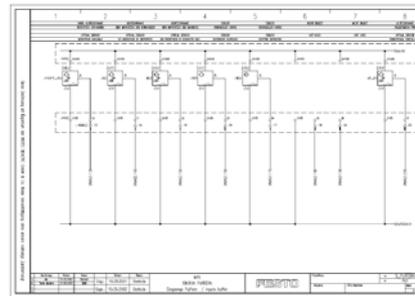


EasyKit - Innovative Entwicklungs- und Didaktikwerkzeuge für mechatronische Systeme

Dr.-Ing. Reinhard Pittschellis
Festo Didactic GmbH & Co. KG
Denkendorf
pitt@de.festo.com
www.festo-didactic.de

Ausgangslage



Schaltungsentwurf mit E-CAD

Nachteile:

- Sequentielles Vorgehen
- Spezialkenntnisse auf allen Ebenen
- Hohe Einmalkosten



Platinenherstellung



Mechatronisches System mit Embedded Controller

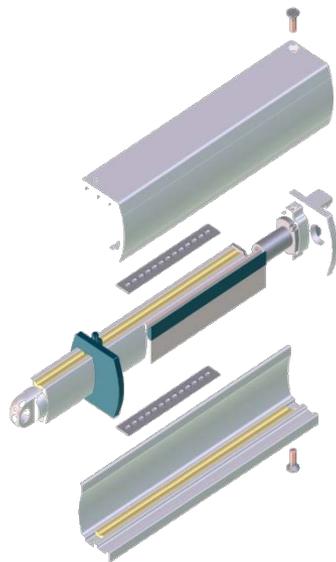
```

perspektive_02_13ab -Modell (code)
[Abgelesen] [Rechnen]

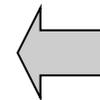
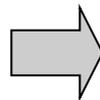
Dem Monatliche Leistung ermitteln
For MA = 1 To Anzahl_MA Step 1                                'Über alle Monate
For Monat = 1 To 12 Step 1                                    'Über alle Monate
summe = 0                                                    'Summe auf Null set
For projekte = 1 To Anzahl_projekte Step 1                    'Über alle Projekte
    'wenn Projekt läuft und Starttermin <
    status = Worksheets("Planung").Cells(projekte + aspc, spalte_status).Value und Starttermin <
    beginn = Worksheets("Planung").Cells(projekte + aspc, spalte_beginn).Value beginn = Worksheets("Planung").Cells(projekte + aspc, spalte_beginn).Value
    ende = Worksheets("Planung").Cells(projekte + aspc, spalte_ende).Value ende = Worksheets("Planung").Cells(projekte + aspc, spalte_ende).Value
    dauer = ende - beginn + 1
    If ((status = 1) Or (status = 2) Or (status = 3) And (beginn <= Monat) And (ende >= Monat) And (dauer >= ende - beginn + 1)
        Next projekte
        Worksheets("Auslastung").Cells(Monat + asau, MA + asau).Value = summe 'speichere
    Next Monat
Dem Monatlichen Budget ermitteln
For Monat = 1 To 12 Step 1                                    'Über alle Monate
If Monat = 1 Then
summe = 0
Else
summe = Worksheets("Auslastung").Cells(Monat + asau - 1, 25).Value
End If
For projekte = 1 To Anzahl_projekte Step 1                    'Über alle Projekte
    'wenn Projekt läuft und Starttermin <
    status = Worksheets("Planung").Cells(aspc + projekte, spalte_status).Value und Starttermin <
    beginn = Worksheets("Planung").Cells(projekte + aspc, spalte_beginn).Value beginn = Worksheets("Planung").Cells(projekte + aspc, spalte_beginn).Value

```

Programmierung

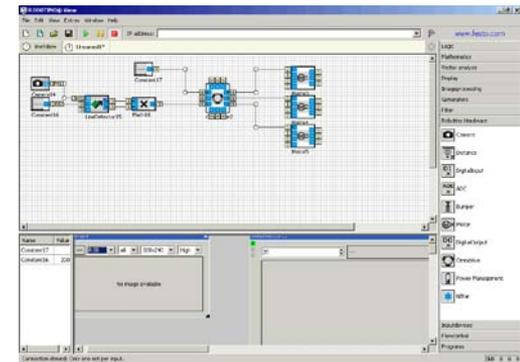
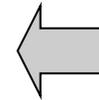
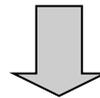
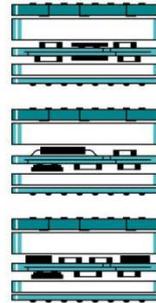
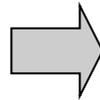
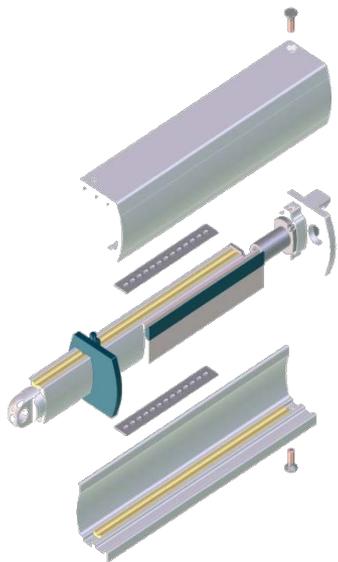


Konstruktion mit CAD



Neues Konzept mit Easy-Kit

Schaltung aus Funktionsblöcken



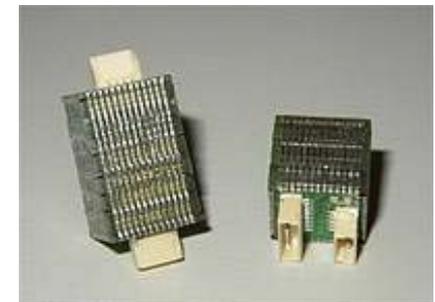
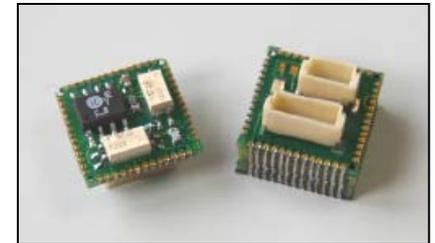
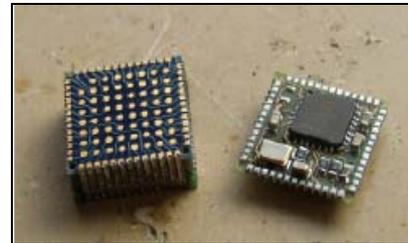
Konstruktion mit CAD

Mechatronisches System

Blockorientierte Programmierung

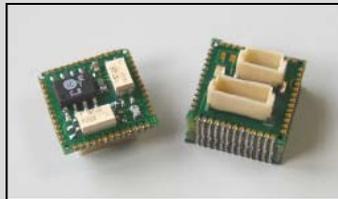
Was es schon gab – MatchX-Bausteine

- Genormt im VDMA-Einheitsblatt 66305
- Einfach stapelbar
- Unterschiedliche Bausteine:
 - Mikroprozessoren
 - Signalwandler
 - Treiberbausteine
 - Buskoppler
 - Spannungsversorgung

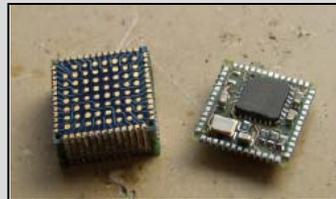


Die Idee- Durchgängiges Hardware-Software-Codesign

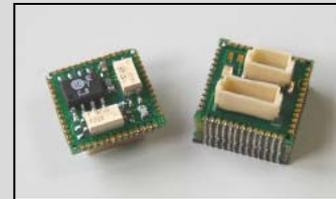
Hardware-
bausteine



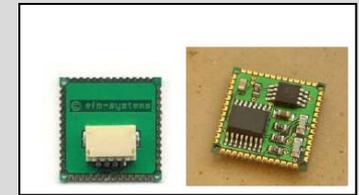
Buskoppler



Prozessor-Baustein

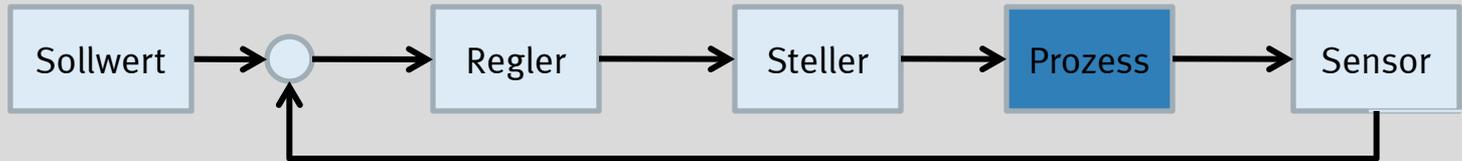


Treiber-Baustein



Signalwandler-Baustein

Regelkreis



Programm

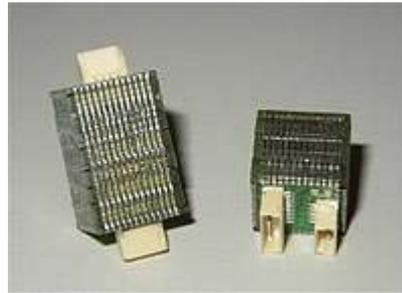


Drei-Stufen-Konzept – wirtschaftlich für alle Stückzahlen

Schritt 1



Schritt 2



Schritt 3



Funktionsmuster

Modulare
Funktionsbausteine

Prototyp

Eigenständiges
Komplettsystem

Serienprodukt

Anwendungsspezifisches
mechatronisches System

Das Konsortium



GEFÖRDERT VOM



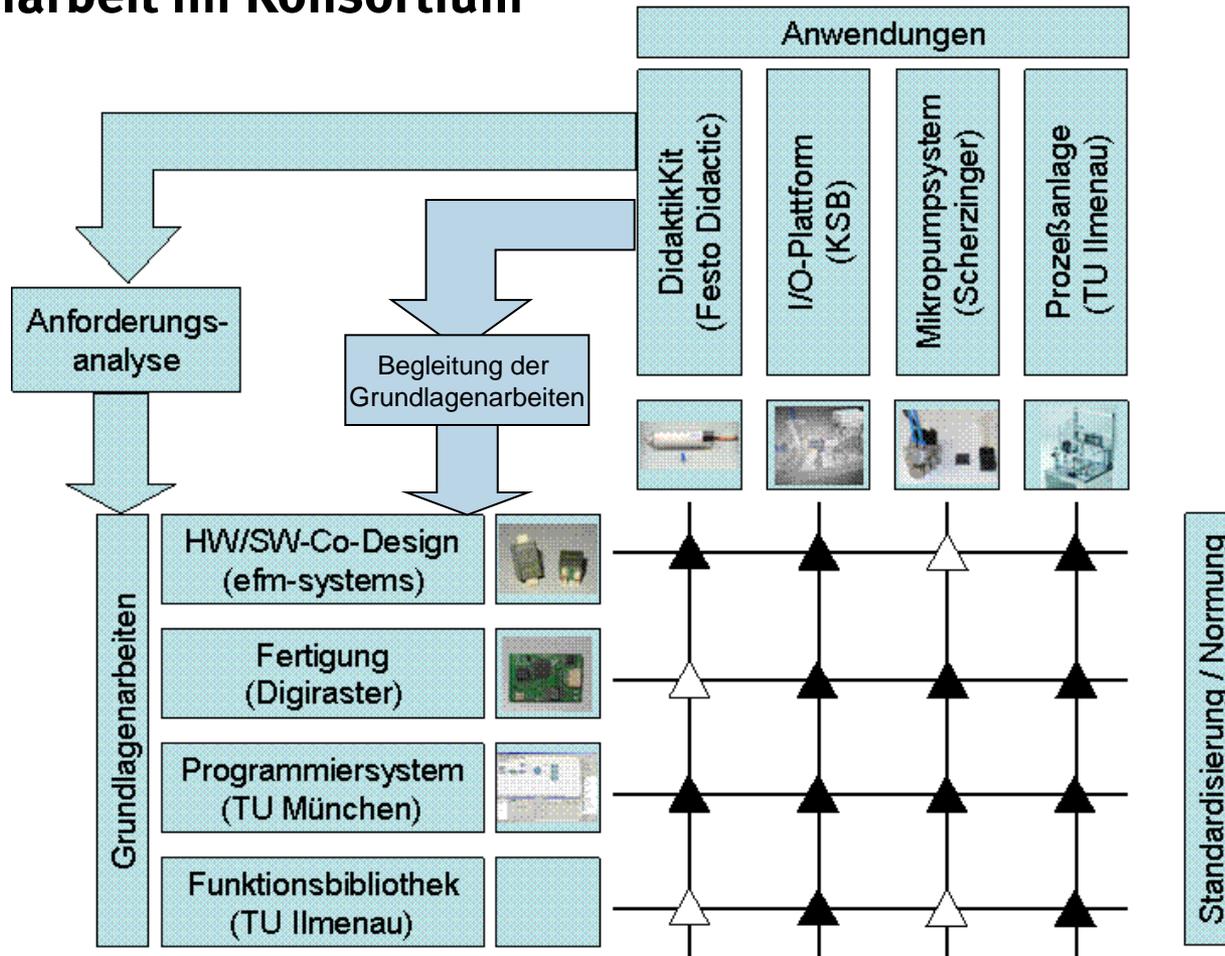
Bundesministerium
für Bildung
und Forschung

BETREUT VOM



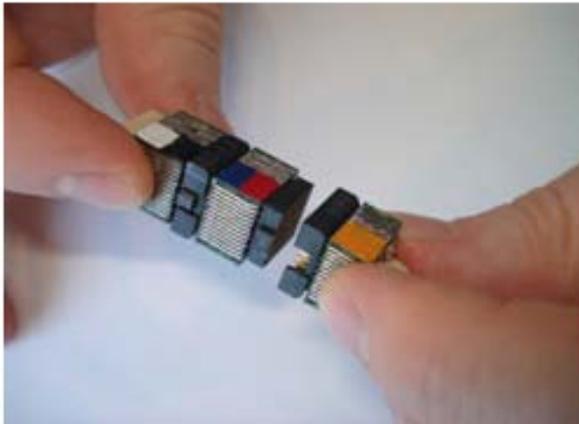
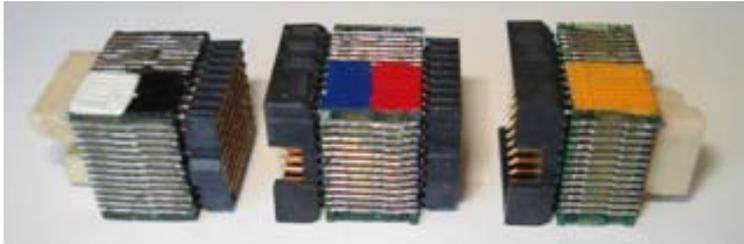
Projektträger
Forschungszentrum
Karlsruhe (PTKA)

Zusammenarbeit im Konsortium

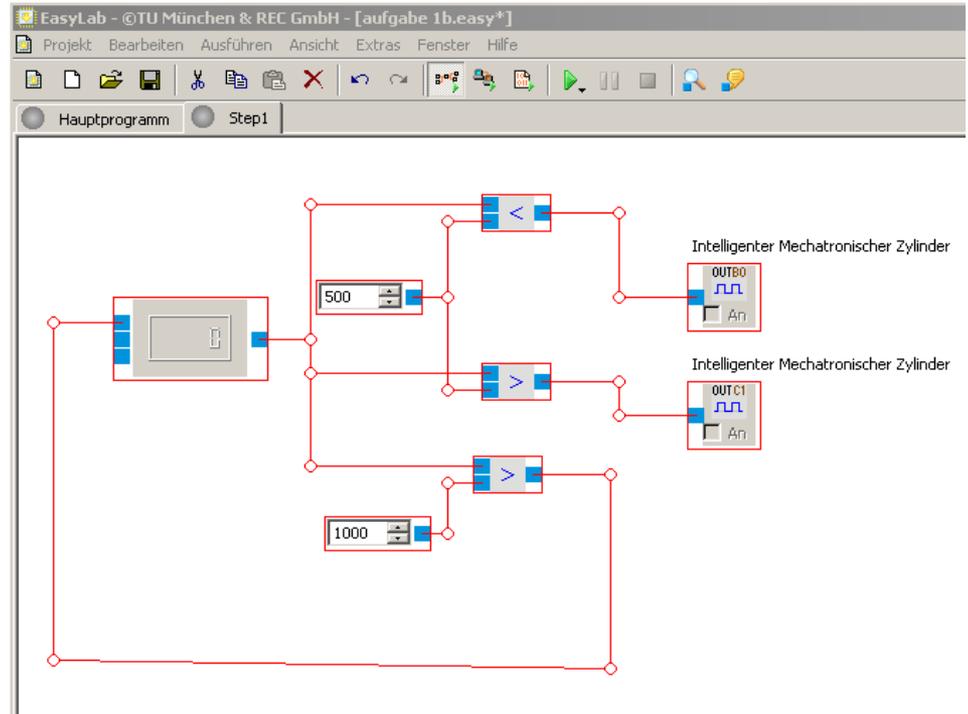


Das Ergebnis

EasyKit – ein Werkzeug für durchgängiges Software-Hardware-Codesign

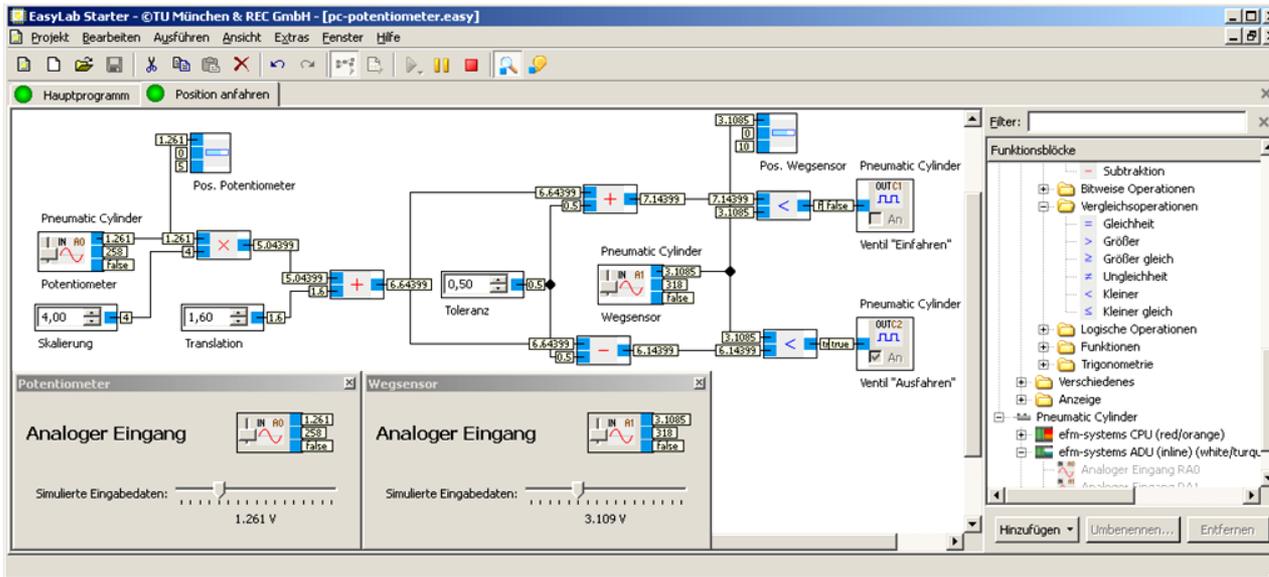


EasyKit-Bausteine



EasyLab - Programmierumgebung

EasyLab



- Komponenten:**
- Simulator
 - Codegenerator
 - Echtzeit-Visualisierung des Programmzustands

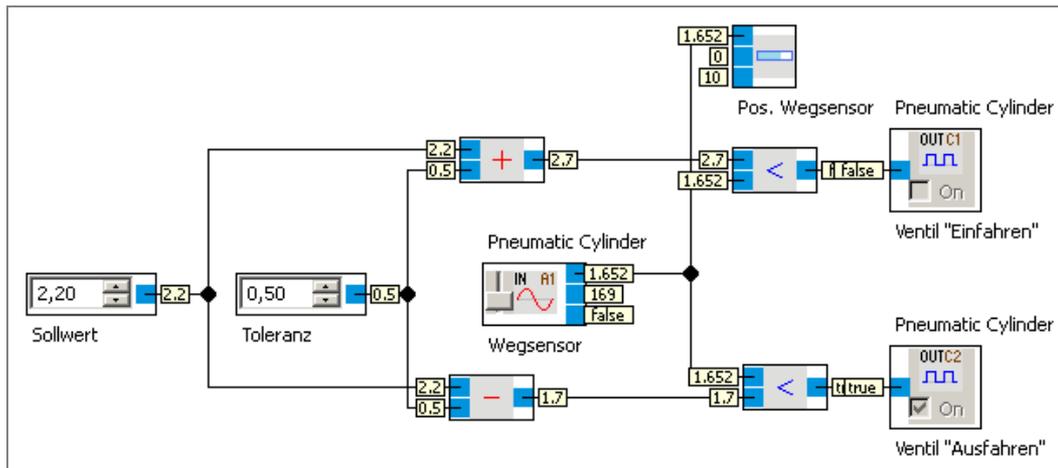
- Modellgetriebene Software-Entwicklung**
- Grafische Programmiersprachen mit hohem Abstraktionsniveau
 - Funktionsblöcke als kleinste Einheiten im Modell
 - Verschaltung von Funktionsblöcken ergibt Programm

- Speziell auf EasyKit-Methodik abgestimmt**
- Funktionsblockbibliothek mit Fokus auf Regelungstechnik und Mechatronik
 - Spezielle Funktionsblöcke zur Ansteuerung der EasyKit-Hardware

EasyLab

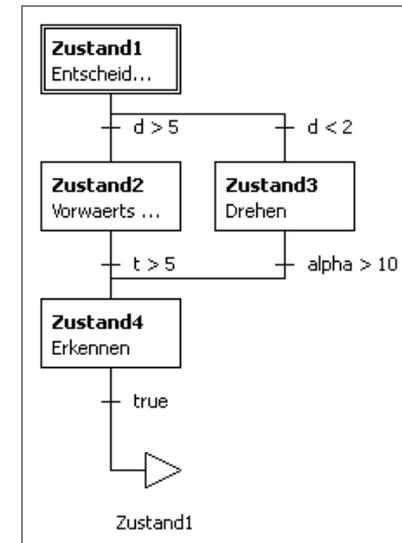
Datenflussprogramm

- Funktionsblöcke mit zugeordneten Code-Templates
- Kanten verbinden
Ein- und Ausgänge
von Funktionsblöcken



Zustandsdiagramm

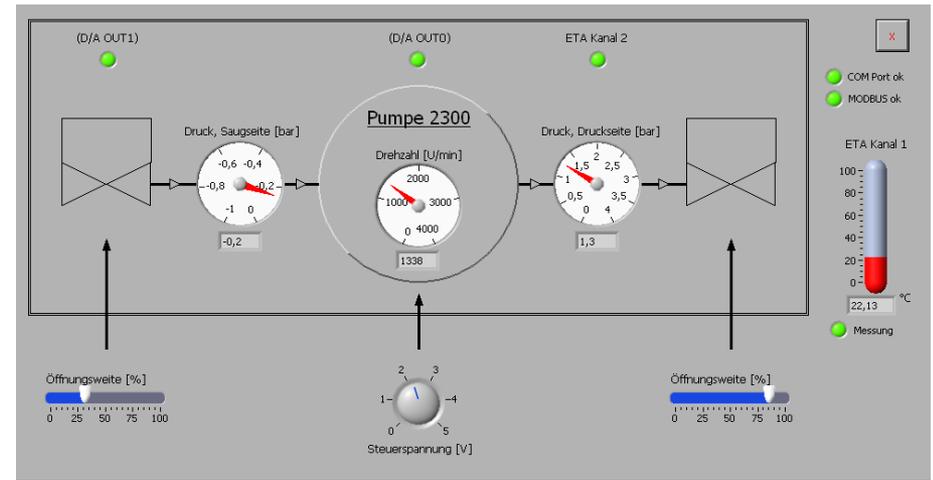
- Zustandssequenzen mit Booleschen Übergangsbedingungen
- Alternative Verzweigungen, Sprünge
- Zustände referenzieren je ein Datenflussprogramm



Anwendung: Pumpen-Prüfplatz

Messe- und Prüfplatz

- 8 Komponenten (Sensoren und Aktoren) über EasyKit Systeme : in einem Netzwerk zusammengeschlossen
- Regelung nach unterschiedlichen Vorgaben (Durchfluss, Druck, Drehzahl) möglich
- Einführung der Mechatronik bei Scherzinger



Anwendung: Intelligente Pumpe

Ziel:

- Entwicklung automatisierte Pumpen mit hoher Funktionsdichte
- Qualitätssicherung durch standardisierten Entwicklungsprozess
- Produktübergreifende Wiederverwendung von Funktionen
- Einsatz standardisierter Hardware

Maßnahmen:

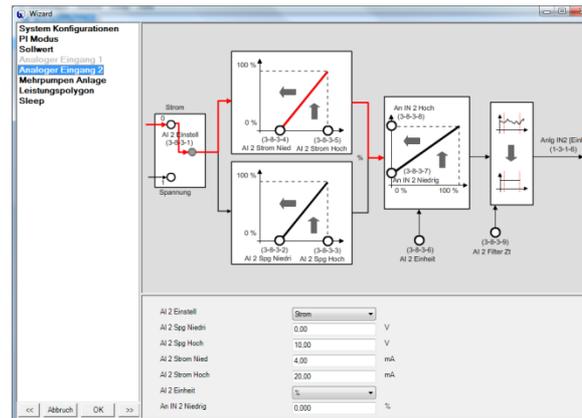
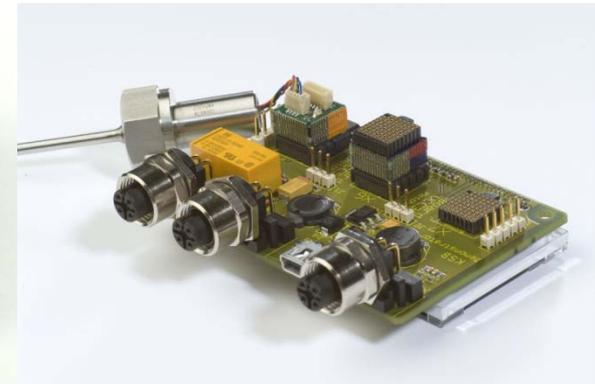
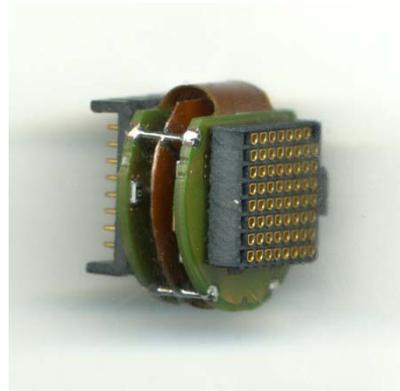
- Entwicklung der „KSB Hydraulik-Lib“ in Easy-Lab
- Entwicklung eines KSB Entwicklungsboards zum Aufbau schneller Prototypen
- Entwicklung von KSB Match-X Bausteinen (Fujitsu CPU, 4-20mA Ein-/Ausgang...)



Anwendung: Intelligente Pumpe

Ergebnisse:

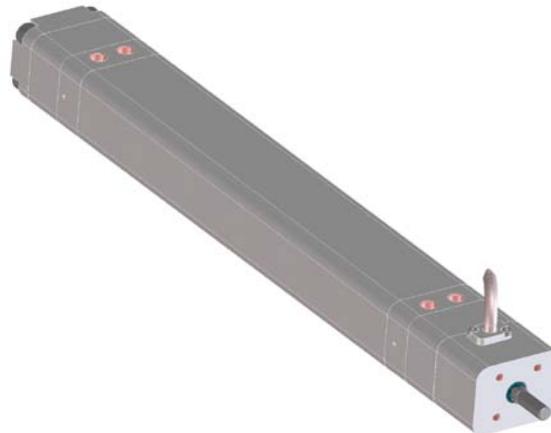
- Integration der KSB Standard Bedieneinheiten
- Integration der KSB Bedienphilosophie in
- EasyLab inkl. KSB-Toolchain
- Anbindung des KSB Servicetools an die mit EaysKit entwickelten Steuerungen



Anwendung: Intelligenter Zylinder

Integration von:

- Zylinder
- Messsysteme
- Ventile
- Steuerung



Problem: viele Varianten

- Hub
- Aufgabe -> Sensorik, Ventile
- Kleine Stückzahlen
- Konfigurierbare Steuerung notwendig



Anwendung: EasyKit Starter

- Lernsystem für Schüler, Auszubildene und Berufseinsteiger
- Basisboard mit leistungsfähigen Prozessor, USB-Schnittstelle und 2 Bausteinsokkeln
- 2 Bausteine
- Anwendungsboard mit LED's, Empfängern und Motor
- EasyLab
- Teachware
- 1000 Stück werden produziert und an Schulen verteilt



Abschlussveranstaltung 11.11.2009

Termin: 11.11.2009

Ort: Festo Berkheim

Anmeldung unter:

Tel: 0711/3467-1247

Fax: 0711/347-54-1247

hor@de.festo.com

Freier Eintritt!

09:00	Anmeldung
09:30	Begrüßung Dr. Reinhard Pittschellis Festo Didactic GmbH & Co. KG Koordinator des Verbundprojektes EasyKit
09:40	Rahmenkonzept – „Forschung für die Produktion von morgen“ Themenfeld „Zuverlässigere mechatronische Systeme“ Dr. Isabella Wieczorek, Bundesministerium für Bildung und Forschung (BMBF)
09:50	EasyKit – Ein Überblick Dr. Reinhard Pittschellis
10:30	EasyKit – Die Hardware Gerd Bauer, efm-systems GmbH
11:15	EasyLab – die Software Prof. Alois Knoll, TU München Prof. Christoph Ament, TU Ilmenau
12:00	EasyKit in der Praxis Rapid Prototyping bei Pumpensteuerungen Dr. Jochen Mades, KSB AG
12:30	Mittag
13:30	Parallelworkshops: Workshop 1 Einführung in EasyKit mit dem Lernsystem mechatronischer Zylinder Workshop 2 Einführung in EasyKit mit dem EasyKit Starter von Festo Parallel zum Workshop: Werksführung und Besuch des Festo Forums
16:00	Abschlussdiskussion
17:00	Ende der Veranstaltung

Vielen Dank für Ihre Aufmerksamkeit!