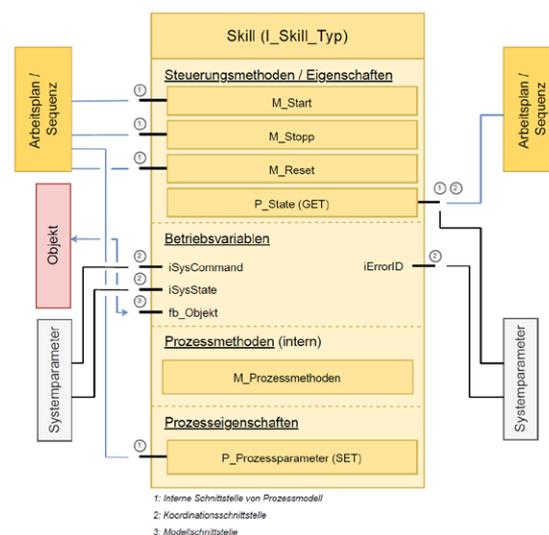
des Prozessmodells und repräsentieren die einzelnen Arbeitsschritte der Anlage. Auch für die Erstellung von Skills wurde eine standardisierte Struktur entwickelt, die sowohl die Interaktion mit anderen Elementen innerhalb des Prozessmodells als auch die Interaktion mit dem Anlagenmodell umfasst (Ersichtlich in Abbildung).

Resultate

- Die Trennung zwischen Prozess- und Anlagenmodell erlaubt das einfache Entfernen, Ersetzen oder Hinzufügen von Komponenten, ohne dass umfangreiche Änderungen am definierten Prozess notwendig sind.
- Durch eine Skill- und Objektbibliothek können Prozesse und Anlagen schnell und effizient realisiert werden. Die erstellten Bausteine reduzieren Entwicklungszeiten erheblich.
- Die standardisierte Struktur sorgt für Konsistenz im Aufbau und minimiert Fehlerquellen. Gleichzeitig wird die Realisierung des Programms durch klare Vorgaben und Strukturen vereinfacht.



Yannick Spatz
yannick.spatz@gmail.com



Schematischer Aufbau eines Skills und dessen Schnittstellen