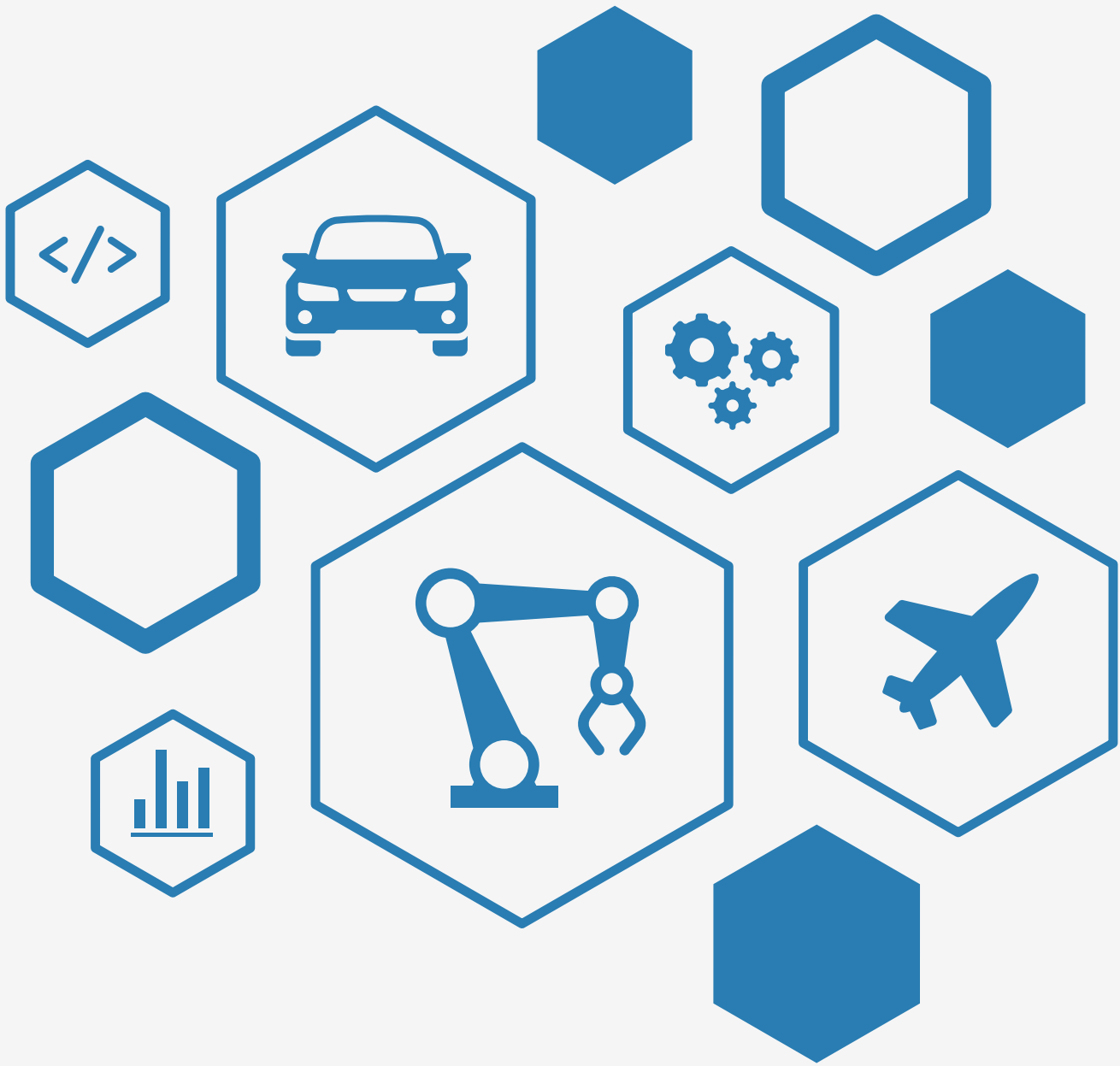


Systementwicklung für Industrie 4.0 mit MATLAB und Simulink

Antworten auf die Top 3 Fragen von Ingenieuren



Daten – im Zentrum von Industrie 4.0

Philipp H. F. Wallner, Industry Manager, MathWorks

Seth DeLand, Product Manager, MathWorks

Die Welt der Mechatronik ist im Umbruch. Moderne Produktionsanlagen sind hochintegrierte mechatronische Systeme mit erheblichem Anteil an Embedded Software. Dies zwingt Ingenieure aus allen drei mechatronischen Disziplinen – Maschinenbau, Elektrotechnik und Softwareentwicklung – dazu, parallel zueinander zu arbeiten und Entwicklung, Test und Verifikation von Maschinensoftware so weiterzuentwickeln, dass sie am Ende die geforderte Funktionalität und Qualität erzielen.

Erkenntnisse und Wertschöpfung aus Daten

Ein zentraler Motor von Industrie 4.0 sind wachsende Datenmengen. Vision-Sensoren, elektrische und hydraulische Antriebe und sogar Fertigungsmaschinen und Kraftwerke sammeln während ihres Betriebes ständig Messdaten. Diese Daten enthalten Informationen, aus denen sich mit prädiktiven Modellen und Algorithmen ein konkreter wirtschaftlicher Mehrwert gewinnen lässt. Durch Machine Learning etwa lassen sich Modelle anhand historischer Sensordaten so trainieren, dass sie künftige Maschinenausfälle vorhersagen können und damit Produktionsausfälle verhindern.

Datenanalyse als Teil von Embedded Systems

Im Zuge der Digitalisierung mit Industrie 4.0 liefern Softwarekomponenten einen erheblichen Anteil des Mehrwerts künftiger Maschinen und Fertigungsanlagen. Embedded Software auf SPSen, Industrie-PCs oder FPGAs nutzen Reglerkomponenten, die die Produktqualität sicherstellen, sowie Predictive Maintenance-Algorithmen, die längere Laufzeiten ohne Wartungseingriffe ermöglichen. Embedded Software bildet zudem die Überwachungslogik für Zustandsmaschinen und dient zur Berechnung optimierter Bewegungstrajektorien – auch in sicherheitskritischen Systemen.

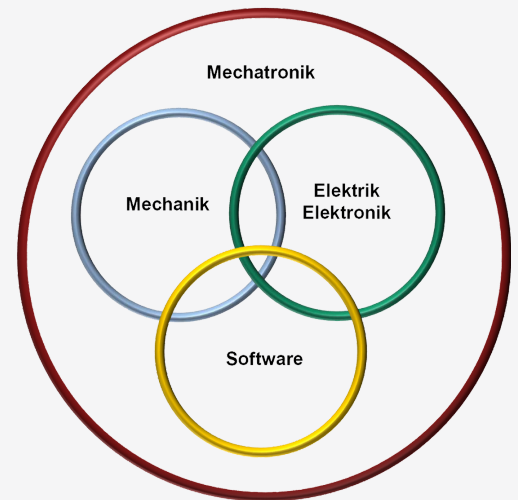
Effizienz durch modellbasierte Entwicklung

Um die Marktführerschaft zu behaupten, müssen Ausrüstungshersteller Kompetenzen und Fachwissen hinsichtlich moderner Ansätze und Methoden der Softwareentwicklung aufbauen – um die Effizienz-, Kosten- und damit Wettbewerbsvorteile auszuschöpfen, die Industrie 4.0 verspricht. Werkzeuge wie MATLAB® und Simulink® für die modellbasierte Entwicklung erhöhen sowohl die Produktivität der Softwareentwicklung als auch die Zuverlässigkeit der entwickelten Systeme. Mit diesen Tools lässt sich Software für Automatisierungskomponenten modular entwickeln und unabhängig von Hardware testen. Und mittels automatischer Codegenerierung lassen sich Algorithmen per Knopfdruck auf Embedded-Plattformen implementieren.

Realisierung von Industrie 4.0

Industrie 4.0 steht und fällt damit, wie mit der wachsenden Komplexität der Software und der stetig zunehmenden Menge gesammelter Maschinendaten umgegangen wird. Langfristig stellt dieser Entwicklungstrend Ingenieure vor die Herausforderung, sich neue Methoden der Softwareentwicklung sowie moderne datenanalytische Verfahren aneignen zu müssen, um diese Komplexität zu beherrschen.

[Lesen Sie hier den Gesamtartikel in Embedded Computing Design \(Englisch\)](#)



Die 3 drängendsten Fragen von Ingenieuren

1. Wie lassen sich Maschinendaten zur Verbesserung der Produktionsleistung oder zur Vorhersage von Ausfällen nutzen?

Mit MATLAB können Sie **Maschinen- und Produktionsdaten analysieren**, Muster erkennen sowie **Maschinenparameter optimieren** und Wartungsintervalle reduzieren.

Workflows

Offline Datenanalysen werden mit Daten durchgeführt, die in Dateien, Datenbanken oder in der Cloud gespeichert sind.

Workflows:

- Datenimport nach MATLAB
- Analyse der Daten
- Einsatz der Ergebnisse zur Optimierung der maschinellen Produktion oder von Wartungsintervallen

Online Datenanalysen erlauben die laufende Analyse des Maschinenverhaltens

Workflows:

- Einlesen von Daten über OPC UA und andere Standards
- Implementierung von Echtzeit-Code (C/C++, IEC 61131-3) auf SPS oder Embedded Controllern

MATLAB hilft bei der **Analyse gemessener Signale** (z. B. Vibrationsdaten) durch:

- Transformation in den Frequenzbereich
- Anwendung von Filtern

Wussten sie schon? Vorgefertigte Funktionen sparen Zeit im Vergleich zu manueller Programmierung.

Ingenieure, die mit großen Mengen an Businessdaten und technischen Daten arbeiten, nutzen **Machine Learning** zur Mustererkennung und zum Aufbau von Modellen, die künftige Ereignisse mithilfe historische Daten vorhersagen (etwa für **Predictive Maintenance**).



Mondi implementiert eine statistikgestützte Zustandsüberwachung und Predictive Maintenance für Fertigungsprozesse mit Machine Learning

Kernprodukte

- [MATLAB](#)
- [OPC Toolbox™](#)
- [Data Acquisition Toolbox™](#)

Anwenderbericht: [Newport Corporation verkürzt die Zeit für die Datenerfassung und -analyse um Hunderte Stunden](#)

Webinar: [Datenanalyse im Internet der Dinge \(IoT\) – wie MATLAB Ihnen hilft, die Datenflut zu meistern](#)

Technischer Artikel: [5 Arten von Daten und wie man sie mit MATLAB analysiert](#)

Kernprodukte

- [MATLAB](#)
- [Signal Processing Toolbox™](#)

Anwenderbericht: [BuildingIQ entwickelt proaktive Algorithmen für die Verbrauchsoptimierung von Klimasystemen in Großgebäuden.](#)

Kernprodukte

- [MATLAB](#)
- [Statistics and Machine Learning Toolbox™](#)
- [Neural Network Toolbox™](#)

Anwenderbericht: [Mondi implementiert mit Machine Learning eine statistikgestützte Zustandsüberwachung und Predictive Maintenance für Fertigungsprozesse](#)

Webinar: [Predictive Maintenance mit MATLAB: Eine Anwendungsstudie](#)

Technischer Artikel: [Datengetriebene Einblicke durch MATLAB-Analysen: Eine Studie zur Vorhersage von Stromverbräuchen](#)

Überblick: [Predictive Maintenance](#)

2. Wie lassen sich optimierte Regler für immer komplexere Systeme erfolgreich entwickeln?

Die **Virtuelle Inbetriebnahme auf Grundlage von modellbasierter Entwicklung** hilft Ihnen bei der Optimierung von Reglerstrukturen und -parametern. Mittels **Simulation** können Sie Regler und Regelparameter schneller, sicherer und kostengünstiger testen und optimieren.

Workflows

MATLAB und Simulink Produkte für den **Reglerentwurf** unterstützen:

- Modellierung von Regelstrecken und Anlagen
- Optimierung von Parametern
- Automatisierte Tests und Berichte
- Automatische Codegenerierung (C/C++, IEC 61131-3, HDL)

Wussten Sie schon? Automatische Codegenerierung erzeugt hocheffizienten Code für Ihren Regler und hilft gleichzeitig, Programmierfehler zu vermeiden.

Kernprodukte

- *Simulink*
- *Stateflow®*
- *Simulink Control Design™*
- *Control System Toolbox™*
- *Model Predictive Control Toolbox™*

Anwenderbericht: *Metso entwickelt mit Model-Based Design einen Controller für energiesparende digitalhydraulische Systeme in der Papierindustrie*

Übersicht: *Software für den Reglerentwurf*

Der Entwurf eines Regelungssystems beginnt mit einem **Modell der Regelstrecke**, das zur Systemsimulation dient. MATLAB und Simulink Produkte unterstützen verschiedene Ansätze zur Modellierung von Regelstrecken:

- **Physikalische Modellierung** mit Blöcken für mechanische, elektrische, hydraulische, pneumatische und thermische Komponenten.
- **Systemidentifikation** auf der Basis von Messdaten

Wussten Sie schon? Sie können Ihr gesamtes System entwerfen und testen, ohne auf die Mechanik warten zu müssen.

Kernprodukte

- *Simscape™ Produkte für die Physikalische Modellierung*
- *System Identification Toolbox™*

Webinar: *Virtuelle Inbetriebnahme mit Simulink*

Anwenderbericht: *manroland entwickelt einen Präzisionsregler für kommerzielle Druckmaschinen*

Übersicht: *Erzeugung exakter Regelstrecken-Modelle*

Technischer Artikel: *Modellierung komplexer mechanischer Strukturen mit SimMechanics*

Systemmodelle lassen sich **frühzeitig testen und verifizieren** durch:

- Desktop-Simulationen
- Echtzeit-Tests
- Automatisierte Testszenerien
- Formale Verifikation

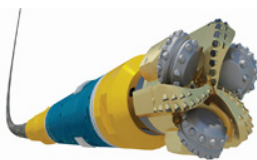
Wussten Sie schon? Entwicklungsfehler können Sie bereits vor dem Bau der Mechanik und der Programmierung von Produktionssoftware entdecken.

Kernprodukte

- *Simulink Design Verifier™*
- *Simulink Verification and Validation™*
- *Simulink Test™*
- *Polyspace Bug Finder™ und Polyspace Code Prover™*
- *Simulink Real-Time™*

Anwenderbericht: *Baker Hughes verbessert die Genauigkeit von Ausrüstung für Öl- und Gasbohrungen*

Übersicht: *Verifikation, Validierung und Testen*



Baker Hughes verbessert die Genauigkeit von Ausrüstung für Öl- und Gasbohrungen

3. Wie kann ich Regelalgorithmen und Zustandsmaschinen auf Industriesteuerungen (SPS) oder Industrie-PCs implementieren?

Durch **Automatische Codegenerierung** können Sie Ihre in Simulationen getesteten Algorithmen auf üblichen SPS- oder Industrie-PC-Plattformen implementieren – entweder **IEC 61131-3-konform** (Structured Text oder Kontaktplan) oder als **C/C++ Code**.

Workflows

Automatische Codegenerierung konvertiert getestete Modelle und Algorithmen in hocheffizienten Echtzeit-Code (C/C++ oder IEC 61131-3).

Wussten Sie schon? Dieser Workflow ist erheblich schneller als die manuelle Programmierung. Entwickler erhöhen damit ihre Produktivität für Industriesteuerungen um 200 bis 300% - bei erheblich verringerter Fehleranfälligkeit.

IEC 61508 ist der zentrale Sicherheitsstandard für Industriesteuerungen. Die folgenden Produkte sind vom **TÜV SÜD** zertifiziert:

- Embedded Coder
- Polyspace code verifiers
- Simulink PLC Coder

Wussten Sie schon? Zertifizierungen des TÜV SÜD basieren auf anwendungsspezifischen Verifikations- und Validierungs-Workflows, wie sie im IEC Certification Kit vorgesehen sind.

Automatische Codegenerierung ist für nahezu alle gängigen **SPS- und Industrie-PC-Plattformen** verfügbar. MathWorks kooperiert eng mit seinen Hardware-Partnern, um die Kompatibilität mit deren Integrierten Entwicklungsumgebungen (IDEs) sicherzustellen.

SPS-Partner und deren Entwicklungsumgebungen:

- 3S CODESYS
- B&R Automation Studio™
- Bachmann M-Target
- Beckhoff® TwinCAT™
- Mitsubishi CW Workbench
- Omron Sysmac Studio
- Phoenix Contact® PC WORX™
- Rexroth IndraWorks
- Rockwell Automation® RSLogix™ and Studio 5000
- Siemens STEP 7, WinAC and TIA Portal

Kernprodukte

- *Simulink PLC Coder™*
- *MATLAB Coder™*
- *Simulink Coder™*
- *Embedded Coder™*

Anwenderbericht: *ENGEL beschleunigt die Entwicklung von Reglern für Spritzgussmaschinen*

Webinar: *Elektrisch und funktional sichere Antriebssysteme mit Model-Based Design*

Kernprodukte

- *IEC Certification Kit™*
- *Polyspace Bug Finder™ and Polyspace Code Prover™*
- *Embedded Coder*
- *Simulink PLC Coder*

Technischer Artikel: *Verbesserung der Softwarequalität durch statische Codeanalyse*

Link: *Unterstützte Standards und Produkte*

Unterstützte SPS-Plattformen

- *Unterstützung für Drittanbieter-IDEs in Simulink PLC Coder*
- *Hardwareunterstützung für die Erzeugung von SPS-Code (C/C++ and IEC 61131-3)*

User Story: *AVL entwickelt einen dynamischen Regler für ein System zur Motorabstimmung durch Generierung von Embedded Code für SPS-Plattformen*



AVL entwickelt einen dynamischen Regler für die Motorabstimmung durch automatische Generierung von Embedded Code für ihre SPS.

Stimmen von Anwendern

“Durch Model-Based Design verkürzen wir Entwicklungszeiten, reduzieren Kosten und sind schneller auf dem Markt.

Automatische Codegenerierung spart enorm viel Zeit. Zudem können wir Entwürfe ganz einfach aktualisieren, indem wir das Modell verändern, und sind sicher, dass alles so implementiert wird, wie es sein soll. Unsere **Entwurfsiterationen** – von Änderungen bis zum Hardwaretest – **dauern nur fünf Minuten.**“

– AREND-JAN BELTMAN, CENTRE FOR CONCEPTS IN MECHATRONICS

“Bei früheren Projekten haben wir Structured Text für Vorsteuerungen per Hand geschrieben. Mit diesem manuellen Ansatz wäre es aber viel zu kompliziert gewesen, die Regelung für den Bionischen Handling-Assistenten auf einer SPS zu implementieren – **wir hätten es schlicht nicht hinbekommen.** Mit dem Simulink PLC Coder konnten wir den **Structured Text in Minuten automatisch generieren.**“

– DR. RÜDIGER NEUMANN, FESTO

“Als Fertigungsunternehmen haben wir **keine mit Machine Learning erfahrenen Datenexperten**, aber MathWorks hat uns die Tools und das technische Know-How zur Verfügung gestellt, mit denen wir innerhalb von Monaten ein marktreifes vorbeugendes Maintenance-System entwickeln konnten.“

– DR. MICHAEL KOHLERT, MONDI

“Als Prozessingenieur hatte ich keinerlei Erfahrung mit **Neuronalen Netzen oder Machine Learning.** Ich habe mich durch die MATLAB-Beispiele gearbeitet, bis ich die besten Machine Learning-Funktionen für die Erzeugung von virtueller Messtechnik gefunden hatte. **In C oder Python wäre das völlig unmöglich gewesen** – es hätte viel zu lange gedauert, die richtigen Pakete zu finden, zu validieren und zu integrieren.“

– EMIL SCHMITT-WEAVER, ASML

“Unser Shopfloor Management-System AMS ZPoint-CI sammelt rund um die Uhr **enorm viele Daten von Maschinen, Prozessen und Produkten.** Durch sofortige Analyse dieser Daten in MATLAB und AMS Analysis-CI haben wir **die Genauigkeit um das zehnfache gesteigert und die Taktzeit um 30% verkürzt**, was einen erheblich gesteigerten Produktionsausstoß bedeutet.“

– ALEXANDER MEISINGER, STIWA

Weitere Ressourcen

Mit MATLAB und Simulink müssen Ingenieure keine Fachleute für die Programmierung sein, um hochwertige Software zu entwickeln und zu implementieren. Sie können sich beim Entwurf der Funktionalität ihrer Systeme ganz auf ihre technischen Kernkompetenzen konzentrieren. MATLAB und Simulink können auch Sie einfach

- lernen – mit [kostenlosen Tutorials](#) und [professionellen Schulungen](#)
- anwenden – mit [tausenden Codebeispielen](#) und Applikationen von MATLAB-Ingenieuren sowie aus der [User Community](#)
- beherrschen – weil Sie [Antworten](#) auf Ihre kniffligsten Fragen von der User Community und unserem [Technischen Support](#) erhalten oder in der umfassenden [Dokumentation](#) finden.