

A *Computer &* AUTOMATION

SONDERDRUCK



Den Test automatisieren

Maschinen- und Anlagenbauer gehen mehr und mehr dazu über, Steuerungssoftware vor der Inbetriebnahme zu simulieren, um etwaige Fehler vorab korrigieren zu können.

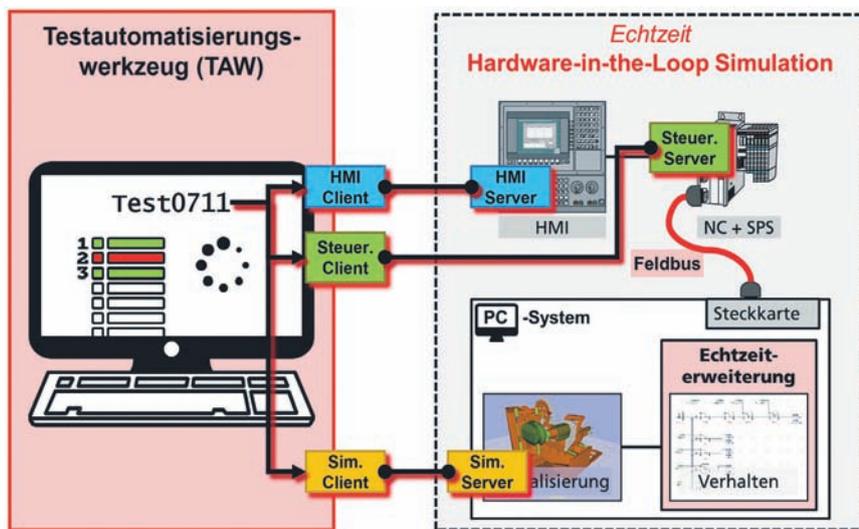
Statt die Tests wie bisher manuell durchzuführen, empfiehlt sich der Umstieg auf ein automatisiertes Vorgehen. Worauf kommt es dabei an?

Die Anforderungen im Maschinen- und Anlagenbau seitens der Kunden wachsen stetig: Immer kürzere Produktions- und Umrüstzeiten, permanenter Termindruck, wachsende Variantenvielfalt und so weiter. Um diese Herausforderungen zu bewältigen, kommen zunehmend schon vor der Inbetriebnahme Digitale Zwillinge zum Einsatz. Sie ermöglichen neben einer methodischen Testplanung und -durchführung die frühzeitige

Behebung etwaiger Software-Fehler. Gleichzeitig wird die Systemarchitektur der Steuerungssoftware moderner Anlagen allerdings zunehmend heterogener; nicht zuletzt aufgrund von Maschinenführern, die – dank Technologien wie Smartphones – immer offener gegenüber intuitiven Bedienkonzepten und anpassbaren Oberflächen in ihren Anlagen sind.

Häufig besteht die Systemarchitektur aus den Kernbestandteilen

HMI, SPS und NC-Kern. Da es zwischen diesen Komponenten sowie der Anlage und der Leitebene im Unternehmen unterschiedliche Schnittstellen gibt, steigt die Komplexität solcher Systeme weiter an. Um zu überprüfen, ob eine Steuerungssoftware tatsächlich im gewünschten Sinne funktioniert, sind daher Tests unabdingbar. Der wichtigste Teil der Testaktivitäten – nämlich die Bewertung der Interaktion der Steuerungssoftware – findet



Prinzipieller Aufbau eines Testplatzes zum automatisierten Test von Steuerungssoftware.

allerdings auch heute noch vielfach erst während der Integration und Inbetriebnahme statt.

So viel zum Status quo. Im Zeitalter der Digitalisierung wünschen sich nun viele Unternehmen, früher mit der Testphase beginnen zu können. Um das zu ermöglichen, setzen immer mehr Firmen auf die virtuelle Inbetriebnahme (VIBN). Hierzu verbinden sie die reale Steuerungshardware und -software mit einem virtuellen Modell der Anlage. In letzterem lässt sich das Verhalten in Echtzeit simulieren, wodurch ein sogenannter Hardware-in-the-Loop-Aufbau (HiL) entsteht. Damit kann das Gesamtsystem schon deutlich früher sowohl einem Gut- als auch einem Schlechtfalltest unterzogen werden. Während die Gutfalltests darauf abzielen, die vom Kunden gewünschte Performance und Funktionalität zu überprüfen, werden bei den Schlechtfalltests Zuverlässigkeit und Robustheit der Software getestet, indem die Reaktion der Steuerungssoftware in expliziten Gefahren- und Störsituationen überprüft wird. Unverzichtbare Abnahmetests an der realen Anlage beim Hersteller (Factory Acceptance Test) beziehungsweise beim Kunden (Site Acceptance Test) lassen sich damit wesentlich schneller abschließen.

Bis dato stellen VIBN-Tests Unternehmen jedoch vor gewisse Heraus-

forderungen. Prinzipiell folgen sie zumeist einem ähnlichen Schema: Ein Prüfer führt die Testschritte durch, während er anhand einer umfangreichen Checkliste sämtliche vorgegebenen Bedienhandlungen am HiL-Aufbau manuell durchführt. Er provoziert Störsituationen, indem er entsprechende Eingriffe am Digitalen Zwilling vornimmt. Damit lässt sich die Wirkung der steuerungstechnischen Algorithmen zur Fehlererkennung und primären Fehlerreaktionen beurteilen.

Das Problem bei der manuellen Ausführung dieser Tests: Es können sich immer wieder Fehler einschleichen, wenn es darum geht, die Zustände der Anlage zu beobachten – etwa die falsche Interpretation eines Testschritts oder eine Unaufmerksamkeit des Prüfers. Außerdem müssen bei Änderungen an der Steuerungssoftware oder an Anlagenkomponenten häufig Bedienhandlungen komplett neu ausgeführt werden (Regressionstests) – was einen hohen Zeitaufwand verursacht.

Außerdem lässt es sich nicht vermeiden, dass aufgrund der Modularität und der Konfigurierbarkeit der Anlagen hohe Wiederholungsraten einzelner Tests entstehen. Nicht zuletzt sorgen kontinuierliche Updates seitens des Steuerungsherstellers ebenfalls dafür, dass regelmäßige

Regressionstests der gesamten Steuerungssoftware notwendig sind.

Um diese Herausforderungen meistern zu können, bedarf es eines geeigneten Testautomatisierungswerkzeugs (TAW).

Die Kriterien automatisierter Testwerkzeuge

Bei der Auswahl einer entsprechenden Lösung sind verschiedene Aspekte zu beachten: Zunächst sollte das TAW die Definition von wiederverwendbaren Testbausteinen unterstützen. Das ist erforderlich, um die erstellten Testabläufe in verschiedenen Phasen der Entwicklung einer Anlage wiederverwenden zu können. Es ist außerdem essenziell, dass Testbausteine ohne Programmierkenntnisse erstellbar sind. Nur so ist sichergestellt, dass sich alle Projektteilnehmer auch an der Testphase beteiligen können.

Dabei wäre es zudem hilfreich, wenn das TAW die Testabläufe leicht verständlich darstellt und intuitiv zu bedienen ist. Das betrifft im Übrigen auch die Interaktion des TAW mit der Steuerung: Hierfür darf kein Expertenwissen hinsichtlich der Kommunikationsprotokolle oder ähnlichem notwendig sein. Da mit dem TAW zumeist sehr komplexe Anlagen getestet werden, ist es weiterhin von Vorteil, wenn das Werkzeug parallele Abläufe, Verzweigungen und Schleifen im Testablauf zulässt.

Mehrere Steuerungen im Zusammenspiel

Häufig stehen Unternehmen bei der Testautomatisierung vor der Herausforderung, dass komplexe Anlagen mehrere Steuerungen beinhalten. Darum sollte das TAW in der Lage sein, unterschiedliche Digitale Zwillinge und deren Steuerungen im Zusammenspiel zu testen. Darüber hinaus ist eine detaillierte Protokollfunktion nicht zu vernachlässigen: Es ist schließlich Sinn und Zweck der Testautomatisierung, Fehlerursachen schnell und sicher zu lokalisieren, damit sie einfach behebbar

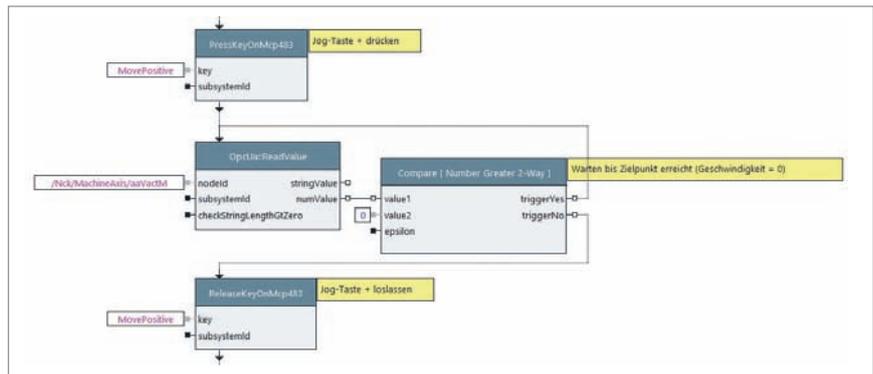
sind. Das betrifft auch die Aufzeichnung von Eingabe- und Ausgabedaten der einzelnen Testbausteine sowie die Testreports.

Im Zuge immer kürzerer Produktionszeiten ist bei der Testautomatisierung die zeitliche Komponente ganz entscheidend: Um möglichst kurze Turn-around-Zeiten zu erzielen, sollte es möglich sein, Änderungen unmittelbar während des Testablaufs durchführen und überprüfen zu können. Ein weiterer wichtiger Aspekt ist das sogenannte Debugging der Testbausteine: Haltepunkte sollten sich setzen und Tests schrittweise abarbeiten und mitverfolgen lassen.

Last but not least spielt der spezifische Aufbau der Anlagen eine nicht zu unterschätzende Rolle bei der Testautomatisierung. In der Regel sind Anlagen modular aufgebaut und verfügen daher über eine hohe Varianz hinsichtlich der jeweiligen kundenspezifischen Ausführung. Je nach Anlagenaufbau können sich die Hardware- und Software-Adressen von Komponenten, Parametern, Ein- und Ausgängen etc. ändern. Darauf sollte sich das TAW einstellen. Um Testbausteine einfach, sicher und schnell an die jeweilige Ausführung der Anlage anpassen zu können, ist es notwendig, Parameter symbolisch zu adressieren. Eine weitere Herausforderung stellt auch das vom Kunden frei wählbare Steuerungssystem dar. Darum braucht es ein TAW mit einer möglichst herstellerunabhängigen Kompatibilität zu Steuerungssystemen und -komponenten. Ziel ist es schließlich, dieselben Testbausteine völlig unabhängig vom Hersteller des jeweiligen Steuerungssystems zum Test einzusetzen.

Entscheiden sich Unternehmen schlussendlich für eine Testautomatisierung, profitieren sie in vielerlei Hinsicht: Mannlose Tests sind problemlos während der Nacht oder am

Aus der Liste der geladenen Bibliotheken, die für den Testablauf zur Verfügung stehen, wählt der Benutzer Bausteine aus, die er für seinen Testablauf nutzen möchte.



Ausschnitt aus einem Testablauf. In diesem Beispiel wird eine Achse bis zum Software-Endschalterverfahren.

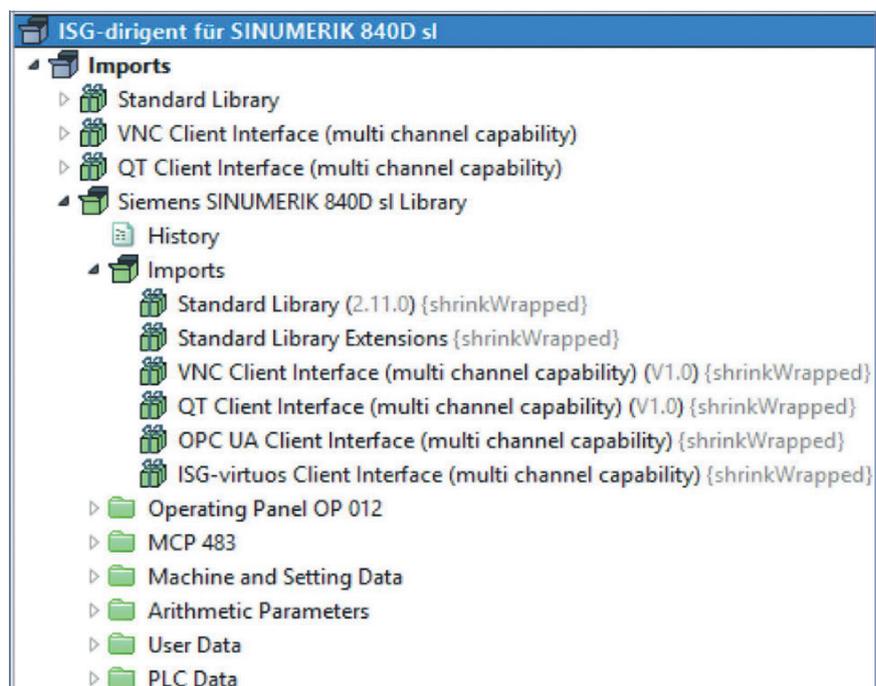
Wochenende möglich. Dementsprechend sind Dauertests kein Problem, sodass Unternehmen selbst sporadisch auftretende Fehler entdecken können. Auch sind ‚Feuerwehreinsätze‘ direkt vor Ort nicht mehr notwendig, da sich Fehler an der Steuerungssoftware bereits vorab beseitigen lassen, was wiederum ein langfristig gesteigertes Qualitäts-Image beim Endkunden schafft.

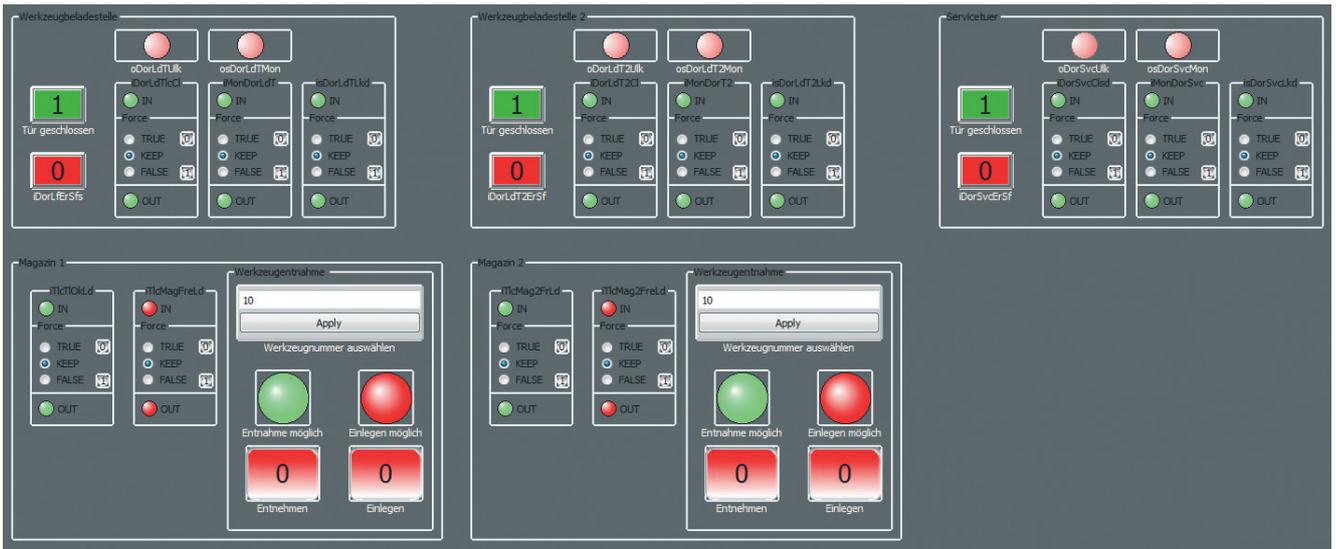
Ein Beispiel für ein modernes Testautomatisierungswerkzeug, wie vorab beschrieben, ist das sogenannte Tool ‚ISG-dirigent‘. Entstanden ist es im Rahmen eines gemeinsamen

Forschungsvorhaben des Institutes für Steuerungstechnik der Werkzeugmaschinen und Fertigungseinrichtungen (ISW) der Universität Stuttgart mit dem Unternehmen Industrielle Steuerungstechnik GmbH (ISG). Die Software baut auf zwei Komponenten auf: der Simulationslösung ISG-virtuos, die in der Lage ist, Digitale Zwillinge zu simulieren, sowie expecco, einem Framework zur Testautomatisierung des Unternehmens Exept AG.

TAW am konkreten Beispiel

Eine im Werkzeugmaschinenumfeld häufig eingesetzte Steuerung ist die





Digitaler Zwilling einer Steuertafel zur Bedienung von Maschinenkomponenten. Mit den Tasten (Buttons) kann der Bediener die Aktionen der Maschinenkomponenten anstoßen.

Sinumerik 840D sl von Siemens, an der stellvertretend im Folgenden der Aufbau von ISG-dirigent beschrieben wird: Bereits das Basiswerkzeug expecco bringt neben der Standard-Bibliothek weitere wichtige Bibliotheken mit, die für den Test solcher Anlagen notwendig sind. So ermöglicht zum Beispiel die Qt-Bibliothek die Kommunikation der Testbausteine mit den Elementen der Bedienoberfläche der CNC-Steuerung, während die VNC-Bibliothek die Übertragung von Tastatureingaben an das Bedienfeld der Sinumerik erlaubt. Die SCP-Bibliothek wiederum ermöglicht die Übertragung von Dateien (etwa von NC-Programmen) von und zur Steuerung.

Doch erst die ISG-Bibliothek für die Siemens Sinumerik 840D sl ergänzt das Werkzeug zur vollen Funktionalität von ISG-dirigent. Aufbauend auf den Bibliotheken ‚OPC UA Client Interface‘ und ‚ISG-virtuos Client Interface‘ zur Kommunikation mit der Steuerung beziehungsweise mit dem Digitalen Zwilling, ist diese Bibliothek in folgende Unterbereiche unterteilt:

- Bausteine der Gruppe Bedienfeld (OP 012) zur nutzergerechten Interaktion der Testbausteine mit der Steuerung bezogen auf das Bedienfeld.

- Bausteine der Gruppe Maschinensteuertafel (MCP483) zur Interaktion der Testbausteine mit der Steuerung bezogen auf die Maschinensteuertafel.
- Bausteine der Gruppe Maschinen- und Settingdaten (Machine and Settings Data) für einen Zugriff der Testbausteine auf diese Daten der Steuerung.
- Bausteine der Gruppen Arithmetikparameter (Arithmetic Parameters) und Anwenderdaten (User Data) für den Zugriff der Testbausteine auf diese Daten der Steuerung.
- Bausteine der Gruppe SPS-Daten (PLC Data) für den Zugriff der Testbausteine auf diese Daten der Steuerung.

Wenn der Anlagenhersteller eine Systematisierung seiner Testabläufe vorgenommen hat, gelingt der Einstieg in die Testautomatisierung während der virtuellen Inbetriebnahme am effizientesten. Dazu gehört es jedoch auch, die Testabläufe auf Basis einzelner Testschritte zu definieren, die erwarteten Ergebnisse eines Testablaufs zu erfassen und Protokoll-Templates zu erstellen. Danach können Anlagenhersteller mit der Identifizierung der Gleichteile zwischen Testabläufen beginnen. So lässt sich

Schritt für Schritt eine Bibliothek an wiederverwendbaren Testschritten aufbauen. Das Protokoll-Template im TAW gewährt dabei eine schnellere Übersicht, wie ein Test verlaufen ist und an welcher Stelle eventuell Fehler aufgetreten sind.

Am ISW wird zudem im Bereich der Automatisierung von Inbetriebnahme-Tests geforscht. Darunter fällt auch die intelligente Generierung von Daten für den Testablauf sowie die Erweiterung der Testausführung in den Echtzeit-Teil des HiL-Aufbaus. Geplant ist, im Rahmen der Arbeiten an ISG-dirigent weitere Bibliotheken für die Steuerungen der Firmen Fanuc, Heidenhain, Bosch-Rexroth oder auch Beckhoff zu ergänzen.

gh



DR.-ING. GERHARD KRESBER
ist verantwortlich für die Testautomatisierung von Steuerungssoftware bei ISG Industrielle Steuerungstechnik.



KARL KÜBLER
ist wissenschaftlicher Mitarbeiter am ISW der Uni Stuttgart.