



# PEARL-News

Ausgabe 1/2009

Mitteilungen  
der GI-Fachgruppe Echtzeitsysteme

ISSN 1437-5966

# Impressum

Herausgeber	GI-Fachgruppe Echtzeitsysteme (RT) URL: <a href="http://www.real-time.de">http://www.real-time.de</a>
Sprecher	Dr. P. Holleczeck Universität Erlangen-Nürnberg, Regionales Rechenzentrum Martensstraße 1, D-91058 Erlangen Telefon: 09131/85-27817 Telefax: 09131/30 29 41 E-Mail: <a href="mailto:peter.holleczeck@rrze.uni-erlangen.de">peter.holleczeck@rrze.uni-erlangen.de</a>
Stellvertreter	Prof. Dr. Dr. W. Halang FernUniversität in Hagen Universitätsstraße 27 - PRG D-58084 Hagen Telefon: 02331/987-372 Telefax: 02331/987-375 E-Mail: <a href="mailto:wolfgang.halang@fernuni-hagen.de">wolfgang.halang@fernuni-hagen.de</a>
Redaktion	Prof. Dr. R. Müller FH Furtwangen, Fachbereich Computer- & Electrical Engineering Robert-Gerwig-Platz 1, 78120 Furtwangen Telefon: 07723/920-2416 Telefax: 07723/920-2610 E-Mail: <a href="mailto:mueller@hs-furtwangen.de">mueller@hs-furtwangen.de</a>
ISSN	1437-5966

Redaktionell abgeschlossen am 6. Juli 2009

## Einreichung von Beiträgen

Diese Zeitschrift soll nicht nur Mitteilungsblatt sein, sondern auch eine Plattform für den Informations- und Meinungsaustausch zwischen allen an den Fragen der Echtzeitprogrammierung Interessierten bilden. Diskussionsstoff bzw. offene Fragen gibt es auf unserem Gebiet reichlich.

Wir möchten Sie, liebe Leserinnen und Leser, daher ausdrücklich ermuntern, auch in Zukunft die PEARL-News durch Ihre Beiträge mit zu gestalten. Für ein ausgewogenes Bild der News sollten Beiträge nicht länger als 5 Seiten sein.

Rainer Müller (Furtwangen)

## Inhaltsverzeichnis

<b>1 Programm des Workshops PEARL 2009</b>	<b>3</b>
<b>2 Zeitmessungen unter Xenomai-Linux</b>	<b>4</b>
<b>3 PEARL-User-Group / AK1 GI-FG REAL-TIME - Treffen 2009</b>	<b>8</b>

# 1 Programm des Workshops PEARL 2009

Der diesjährige Workshop steht unter dem Leitthema „Echtzeit 2009 - Software-intensive verteilte Echtzeitsysteme“. Er wird wieder im Hotel Ebertor in Boppard am Rhein stattfinden. Das Programmkomitee hat auf seiner Sitzung am 13. Mai 2008 in Frankfurt/Main das nachstehende Programm zusammengestellt. Der Tagungsband erscheint - wie in den vergangenen Jahren - in der Reihe „Informatik aktuell“ des Springer-Verlages.

## Erster Workshop-Tag: Donnerstag, der 19. November 2008

11:00 Treffen der Arbeitskreise

13:00 Begrüßung

13:15 Sitzung 1: *Kommunikation in verteilten Echtzeitsystemen* (Sitzungsleitung: Prof. Heitmann)

*Ein Framework für echtzeitfähige Ethernet-Netzwerke*  
(F. Dopatka, Fa. GFU Cyrus AG)

*UniLoG - ein System zur verteilten Lastgenerierung in Netzen*  
(A.W. Kolesnikov, Universität Hamburg)

*Planung der Übertragung von Echtzeitnachrichten in Netzwerken mit Bandbreitenreservierung am Beispiel von Profinet IRT*  
(O. Graeser, Hochschule Ostwestfalen-Lippe)

14:45 Pause

15:15 Sitzung 2: *Engineering-Werkzeuge für Echtzeitsystemen* (Sitzungsleitung: Prof. Wedde)

*Raum-Zeit-Constraints als Programmierabstraktion für Anwendungen in mobilen verteilten Systemen*  
(M. Däumler, TU Chemnitz)

*Formale Bestimmung von Systemparametern zum transparenten Scheduling virtueller Maschinen unter Echtzeitbedingungen*  
(T. Kerstan, Universität Paderborn)

16:15 Pause

16:45 Sitzung 3: *Methoden des Ressourcen-Managements* (Sitzungsleitung: Prof. Zöbel)

*Anwendung von Methoden der Logistik und Netzplantechnik zur präzedenz- und ressourcenbeschränkten Ablaufplanung von Echtzeitsystemen*  
(R. Gumzej, Universität Maribor)

*Bienen-inspirierte Straßenverkehrsroutingsysteme*  
(S. Senge, TU Dortmund)

18:00 Abendessen

19:15 Abendprogramm

Studentische Abschlußarbeiten  
Mitgliederversammlung der Fachgruppe

## Zweiter Workshop-Tag: Freitag, der 20. November 2008

9:00 Sitzung 4: *Fertigungstechnische Anwendungen* (Sitzungsleitung: Prof. Frey)

*Verkehrsmuster und Kommunikationsanforderungen von verteilten Echtzeit Systemen in der Fertigungsautomatisierung*  
(H. Trsek, Hochschule Ostwestfalen-Lippe)

*CAEX-basierte Produkt- und Prozessdaten für die Leittechnik im OPC-UA Engineering-Framework*  
(M. Schlepen, Fraunhofer-Institut für Informations- und Datenverarbeitung Karlsruhe)

10:30 Pause

11:00 Sitzung 5: *Performance-Analyse und Debugging* (Sitzungsleitung: Prof. Schiedermeier)

*Universeller hybrider Systemebeobachter für Echtzeitsysteme*  
(B. Pietsch, Universität Hannover)

*Using CPU Stubs to Achieve Realtime Behaviour*  
(P. Trapp, Fachhochschule Ingolstadt)

*Protokollanalyse und Informationsverfolgung zur Fehlerdiagnose in verteilten Echtzeitsystemen*  
(E. Noak, Hochschule Albstadt-Sigmaringen)

12:30 Verabschiedung

12:45 Imbiss

## 2 Zeitmessungen unter Xenomai-Linux

### 2.1 Motivation

Nach den meinen letzten Messungen der zeitlichen Zuverlässigkeit in Linuxsystemen [1] sollte die Untersuchung auf eine Realzeiterweiterung dieses Betriebssystems ausgedehnt werden. Hierzu wurde die gleiche Hardware, wie in der letzten Messung verwendet, damit die Ergebnisse gut vergleichbar sind. Für Details zur Messhardware sei auf den Vorläuferartikel verwiesen.

Für die hier vorgestellten Messungen kam folgende Konfiguration zum Einsatz: Linux-Kernel 2.6.25.11 und Xenomai-Patch 2.4.4 (vgl. Tabelle 3)

### 2.2 Xenomai

Die Realzeiterweiterung Xenomai [2] arbeitet mit einem eigenem Scheduler, der die Einhaltung von harten Realzeitanforderungen erlaubt. Eine übersichtliche kurze Darstellung der Funktionsweise findet sich in [4]. Die Struktur ist in Abbildung 1 kurz skizziert.

Die Interrupts des Rechnersystems werden dazu über die sogenannte Interrupt-Pipe (I-Pipe) der Adeos (Adaptive Domain Environment for Operating Systems) [5] Schicht an registrierten Konsumenten weitergeleitet. (vgl. Abbildung 2) Wenn ein Interrupt in einer Schicht verarbeitet wurde, so *kann* der Treiber entscheiden, ob dieser Interrupt noch an folgende Schicht weitergeleitet wird. Dadurch, dass Xenomai zuerst bedient wird bekommen dessen Prozesse die höchste Priorität im System.

Xenomai stellt Realzeittasks im Userspace und im Kernelbereich bereit. In diesem Artikel werden nur Usermodetasks im *native skin* benutzt.

### 2.3 Testprogramm

Die zeitliche Verzögerung wird hier im sogenannten *native skin* verfügbare Funktionen `rt_task_set_periodic` und `rt_task_wait_period` benutzt. Die Auswertung der einzeln erzielten Verzögerungen erfolgte gleich im Testprogramm.

Zur Erzeugung von Hintergrundlast wurden zwei Szenarien verfolgt:

1. Prozesse mit Endlosschleifen im normalen Linuxscheduler
2. Prozesse mit Endlosschleifen von Xenomai mit schlechterer Priorität

Zur Messung wurde einerseits die interne Uhr (`rt_timer_read`) herangezogen, wobei hier die Zeiten zwischen den Ticks gemessen wurden. Bei Verwendung der externen Uhr konnte nur die Zeit zwischen 2 Ticks

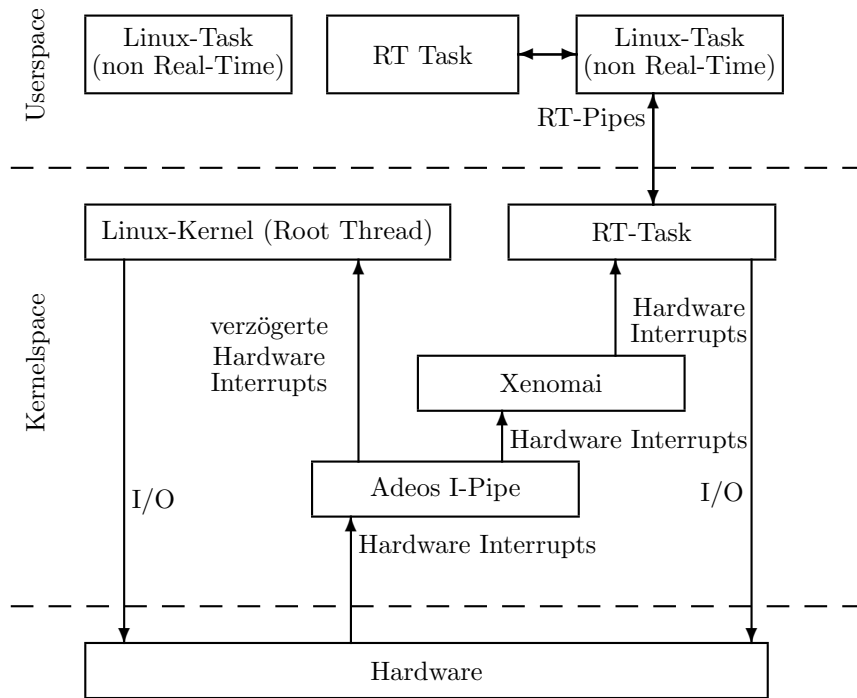


Abbildung 1: Interrupt und Kommunikationsfluß in Xenomai (nach: [3]). API-Aufrufsstellen wurden aus Gründen der Übersichtlichkeit weggelassen

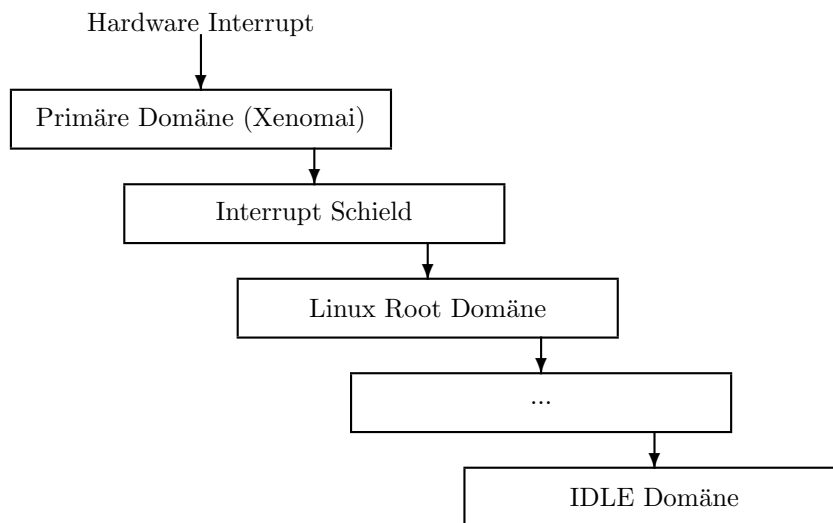


Abbildung 2: Struktur der Adeos I-Pipe in Xenomai (nach: [4, 5]). Registrierte Interrupts werden von den einzelnen Domänen behandelt und ggf. auch noch an nachfolgende Domänen weitergereicht.

gemessen werden. Die Messtask läuft hierbei im *Userspace*. Die Messungen wurden an einem älteren PC durchgeführt (vgl. Tabelle 3).

## 2.4 Messergebnisse

Erste Vormessungen zeigten ein sehr stabiles Verhalten (vgl. Abbildung 3), sodass bei den einzelnen Messungen keine Histogramme erstellt wurden, sondern die charakterisierenden Werte Minimum, Maximum und arithmetischer Mittelwert simultan mitbestimmt wurden.

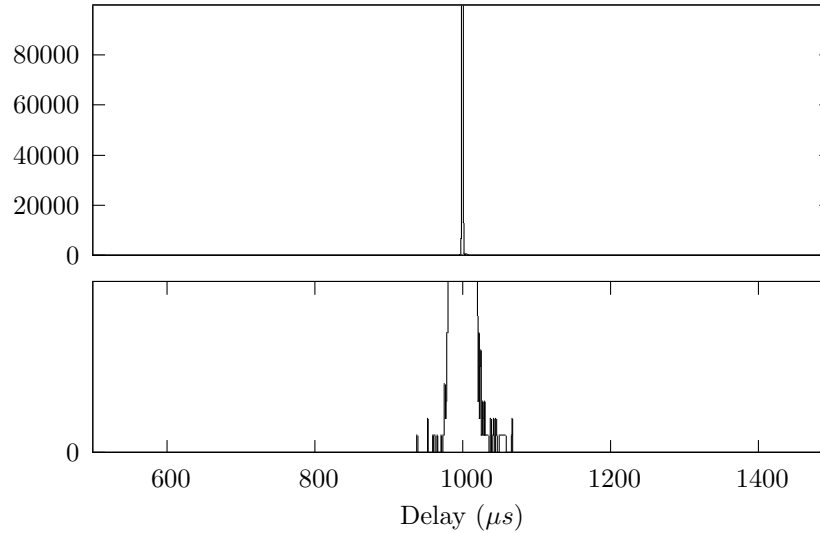


Abbildung 3: Verteilung der erzielten Verzögerungszeiten zwischen 2 Ticks. In der unteren Darstellung wurde die y-Skala gespreizt, sodass die einzelnen Ausreißer sichtbar werden.

Soll - verzögerung	interner Zeitmessung			externe Zeitmessung		
	Min	Max	Mittel	Min	Max	Mittel
10	1	21	9.52	1	18	6.9
50	39	62	49.4	40	64	49.5
100	84	111	99.7	90	110	99.7
200	157	246	199.3	167	247	199.7
500	440	562	499.5	439	557	499.5
1000	934	1073	999.3	951	1074	999.3
2000	1913	2103	1999.5	1920	2013	1999.0
5000	4941	5063	4999.6	4935	5088	4999.8
10000	9935	10078	9999.5	9933	10029	9999.6
100000	99907	100111	99999.5	99975	100036	100002.0

Tabelle 1: Vergleich der mit interner und externer Uhr gemessenen Verzögerungen. Es wurden jeweils  $10^6$  Messungen in einer Schleife durchgeführt. Es ist der minimale, maximale beobachtete Wert der Verzögerung, sowie der arithmetische Mittelwert angegeben. Alle Zeitangaben verstehen sich in  $\mu s$ .

Die Ergebnisse von Tab 1 zeigen eine weitgehende Übereinstimmung zwischen der Messung mit der xenomai-internen Uhr und der externen Uhr. In beiden Fällen wurden Abweichung von der Sollzeit von  $\approx \pm 90 \mu s$  beobachtet.

Auffallend ist dennoch, dass der Mittelwert der Verzögerungszeit die Sollzeit meist leicht unterschreitet. Für eine verlässliche Aussage müßten hierzu die Meßgeräte in ihrer Kalibrierung überprüft werden.

Die Messungen mit Hintergrundlast im normalen Linuxprozessen zeigten - wie zu erwarten war - keinerlei signifikanten Veränderungen.

Die Messungen mit Hintergrundlast durch niederpriorie Realzeittasks zeigte eine Stabilisierung der Werte (vgl. Tabelle 2). Dies kann evtl. darauf zurückgeführt werden, dass die Fortsetzung des Messprozesses hier stets aus der gleichen Umgebung heraus erfolgt, und kein Domänenwechsel zwischen Linuxdomäne und Xenomai die Laufzeit beeinflusst.

Soll - verzögerung	Abweichung			
	ohne Hintergrundprozess		mit Hintergrundprozess	
10	-9	+8	-7	+10
50	-10	+14	-2	+7
100	-10	+10	-2	+8
200	-23	+47	-6	+12
500	-61	+57	-28	+1
1000	-49	+74	-17	+0
2000	-80	+13	-18	+18
5000	-65	+88	-20	+0
10000	-67	+29	-12	+13
100000	-25	+36	-10	+4

Tabelle 2: Maximale Abweichungen von der Sollzeit unter Xenomai zwischen den Szenarien ohne Realzeithintergrundprozess und mit Realzeithintergrundprozess. Es wurden jeweils  $10^6$  Wiederholungen durchgeführt. Alle Zeitangaben verstehen sich in  $\mu s$

## 2.5 Ergebnis

Diese Messungen zeigen, dass eine Genauigkeit der zeitlichen Verzögerung bei diesem System unabhängig von der gewünschten Verzögerung mit  $\approx \pm 90 \mu s$  recht gut ist. Im Vergleich zu den Werten unter normalen Linuxscheduling und dem sogenannten Realtimescheduling (`SCHED_FIFO`), die zeitliche Ausreißer im Bereich von  $> 150 ms$  aufweisen, bringt der Einsatz von Xenomai eine deutliche Verbesserung des Realzeitverhaltens.

Allerdings war festzustellen, dass auch Verzögerungen auftraten, die kürzer als die Sollzeit waren. Dies läßt sich dadurch erklären, dass bei einer periodischen Erzeugung von Pulsen, eine verspätete Ablesung zu einem Punkt sowohl eine Verlängerung der vorherigen Intervalls führt, und gleichzeitig zu einer Verkürzung des nachfolgenden Intervalls führt. Dies müsste sich im Mittelwert aufheben. Dies ist allerdings hier nicht der Fall (vgl. Tabelle 1).

Die Architektur des Adeossschicht sollte sicherstellen, dass Interrupts, die nicht von der Realzeitdomäne benötigt werden, keinen Einfluß auf die Zeitstabilität haben. Allerdings ist die primäre Interrupterfassung sicherlich ein Störfaktor.

In [4] wird dargelegt, dass *kernel based tasks* geringere Latenzzeiten besitzen. Welche Werte sich dann erreichen lassen, ist Thema der weiteren Untersuchungen.

CPU	INTEL Celeron
Takt	1 GHz
Speicherausbau	1GB (128kB Cache)
Betriebssystem	Linux (Kernel 2.6.25.11) Xenomai-Patch 2.4.4
BIOS	Award

Tabelle 3: Eigenschaften des eingesetzten PCs. Es wurden keinerlei Optimierungen im Bereich Cache, usw. durchgeführt.

# Literatur

- [1] *R. Müller*: Zeitmessungen unter Linux; PEARL-News (**2**) 2008
- [2] *Xenomai*: Real-Time Framework for Linux; <http://www.xenomai.org>
- [3] *Xenomai*: Native API Tour; <http://www.xenomai.org>
- [4] *S. Smolorz* Echtzeit-Linux mit Xenomai; Elektronik (**3**) 2007; S86ff
- [5] *K. Yaghmour*: Adaptive Domain Environment for Operating Systems; <http://www.opersys.com/ftp/pub/Adeos/adeos.pdf>

Rainer Müller  
Hochschule Furtwangen  
mueller@hs-furtwangen.de

## 3 PEARL-User-Group / AK1 GI-FG REAL-TIME - Treffen 2009

Das Treffen der PEARL-User-Group bzw. des Arbeitskreises „Embedded Systems, RTOS-UH/PEARL“ der GI-Fachgruppe „Echtzeitsysteme und PEARL (REAL-TIME)“ fand am Donnerstag, 04.06.2009, 14:00-17:00 Uhr, mit 13 Teilnehmern am Institut für Regelungstechnik der Leibniz Universität Hannover statt.

### TOP 1: Neues aus der Fachgruppe „Echtzeitsysteme“

- Die Fachgruppe wird innerhalb der GI jetzt auch als Fachausschuss geführt.
- Nach dem Ausscheiden von Prof. Thiele wird Prof. Gumzej die Leitung des Arbeitskreises „PEARL in der Ausbildung – PEARL-Sprachpflege“ übernehmen.
- Der neue Arbeitskreis „Echtzeitfähige Kommunikationssysteme“ wurde eingerichtet.
- Im Wettbewerb für Nachwuchsarbeiten sollen in 2009 drei Diplomarbeiten ausgezeichnet werden.
- Die Fachgruppe ist finanziell recht gut ausgestattet.
- In Kürze wird das Lehrbuch „Echtzeit“ von Prof. Benra und Prof. Halang erscheinen.
- Ausblick auf den Workshop „Echtzeit 2009“, für den relativ viele Beiträge eingegangen sind.

Es folgt eine Diskussion über die Nachfolge der Leitung des Arbeitskreises „Embedded Systems, RTOS-UH/PEARL“ nach dem Ausscheiden von Prof. Gerth im September 2009. Die Anwesenden sind sich einig, dass die Leitung im Hochschulbereich bleiben sollte, um größtmögliche Neutralität zu wahren. Ein möglicher Kandidat soll in Kürze von Prof. Gerth kontaktiert werden.

### TOP 2: Weiterentwicklungen PEARL90 und Systemkomponenten

- Im letzten Jahr sind weitere 20.000 RTOS-UH Systeme verbaut worden (ohne esd GmbH).
- Korrekturen und Erweiterungen des Systems seit dem letzten Treffen (s. auch Revisionsliste):
  - Anpassung der Compilerausgabe für den PPC405 für Kompatibilität mit dem PE.
  - Debugger-Output bei Langnamen in der Large-Version.
  - IS/ISNT funktionierten nicht bei Zeigern auf Felder.
  - NIL und statische Variablen sind jetzt als INIT-Werte für Zeiger erlaubt.
  - Der Compiler ließ unter Umständen ein ACTIVATE-Statement mit einer CHAR-Variablen als Argument zu.



- Neue Version des RTOS-Systems für den WinSTon Emulator:
    - \* ESC-Taste auch bei max. Geschwindigkeit verwendbar.
    - \* Die Drives /H0, /H1 usw. stehen jetzt ebenfalls zur Verfügung, wobei /H0 der zweiten ATARI-Partition (also /X1) entspricht.
  - In den TAPP wurde eine Altivec-Unterstützung integriert.
  - Das neue Tool SLICES listet alle im System vorhandenen Scheiben mit ihren Inhalten auf.
  - Neues Ausgabelayout des HELP-Befehls.
  - Der UHF3 konnte Dateien mit einem Datum ab 2017 nicht finden.
  - Löschen des #ENV-Speichers beim Setup
  - Neuer Nuc 7.9-P ohne funktionelle Änderungen, aber auch für 68K-Systeme jetzt ohne Checker.
- Diskussion von Möglichkeiten, in PEARL-Programmen ein schnelles ACTIVATE mit der Task-ID ohne Namensuche zu realisieren.

### TOP3: Vorstellung des OSOBS (Operation System Observer)

Herr Pietsch vom Institut für Regelungstechnik stellt einen hybriden Beobachter für Echtzeitsysteme vor. Dieser nimmt mit einer eigens entwickelten Hardware Daten auf, die vom entsprechend modifizierten Betriebssystem an bestimmten Beobachtungspunkten (z.B. Taskumschaltung, Interrupts, Semaphorenzugriff) ausgesendet werden. Aus diesen Daten kann mit der Kenntnis der Systemstruktur der Ablauf rekonstruiert werden. Durch eine flexible Hardware mit einem FPGA können sowohl Mikrocontroller als auch leistungsfähige Systeme beobachtet werden.

### TOP 4: Neue RTOS-UH Implementierungen, Berichte aus den Ingenieurbüros, Entwicklungsabteilungen und Forschungsinstituten

- Der in einer Kooperation zwischen der FH Hannover und der Fa. IEP entwickelte Einplatinencomputer RTTB (RealTime Training Board) auf Basis des MPC561-Prozessors (56 MHz) wird etwa 250 € kosten.
- Aktuelle Arbeiten in den Ingenieurbüros und Firmen umfassen:
  - MS-Filemanager mit FAT32 Unterstützung (IEP).
  - Nachfolge für das MVME5500 System (IEP).
  - Nachfolge des Bioprozessleitsystems UBICON (esd)
  - Betreuung und Erweiterung von diversen Anwendungen unterschiedlichster Größenordnung, wobei mit Ausnahme des 68332-Prozessors der PowerPC immer weitere Verbreitung findet.

#### Protokollführer

Dr.-Ing. T. Lilge  
 Institut für Regelungstechnik  
 Appelstr. 11  
 30167 Hannover  
 lilge@irt.uni-hannover.de

12. Juni 2009