

# Echtzeitsystem für einen zweibeinigen Roboter mit adaptiver Bahnplanung

PEARL Workshop 2007

Boppard

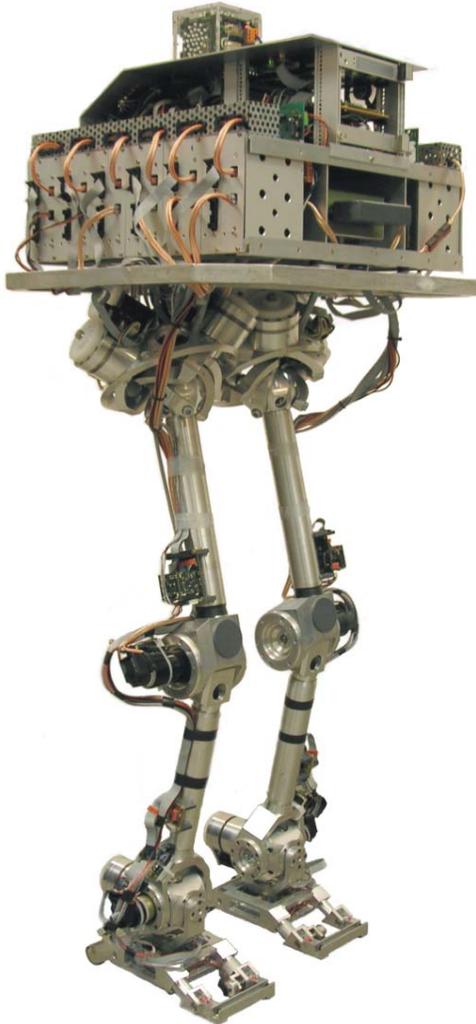
Dipl.-Ing. M. Seebode

Prof. Dr.-Ing. W. Gerth

Institut für Regelungstechnik – Leibniz Universität Hannover

<http://www.irt.uni-hannover.de>

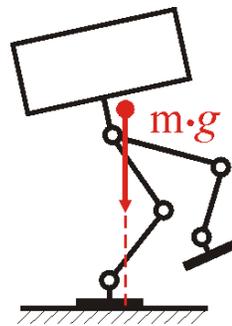
- Zweibeinige Laufmaschine LISA
- Regelung
- Bahnplanung
- Software
- Mikrocontroller und CAN
- Prozessormodul
- FPGA mit OpenCores
- Ergebnisse / Anwendung
- Zusammenfassung / Ausblick



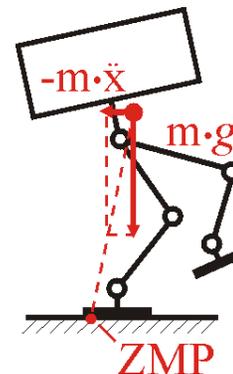
- Versuchsroboter
- Gewicht: 38 Kg, Größe: 140cm
- Parallelkinematisches Hüftgelenk
- 6 aktive Gelenke je Bein
- Kraft-/Momentensensoren
- Inertialsensor
- Autonomes System
  - Leistungselektronik, Rechner und Akkus on-board
  - Compact-PCI-Backplane (3U)

- Servoregelung
  - Soll-Koordinaten der Füße → Soll-Gelenkwinkel (Kinematik)
  - Positionsregelung jeder Achse

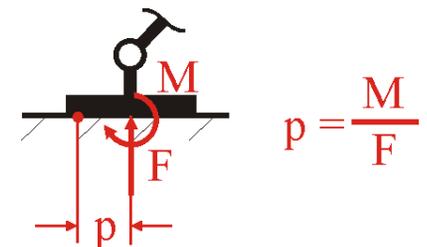
- Servoregelung
  - Soll-Koordinaten der Füße → Soll-Gelenkwinkel (Kinematik)
  - Positionsregelung jeder Achse
- Gleichgewichtsregelung
  - Beispiel: ZMP-Regelung
  - Wann bleibt der Standfuß am Boden?



statisch: Schwerpunkt  
über Fußaufstandsfläche

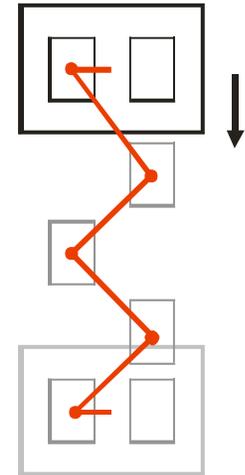


dynamisch: Existenz des  
Zero-Moment-Points

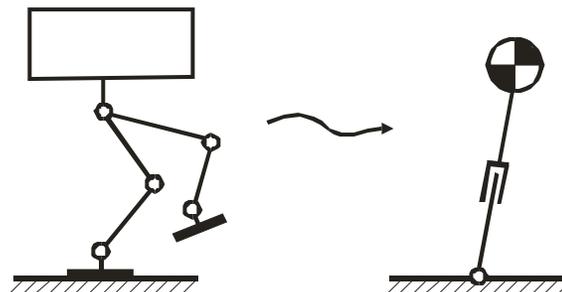


- ZMP (Druckmittelpunkt), Kriterium nur hinreichend
- Messung des ZMP mit Kraft-/Momentensensor

- **Statisch stabile Trajektorien**
  - Schwerpunkt stets über Fußaufstandsfläche
  - Sehr einfach
  - Voraussetzung: Geringe Beschleunigungen
  - Nur für sehr langsame Bewegungen geeignet



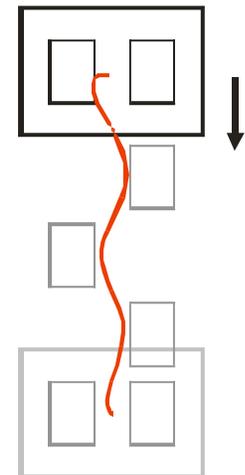
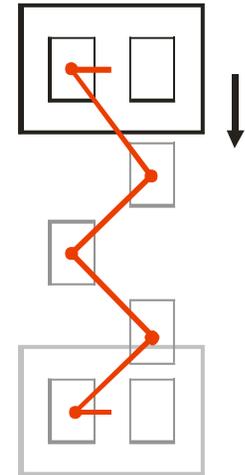
- **Statisch stabile Trajektorien**
  - Schwerpunkt stets über Fußaufstandsfläche
  - Sehr einfach
  - Voraussetzung: Geringe Beschleunigungen
  - Nur für sehr langsame Bewegungen geeignet
- **Dynamisch stabile Trajektorien**
  - Möglicher Ansatz: ZMP in der Aufstandsfläche
  - Hohe Beschleunigungen möglich
  - rechenintensiv → Ersatzmodell: Inverses Pendel



$$p = \frac{M}{F}$$

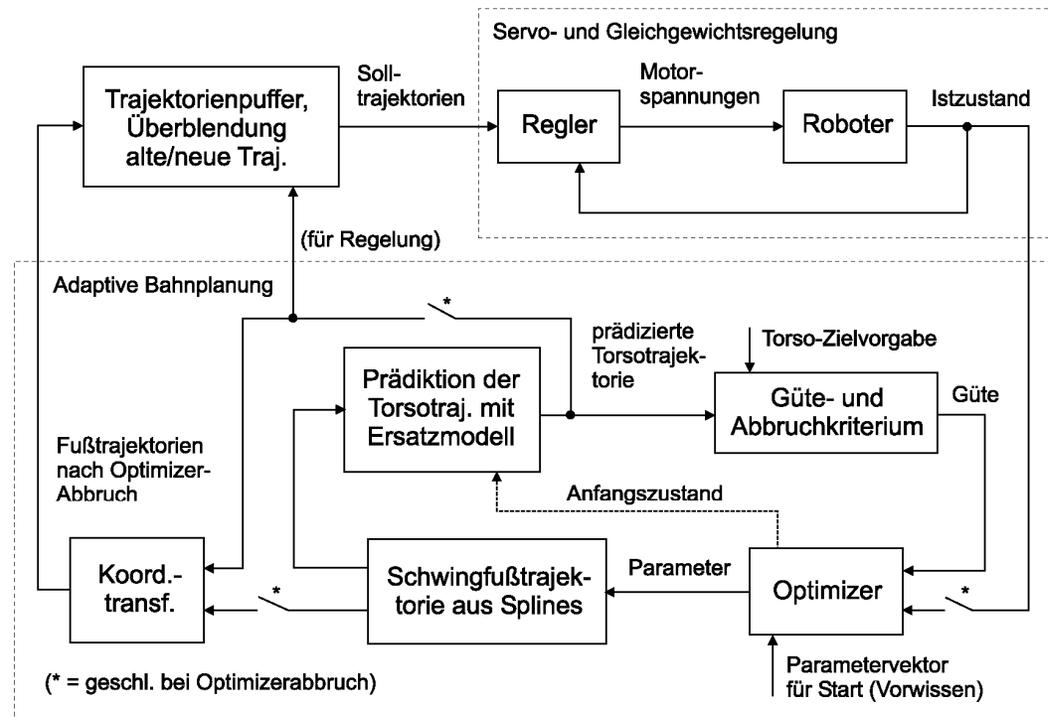
Passives Drehgelenk im Fuß  
=> ZMP bzw. Bodenreaktionsmoment in Planung genullt!

- Kein Abrollen wie bei menschlichem Gang



- Adaptive/Reaktive Bahnplanung

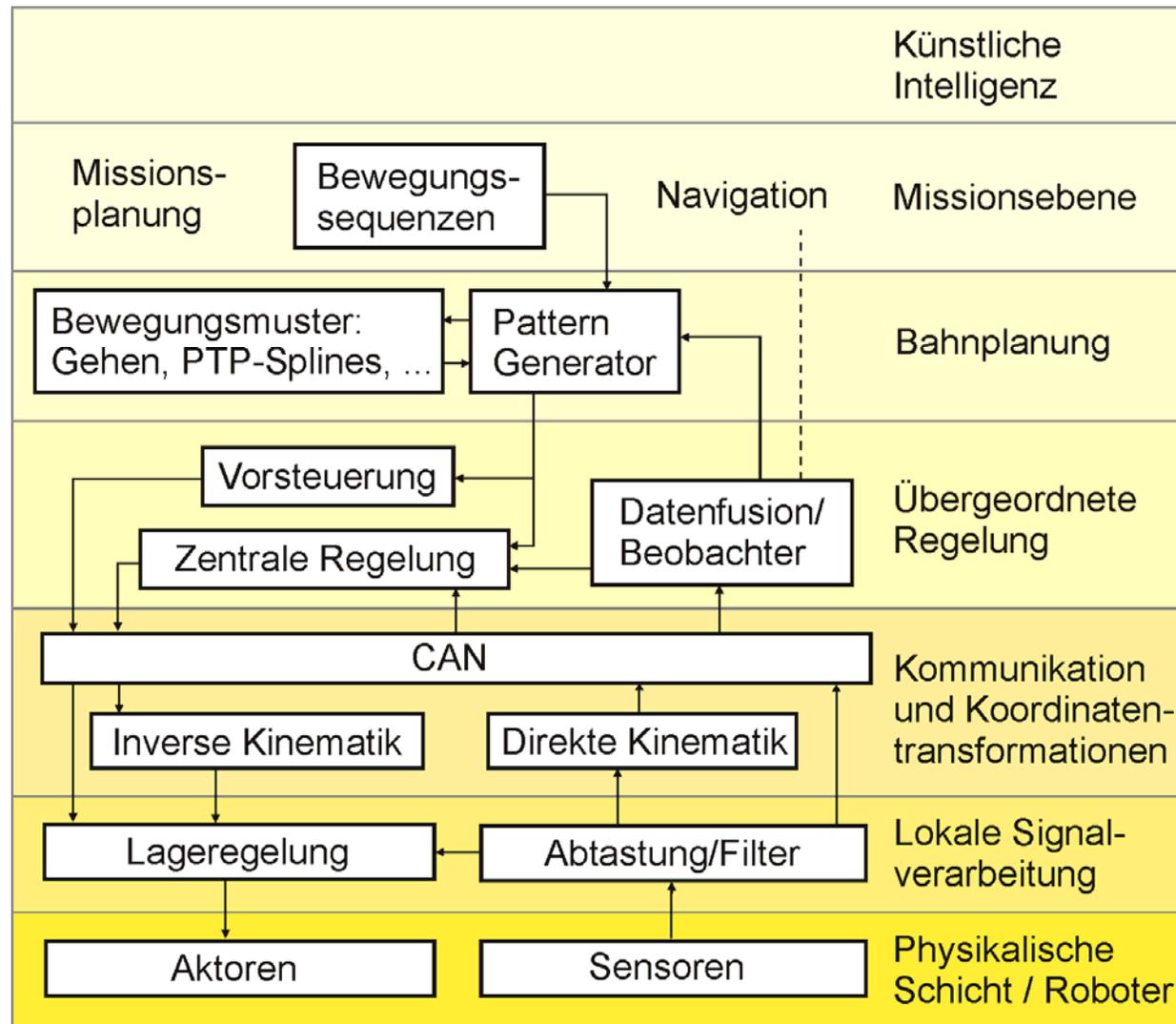
- Gleichgewichtsregelung kompensiert nur kleine Störungen
- Ansatz zur Erzeugung neuer Trajektorien zur Laufzeit:

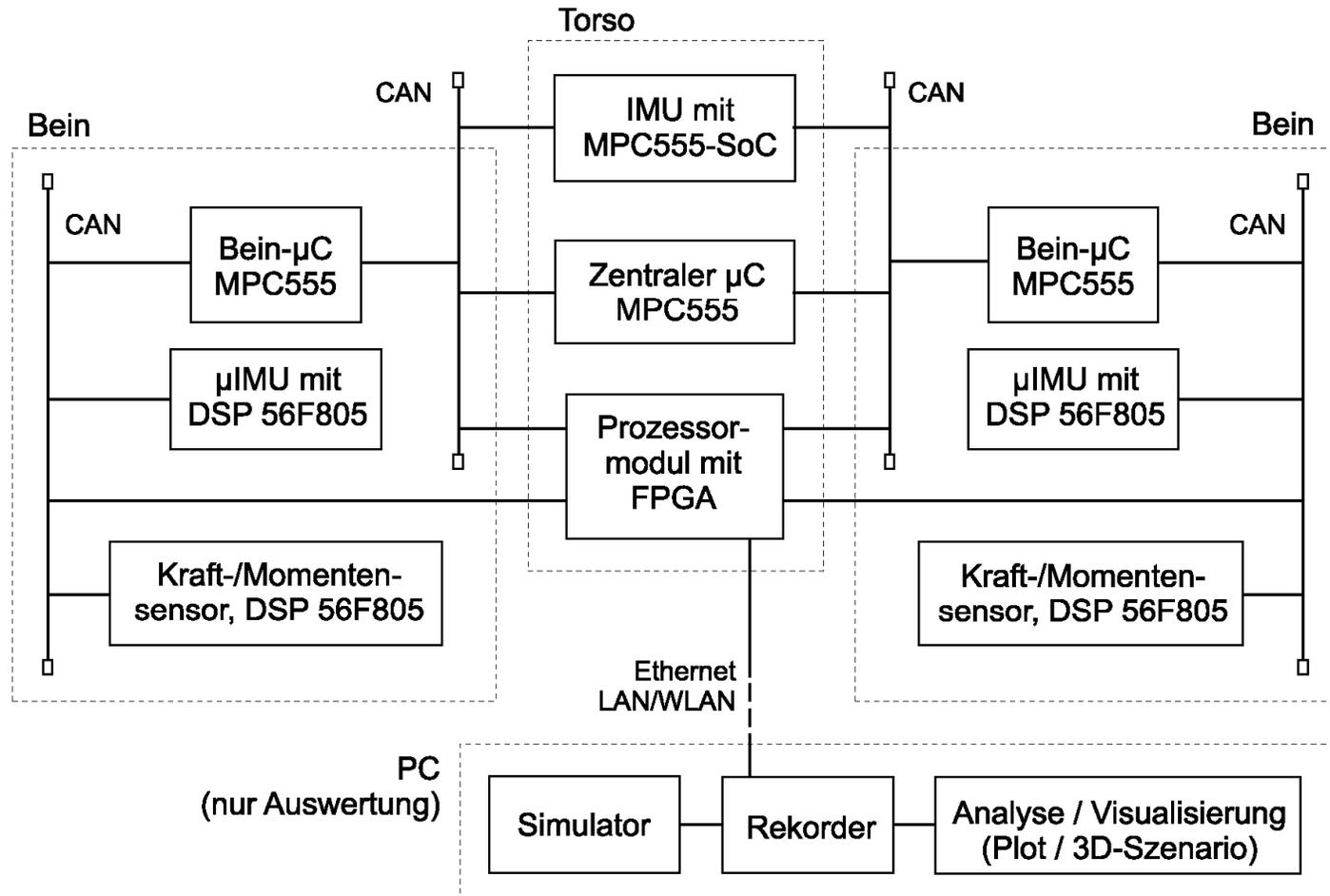


- Simulation

- Windows-PC, Pentium 4 mit 3 GHz: 80 ms (Horizont: 2 Schritte)
- Auf dem Roboter mit G4-PowerPC 1 GHz: 120 ms

- Zweibeinige Laufmaschine LISA
- Regelung
- Bahnplanung
- **Software**
- Mikrocontroller und CAN
- Prozessormodul
- FPGA mit OpenCores
- Ergebnisse / Anwendung
- Zusammenfassung

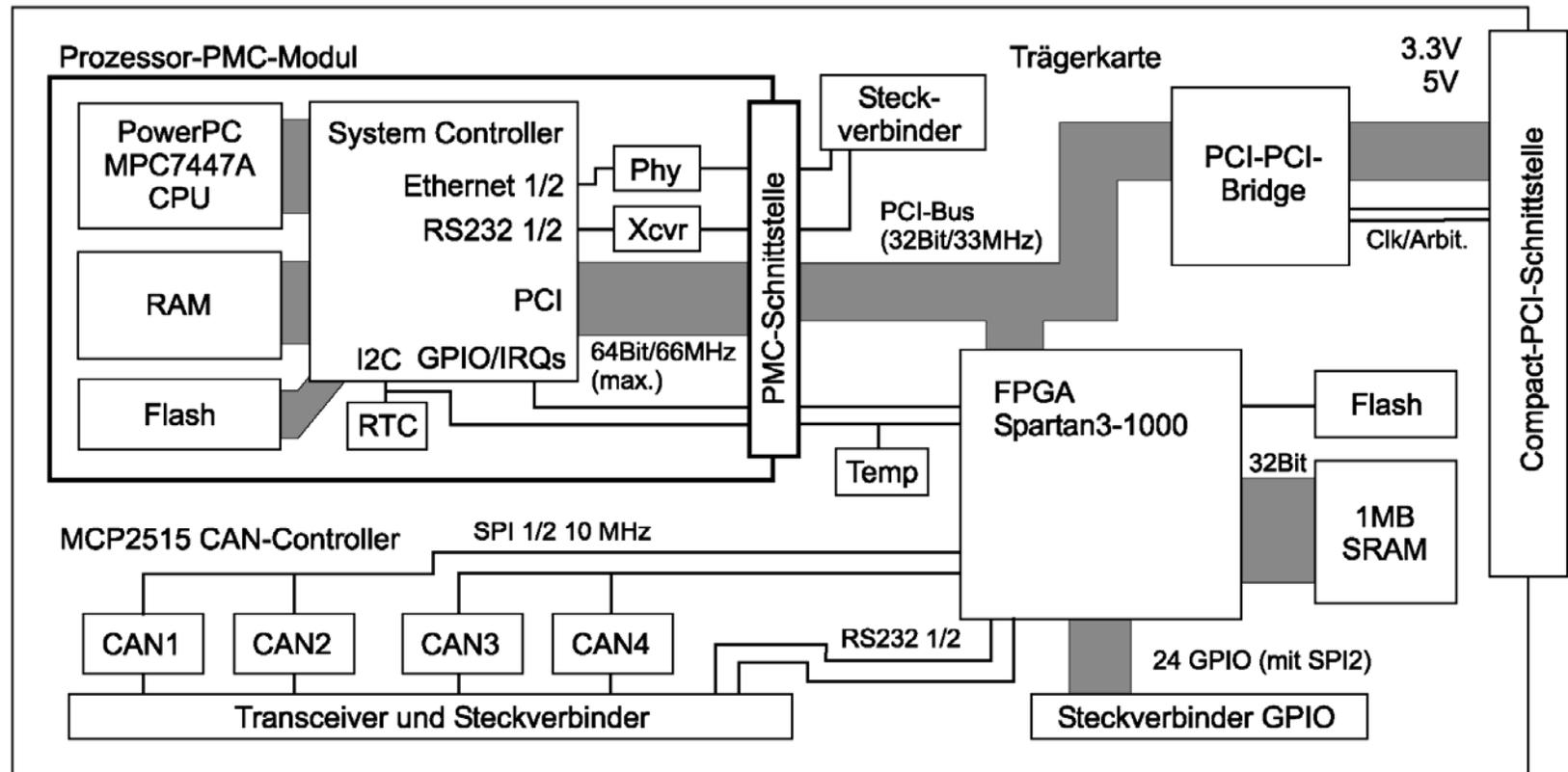




Prozessormodul?

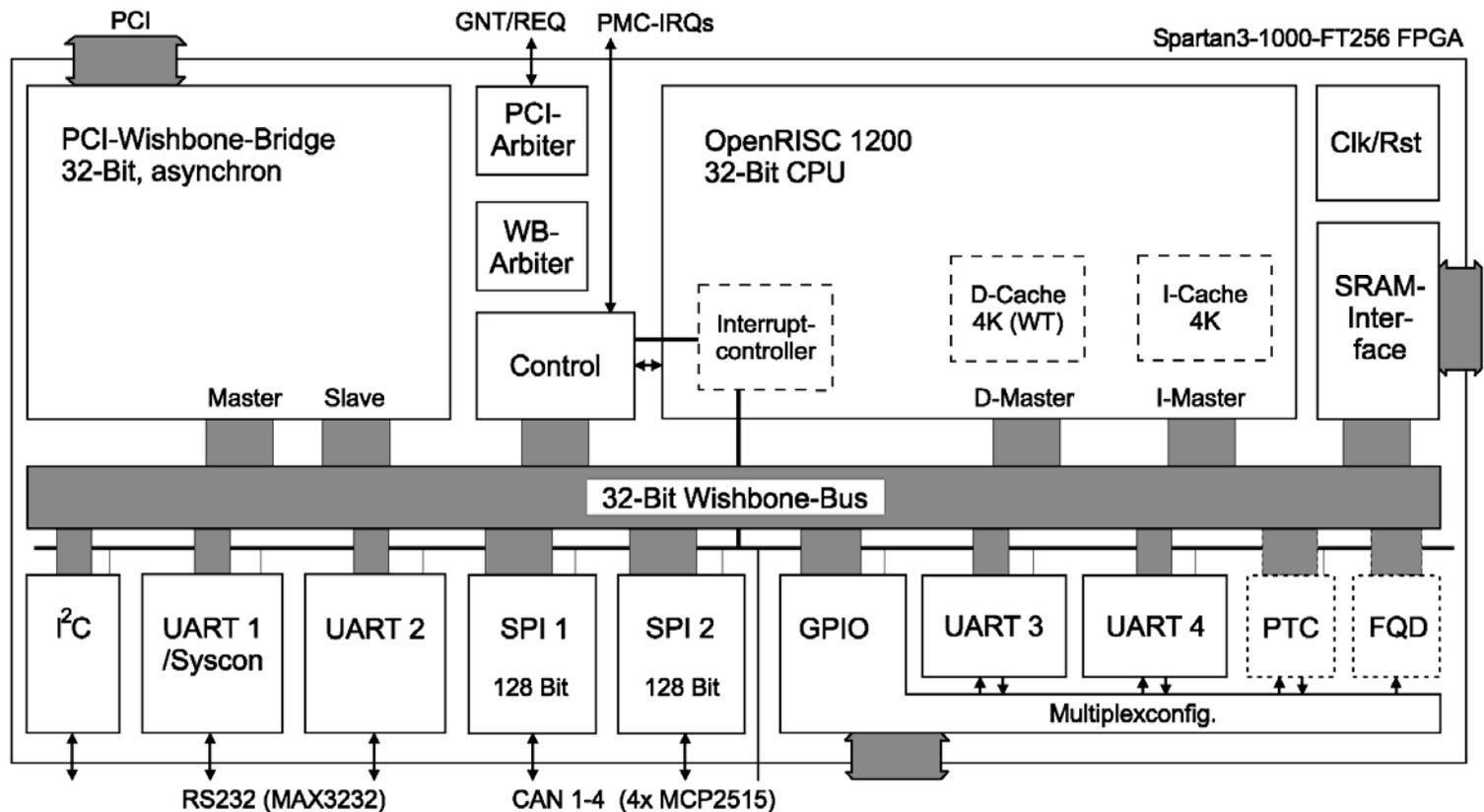
- Prozessormodul
  - PMC-Modul
  - Kompaktes Format, PCI Mezzanine Card (149 mm x 74 mm)
  - Austauschbar / Rechenleistung skalierbar
  - PowerPC MPC7447A mit 1GHz (G4-Prozessor, 32 Bit, AltiVec)
  - Echtzeitbetriebssystem RTOS-UH
- Prozessor-PMC-Trägerkarten für Roboter?
  - nur Karten mit sechs Höheneinheiten erhältlich
  - meist keine passende IO, weitere Karten erforderlich
- Eigene Entwicklung
  - sollte universell einsetzbar sein => Trägerkarte + FPGA
  - Leistungsfähiger Prozessor mit passender IO kompakt in nur 3 Höheneinheiten (160 mm x 100 mm)
  - Betrieb als standalone FPGA- oder Slave-IO-Karte denkbar

- Prozessor-PMC mit Trägerkarte

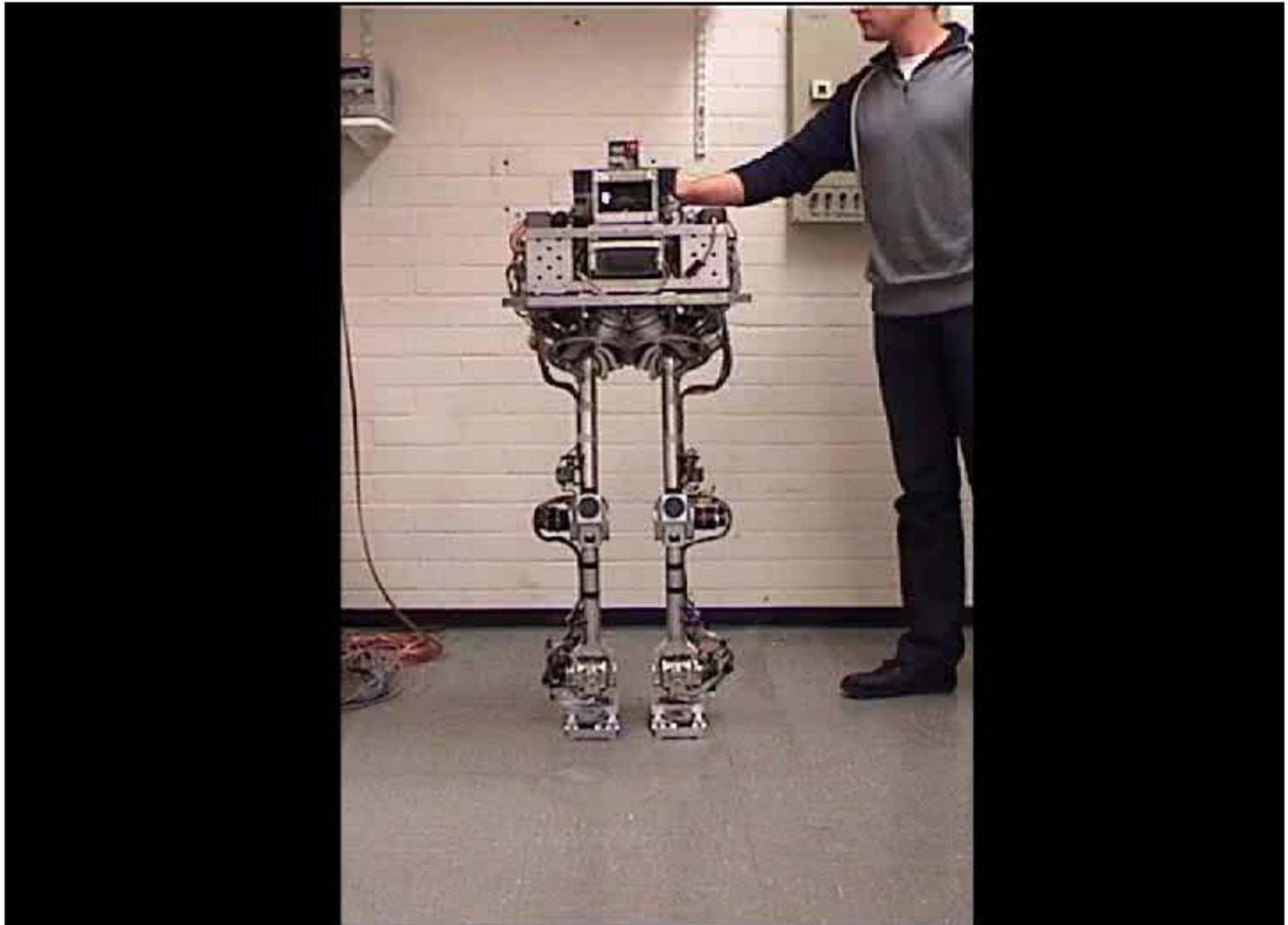


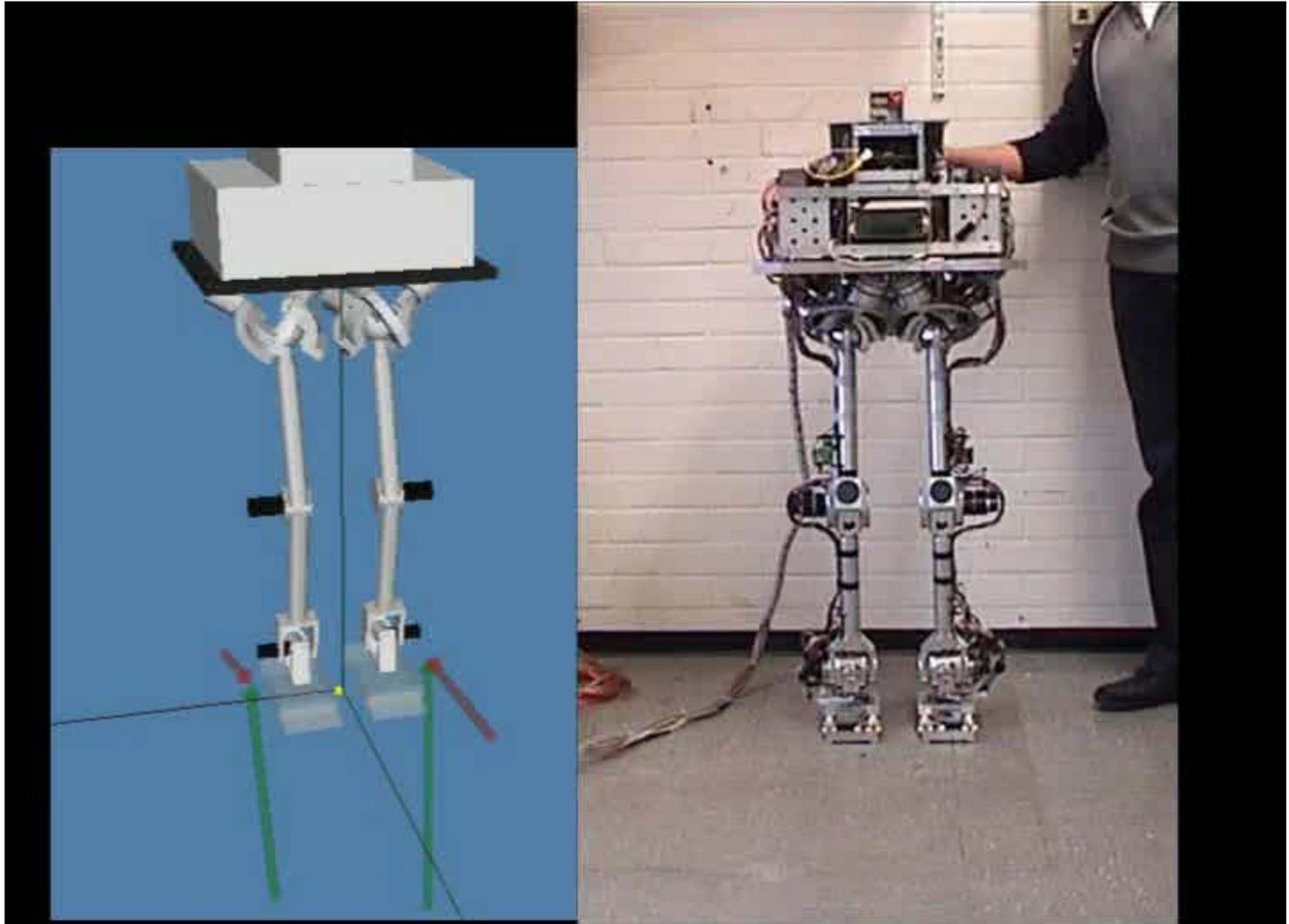
- Resetlogik und PCI-Arbiter für PMC
- CAN-Anbindung
- FPGA
  - Preisklasse: 30 EUR
  - Programmierung?

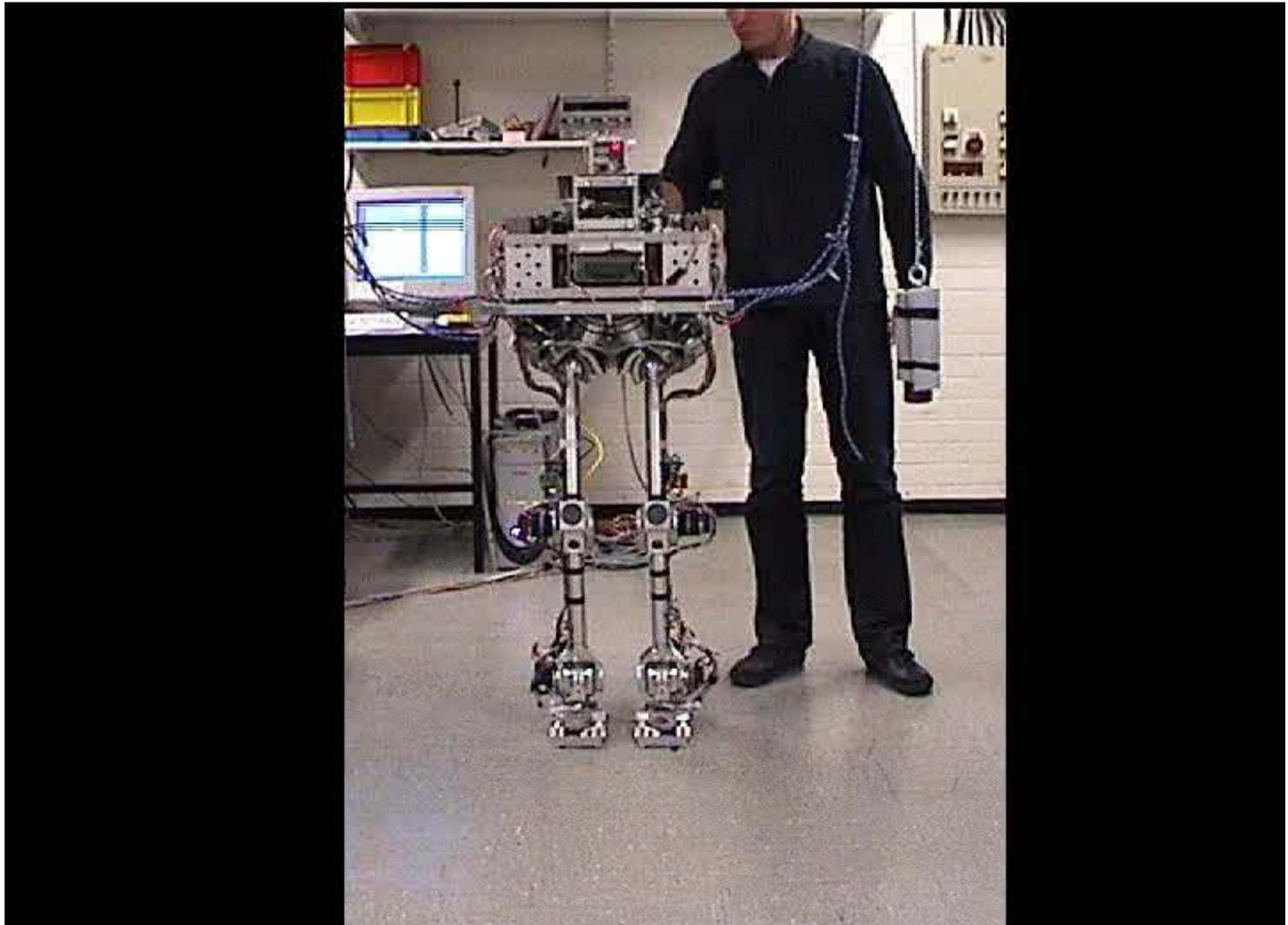
- Open-Source Verhaltensmodelle (Kerne) von OpenCores.org
- Wishbone-Bus/-Spezifikation
- OpenRISC: 32 Register / 1 Condition-Flag, erweiterbar

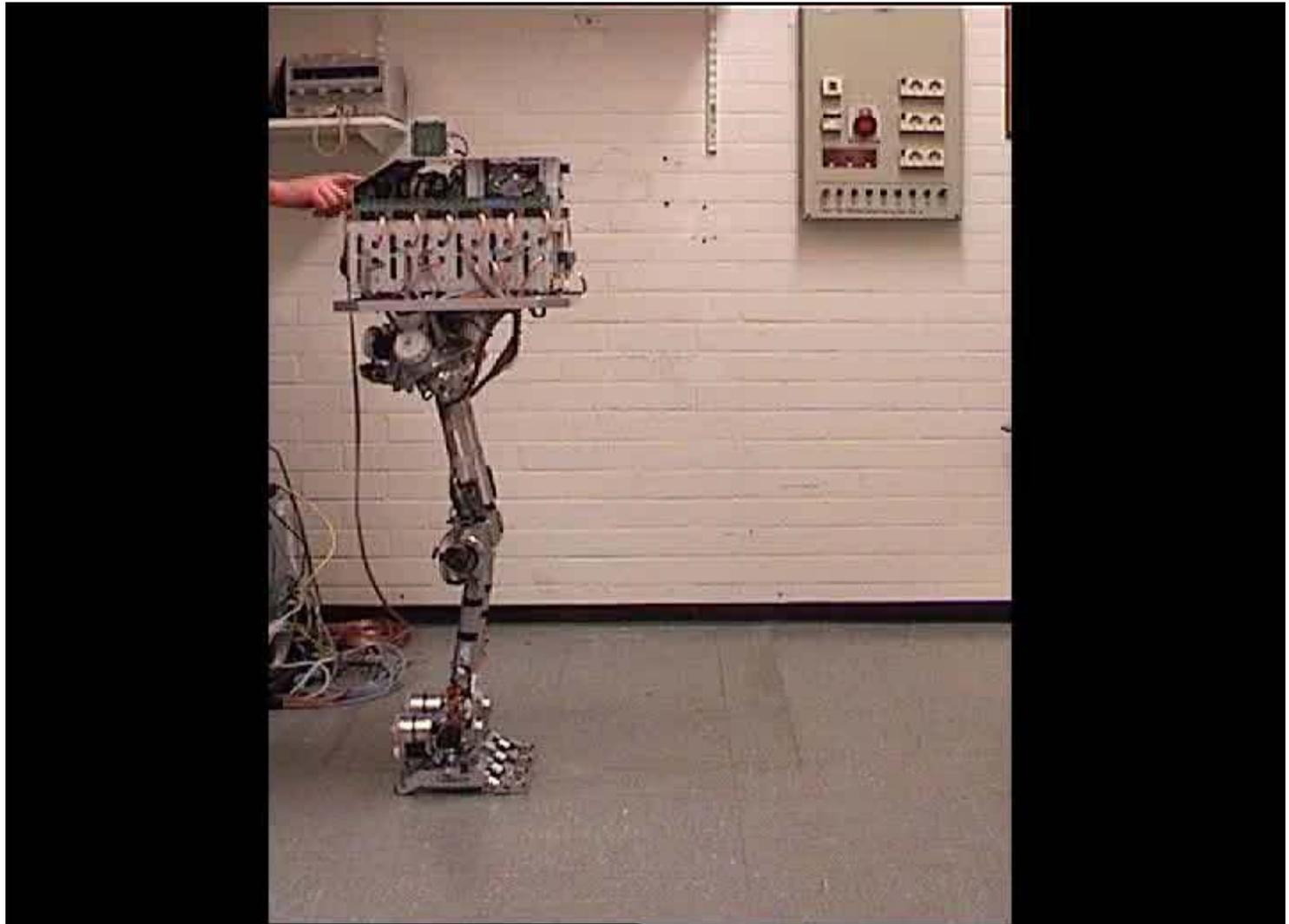


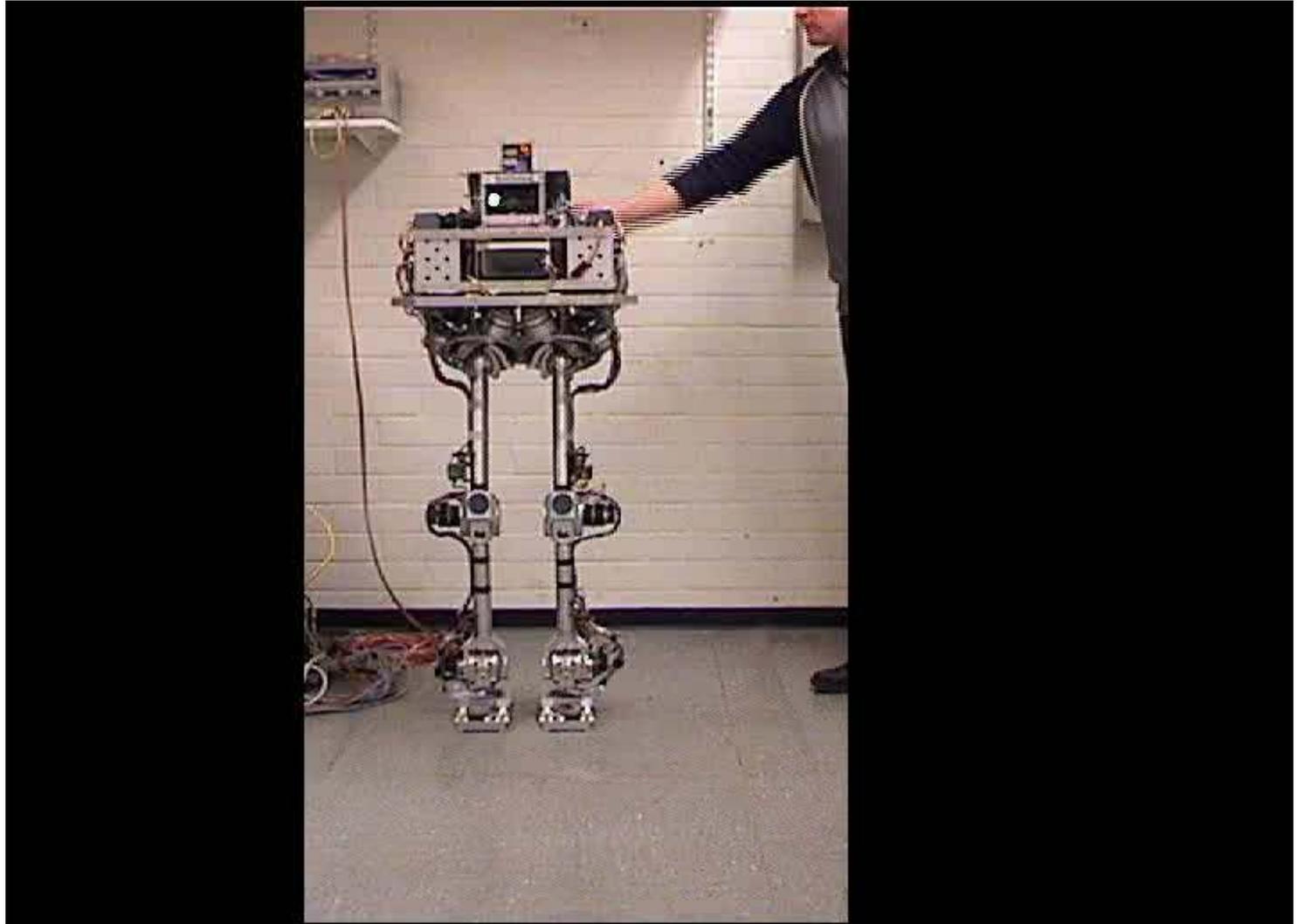
- Test zum Echtzeitverhalten
  - Echtzeitbetriebssystem RTOS-UH mit Zeitatom 50  $\mu$ s
  - Robotersoftware mit 53 Tasks als Anwendung  
(kein wohl definiertes Benchmark-Szenario)
  - L2-Cache aktiviert (512KB)
  - Zeitmessung mit Clock-Counter der CPU (Performance-Monitor)
  - Kontextwechsel: 130ns bis 550ns
  - Reaktionszeit ohne PCI: 1.8 $\mu$ s  
(Auslösen eines GPIO-Interrupts am Chipsatz bis Taskanlauf)
- Vektoreinheit (AltiVec)
  - SIMD-Einheit (Single Instruction Multiple Data)
  - 128 Bit Operanden (z.B. 4 Floats)
  - 4x4-Matrix-Multiplikation
  - C-Code: 260ns FPU, 31ns AltiVec (gcc 4.2,-O3 und -march)
  - Assembler: 189ns FPU, 26ns AltiVec



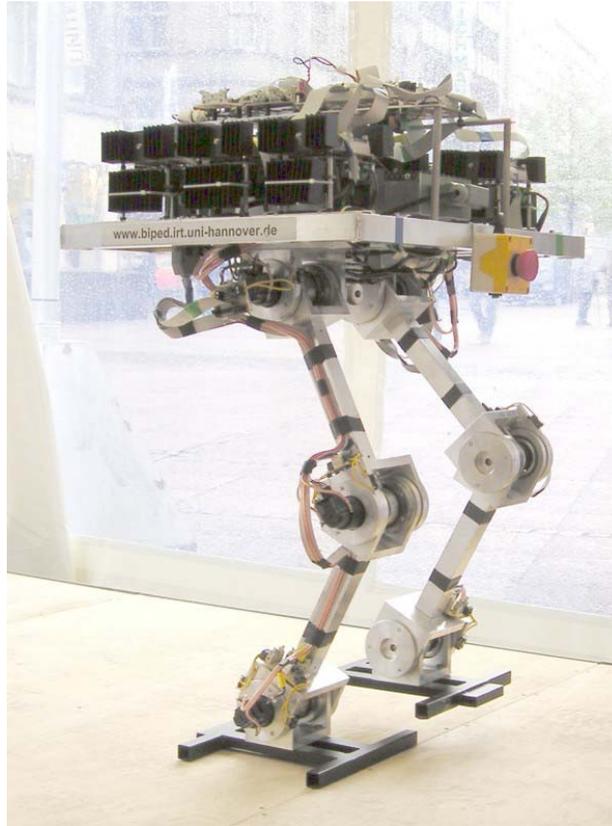








- BARt-UH 1999 - 2003



**IRT** Institute of Automatic Control  
Leibniz University of Hannover

**statically stable  
lie down and get up  
motions of BART-UH**

---

Dipl.-Ing. O.Höhn

hoehn@irt.uni-hannover.de

**IRT** Institute of Automatic Control  
Leibniz University of Hannover

# Dynamical Get Up Motion of BART-UH

---

Dipl.-Ing. O.Höhn

hoehn@irt.uni-hannover.de

- **Zweibeiniger Roboter LISA**
  - Gleichgewichtsregelung weiterentwickeln
  - Flüssige Bewegungen mittels dynamisch stabiler Trajektorien
  - Erzeugung neuer, angepasster Gangmuster zur Laufzeit
- **Prozessor-PMC-Modul**
  - FPGA-Trägerkarte => kompakte Lösung mit IO
  - Echtzeitbetriebssystem RTOS-UH
  - Einsatz in LISA und BART
  - Prozessor-PMC-Formfaktor? Computer on Modules (CoM)?
- **FPGA**
  - OpenCores für Forschung nützlich
  - Synthese des Modells bei knappen FPGA-Ressourcen zeitraubend (Optimierungsaufwand)
  - Für möglichst universelle Leiterkarte: Neben FPGA auch  $\mu\text{C}$ !

# Vielen Dank für Ihre Aufmerksamkeit!

PEARL Workshop 2007

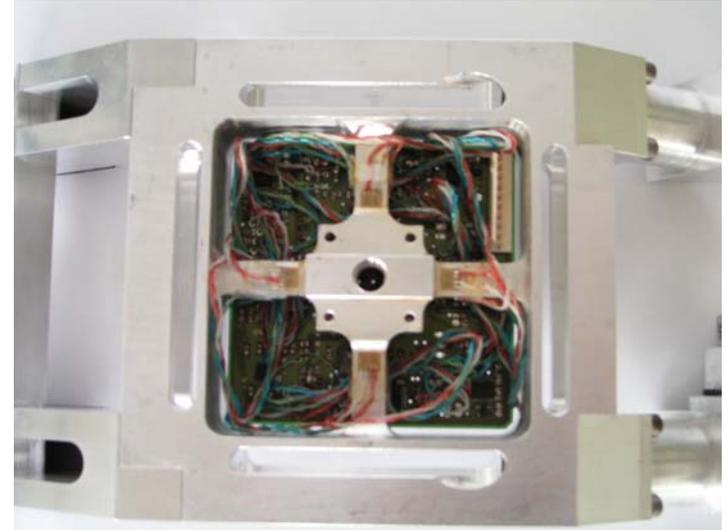
Boppard

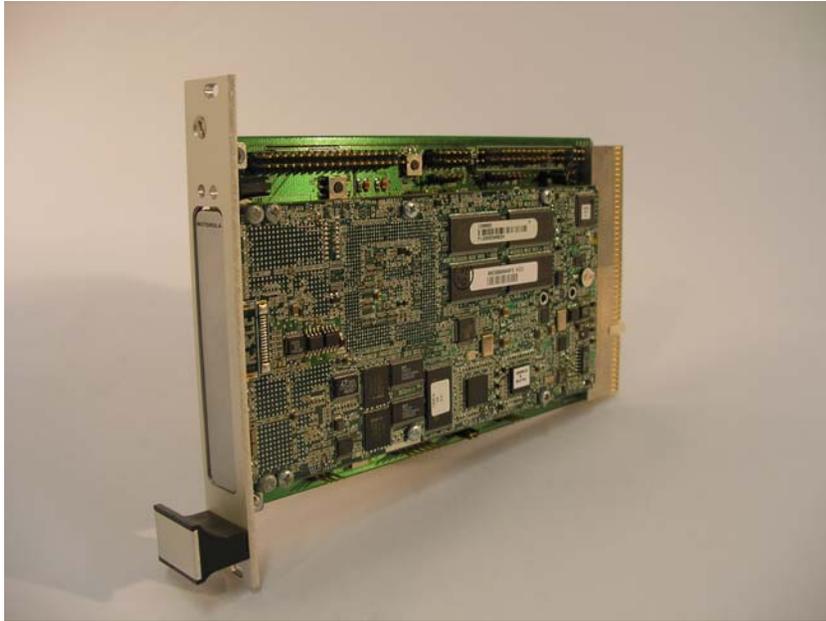
Dipl.-Ing. M. Seebode

Prof. Dr.-Ing. W. Gerth

Institut für Regelungstechnik – Leibniz Universität Hannover

<http://www.irt.uni-hannover.de>





- Parallelkinematisches Hüftgelenk

———— "Agile Eye" nach Prof. Gosselin ————

