



Einführung Echtzeitsysteme

Echtzeitsysteme im Alltag



Zentralübung - Hintergrund

- Ziele:
 - Klärung offener Fragen
 - Anwendung der gelernten Inhalte
 - Ansprechpartner für Probleme
- Methodik:
 - Interaktive Gestaltung: Ihre Mitwirkung ist wichtig
 - Hands-On Übungen: Werkzeuge, Programmierung
 - Durchsprache von Klausuraufgaben



Organisatorisches

- Wegen der erfreulich hohen Zahl an Übungsteilnehmern (bei gleichzeitig ungleicher Verteilung auf die vorgeschlagenen Übungstermine) muss der Zuteilungsprozess für die Übungsgruppen geändert werden. Bitte senden Sie sortiert nach Ihrer Priorität drei der folgenden Übungstermine an buckl@fortiss.org
- Termin 1: Montag 08:30-10:00 Uhr
- Termin 2: Montag 10:15-11:45 Uhr
- Termin 3: Montag 14:00-15:30 Uhr
- Termin 4: Montag 15:30-17:00 Uhr
- Termin 5: Dienstag 08:30-10:00 Uhr
- Termin 6: Dienstag 14:00-15:30 Uhr
- Termin 7: Dienstag 15:30-17:00 Uhr
- Termin 8: Mittwoch 08:30-10:00 Uhr
- Termin 9: Mittwoch 10:15-11:45 Uhr
- Termin 10: Mittwoch 14:00-15:30 Uhr
- Termin 11: Mittwoch 15:30-17:00 Uhr
- Termin 12: Donnerstag 08:30-10:00 Uhr
- Termin 13: Donnerstag 10:15-11:45 Uhr
- Termin 14: Freitag 08:30-10:00 Uhr
- Termin 15: Freitag 10:15-11:45 Uhr
- Termin 16: Freitag 14:00-15:30 Uhr
- Termin 17: Freitag 15:30-17:00 Uhr



Echtzeitsysteme sind allgegenwärtig!





Beispiel: Kuka Robocoaster



<http://www.robocoaster.com>



Einleitung Echtzeitsysteme

Anwendungen am Lehrstuhl / fortiss



Steuerungsaufgaben (Praktika+Studienarbeiten)



Regelungsaufgaben (Praktika+Studienarbeiten)



Schwebender Stab



Produktionstechnik

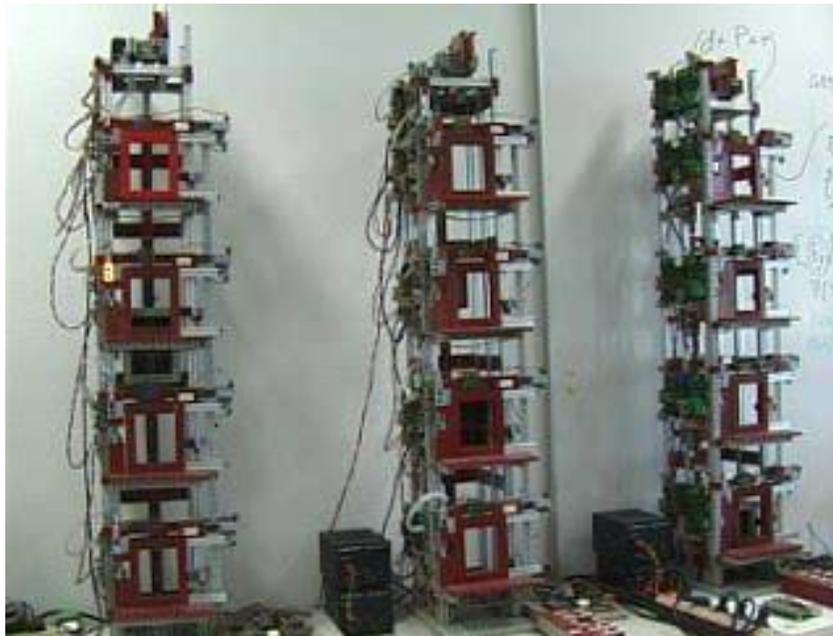


Invertiertes Pendel

Elektroauto



Videos



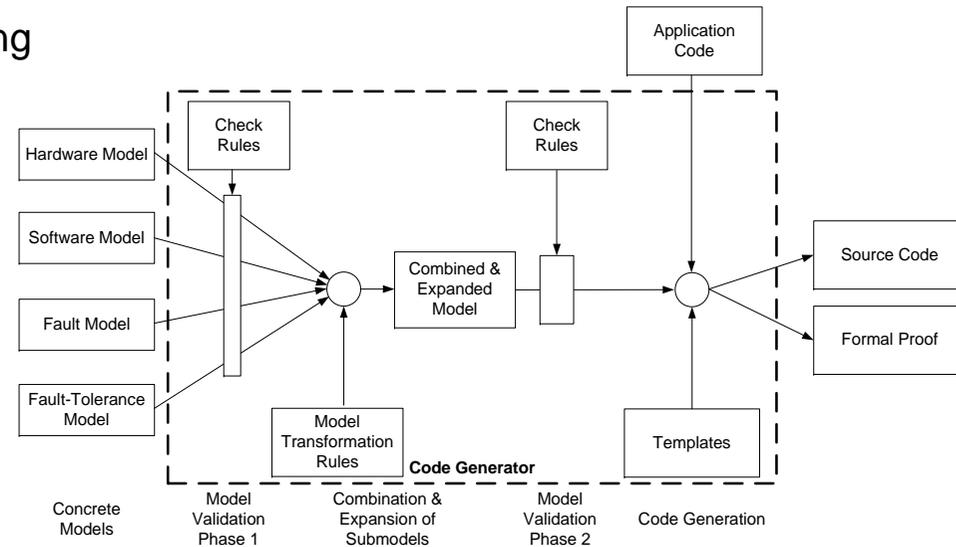
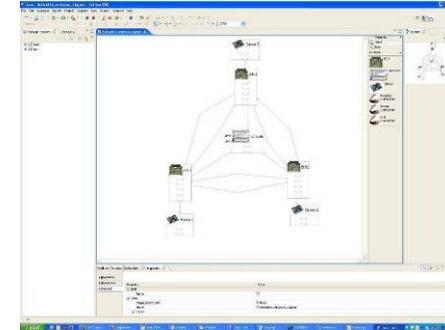
Software entwickelt mit FTOS



Software entwickelt mit EasyLab

FTOS: Modellbasierte Entwicklung fehlertoleranter Echtzeitsysteme

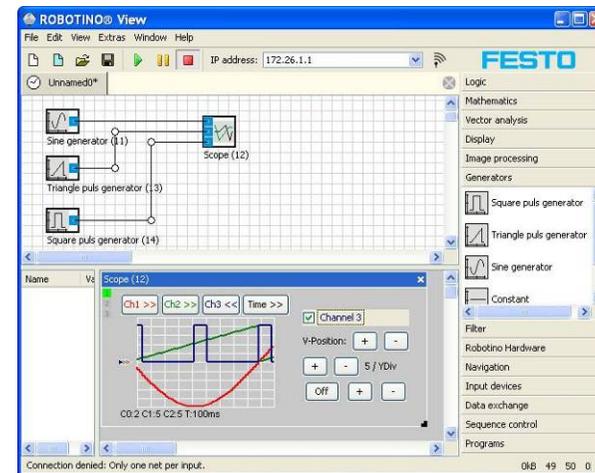
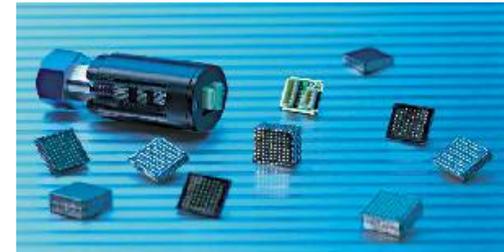
- Ziele:
 - Umfangreiche Generierung von Code auf Systemebene:
 - Fehlertoleranzmechanismen
 - Prozessmanagement, Scheduling
 - Kommunikation
 - Erweiterbarkeit der Codegenerierung durch Verwendung eines vorlagenbasierten Codegenerators
 - Zertifizierung des Codegenerators





Modellbasierte Software-Entwicklung für mechatronische Systeme

- Entwicklung von komponentenbasierten Architekturen für mechatronische Systeme (Mechanik, Elektronik, Software)
- Ziele:
 - Reduzierung der Entwicklungszeiten
 - Vereinfachung des Entwicklungsprozesses
- Komponenten:
 - Hardwaremodule
 - Softwaremodule
 - Werkzeugkette:
 - Codegenerierung
 - Graphische Benutzerschnittstelle
 - Debugging-Werkzeug



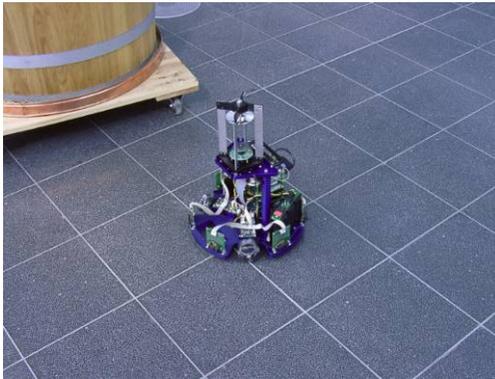
GEFÖRDERT VOM



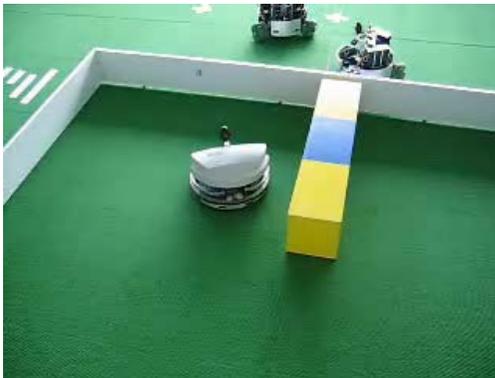
Bundesministerium
für Bildung
und Forschung



Robotersteuerung



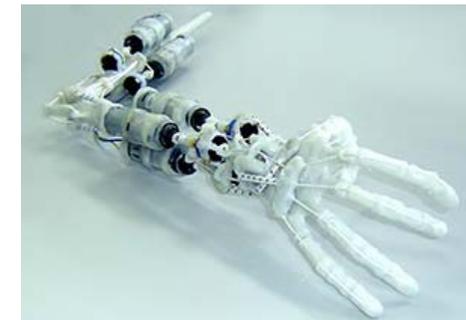
Robotino



Leonardo



Staubli



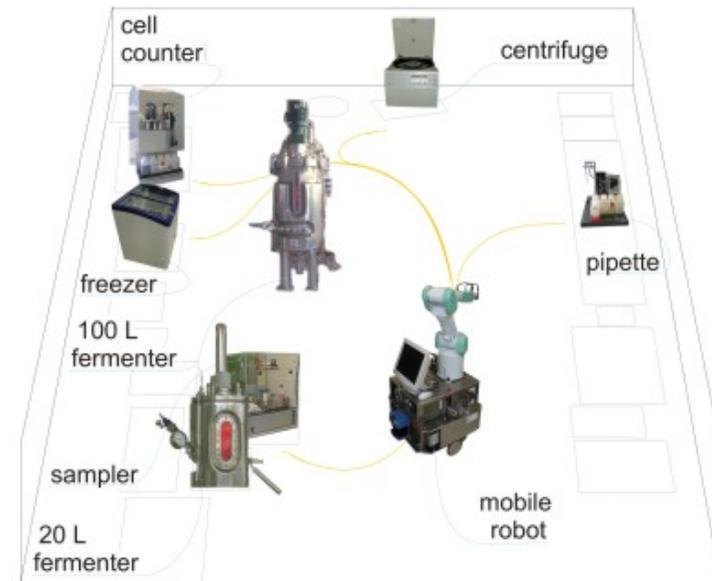
Tumanoid

Anwendungen der Robotik



Telemedizin

Jast



*Automatisiertes
biotechnisches Labor*



Automatisiertes Labor





Klausurfragen

- Klausur WS 06/07
 - Was ist der Unterschied zwischen harten und weichen Echtzeitsystemen?
 - Wieso sollte Virtual Memory nicht in Echtzeitsystemen verwendet werden?
- Wiederholungsklausur WS 06/07
 - Ordnen Sie folgende Anwendungen in die Kategorien harte bzw. weiche Echtzeitsysteme ein und begründen Sie Ihre Antwort:
 - Ampelsteuerung
 - Flugzeugregelung
 - Internettelefonie