

Slide 1

Real-Time Linux Anwendungen

Mario Taschwer
mario@fsmllabs.at
FSMLabs Data GmbH

April 2002

Slide 2

Motivation

- Es sollen einige Anwendungsfälle von Real-Time Linux vorgestellt werden, die Leistungsfähigkeit und typische Einsatzgebiete dieser Echtzeitbetriebssysteme demonstrieren sollen.
- Der Großteil der angeführten Beispiele wurde am Real-Time Linux Workshop 2001 in Milano präsentiert. Siehe <http://www.realtimelinuxfoundation.org/events/rtlws-2001/ws.html> (auch Vorgänger-Workshops erreichbar)
- Das Spektrum der RT Linux Anwendungen ist durch die vorgestellten Beispiele *nicht* erschöpft. Weitere interessante Anwendungen wie Herzschrittmacher und 60 MW Leistungsversorgung für Kernfusionsexperimente findet man unter <http://www.fsmllabs.at/links.html>

Slide 3

Gehörschutz mit digitaler aktiver Dämpfung

- *Passiver Gehörschutz* (absorbierendes Material) wirkt nur schlecht im niederfrequenten Bereich (< 500 Hz).
- *Aktive Dämpfung*: durch Mikrophon und Lautsprecher im Gehörschutz wird das Störsignal mittels eines phasen-invertierenden Verstärkers gedämpft.
- Analoge Verstärker mit konstanter Verstärkung können zu Instabilitäten führen, weil der räumliche Abstand zwischen Mikrophon und Lautsprecher eine frequenzabhängige Phasenverschiebung bewirkt.
- Digitale Regler können durch geeignete adaptive Filter eine wesentlich bessere Dämpfung im niederfrequenten Bereich erzielen.

Slide 4

Digitale aktive Dämpfung mit RT Linux

- Experimenteller Aufbau mit Athlon 700 MHz, A/D-D/A-Wandler, Reglersoftware unter RTAI (RT Linux Variante unter GPL).
- Sampling Frequenz beträgt 5 kHz, RT Reglertask wird interruptgesteuert.
- Resultat: mono-frequente Störungen konnten um 60 dB gedämpft werden.

Slide 5

Datenerfassungssystem für Spektroskopie

- **Mößbauereffekt:** im Kristallgitter gebundene Atomkerne können γ -Quanten mit scharfem Frequenzspektrum emittieren.
- **Messung:** durch Zählen von nicht absorbierten γ -Quanten; Erfassung von Zählraten in Abständen von 20 μ s.
- **Embedded Datenerfassungssystem:** 486DX/4MHz CPU, 16MB EDO-RAM, IDE Flash-Disk, Netzwerkkarte, ISA Datenerfassungskarte (für 2 unabhängige Zähler).
- **Software:** basiert auf *miniRTL*, Boot von Flash-Disk, passwortgeschützte Web-Schnittstelle für Befehle und Monitoring.
- **Benutzerapplikation (remote):** Java, Kontrollbefehle, Statusinformation, graphische Darstellung der Messdaten.
- **Kosten:** nur ca. 150 USD.

Slide 6

CNC-Steuerung mit Real-Time Linux

- Die Steuerung für eine CNC Dreh- und Fräsmaschine wurde von Windows-basierter Software auf RTLinux-Software umgestellt. Gründe für diese Migration:
- Herstellerfirma des proprietären Windows-Echtzeitsystems existiert nicht mehr.
- Echtzeitkomponenten (Achsensteuerung, Programmable Logic Controller) können auf einem embedded PC-Board unter RTLinux laufen, die GUI auf einem Windows-PC; Kommunikation über TCP/IP.
- RTLinux kann die benötigten harten Echtzeitanforderungen ($< 50\mu$ s scheduling jitter) auf dem embedded Board garantieren.
- Ein Standard Linux PC kann als Entwicklungsumgebung benutzt werden. Umstellung auf RTLinux war kostengünstig.

Slide 7

Menschenähnlicher Roboter H7 I

- Besteht aus Kopf, Rumpf, Armen und Beinen; 147 cm groß, 58 kg, insgesamt 30 Gelenke.
- **Sensoren:**
hochauflösende Stereo-Kameras, 6-Achsen-Kraftsensoren in den Füßen, Kreisel sensor in der Körpermitte, 9 Drucksensoren pro Hand.
- **Autonomer Gang auf zwei Beinen:**
dynamisches Ausbalancieren, Kollisionsvermeidung mit eigenen Roboterteilen, Bewegungsplanung (kurz- und mittelfristig), Erkennen von ebenen Segmenten (Treppe), High-Level Steuerung mit Joystick.

Slide 8

Menschenähnlicher Roboter H7 II

- **Vorstufen für Objektmanipulation:**
kollisionsfreie zielgerichtete Bewegungsplanung von Armen und des ganzen Körpers, 3D-visuelle Erfassung der Umgebung, dynamisch-stabile Bewegung des Körpers, Operator spezifiziert Zielobjekt via GUI.
- **Mensch-Roboter-Schnittstelle:**
 - Spracherkennung von Befehlsschlüsselwörtern, Unempfindlichkeit gegenüber Roboterlärm.
 - Erkennen und visuelles Verfolgen von menschlichen Umrissen, Gesichtserkennung.

Slide 9

H7 Hard- und Software

- **Im Roboter:** Dual Pentium-III 1.1 GHz, 256 MB SDRAM, Funknetzkarte, IEEE1394 Bilderfassungskarte, Soundkarte, 2 ISA I/O-Karten mit 12-Bit A/D-D/A-Wandler.
- **RTLinux Kernelmodule:** Motorsteuerung, Gleichgewichtskontrolle, Bewegungsberechnung für 3-5 Schritte.
- **Linux-Prozesse im Roboter:** 3D-Erkennung, Sprachaufnahme, Hindernis- und Kollisionsvermeidung, Trittplaner.
- **Verteilter PC-Cluster (remote):** high-level 3D-Erkennung (Gesichtserkennung, Ebenenerkennung), Bewegungsplanung und -berechnung, Spracherkennung.
- **Benutzerschnittstelle:** Joystick, high-level Befehle, remote Operating, Mensch-Roboter-Interaktion.

Slide 10

Fußball spielende Roboter

- Im **RoboCup** Projekt (www.robocup.org) geht es darum, Roboter zu entwickeln, die ohne menschliche Hilfe ein Fussballmatch austragen können.
- D.h. die Roboter bewegen sich selbständig am Fussballfeld, müssen den Ball und Gegner/Mitspieler lokalisieren und Entscheidungen treffen, welche Aktion als nächstes gesetzt wird.
- **Künstliche Intelligenz:** autonome Agenten, Verhaltenssteuerung, Integration von längerfristigem Planen und kurzfristigem Reagieren, Diagnose der Strategie anderer Spieler.
- **Regel- und Steuertechnik:** Objekterkennung (Ball, Mitspieler, Gegner, Tor), Bestimmung des eigenen Standortes, Bewegungssteuerung (Ballverfolgung, Markierungslinien; Gehbewegung), beträchtliche Echtzeitanforderungen.

Slide 11

RoboCup – Ligen

- **Middle Size League:** 4 Spieler pro Mannschaft, 5×9 m Spielfeld, Roboter sammeln Informationen nur über ihre Sensoren.
- **Small Size League:** 4 Spieler pro Mannschaft, Spielfeld so groß wie Tischtennisplatte, Roboter können auf Bilder einer Videokamera über dem Spielfeld zugreifen (globale Sicht).
- **Simulation League:** virtuelles Spielfeld von "Soccer Server" simuliert, 11 Spieler pro Mannschaft, Schwerpunkt auf Strategie und Kooperation.
- **Sony Legged Robot League:** vierbeinige Roboter, 3 Spieler pro Mannschaft.

Slide 12

RoboCup – Einsatz von Linux

- Die folgenden Roboterteams werden von Linux-Systemen (RT Linux?) gesteuert.
- *Minnow-Roboter* (Carnegie Mellon University): Laptop auf fahrbarem Untersatz: P3/700MHz, 256 MB RAM, 11 MBit/s wireless ethernet, USB Webcam, Debian GNU/Linux System.
<http://www-2.cs.cmu.edu/coral/minnow/>
- *Ulm Sparrows* (Uni Ulm): Siehtnix, Trifftnix, Passtnix und Haeltnix. P3/700MHz, 2 Motoren mit selbstentworfenem Controllerboard, Linux.
<http://smart.informatik.uni-ulm.de/SPARROWS/>
- *AGILO RoboCup Team* (TU München): MPEG Videos verfügbar.
http://wwwradig.in.tum.de/research/mobile_robots/robocup/