



Technische Hochschule
Ingolstadt

*Zukunft in
Bewegung*

Task-Migration in verteilten eingebetteten Mehrkernsystemen

Tobias Meier 16.11.2017





Agenda

1. **Steigender Bedarf an Rechenkapazitäten der Steuergeräte**
2. **Situationsbasierte Verteilung von Rechenkapazität in verteilten Mehrkernsystem**
3. **Situationsbasiertes Scheduler für verteilte Mehrkernsysteme**
4. **Task-Migration in einem verteilten Mehrkernsystem**
5. **Evaluation**

Steigender Bedarf an Rechenkapazitäten der Steuergeräte



-  Fußgängererkennung geringe Servicequalität
-  Fußgängererkennung hohe Servicequalität



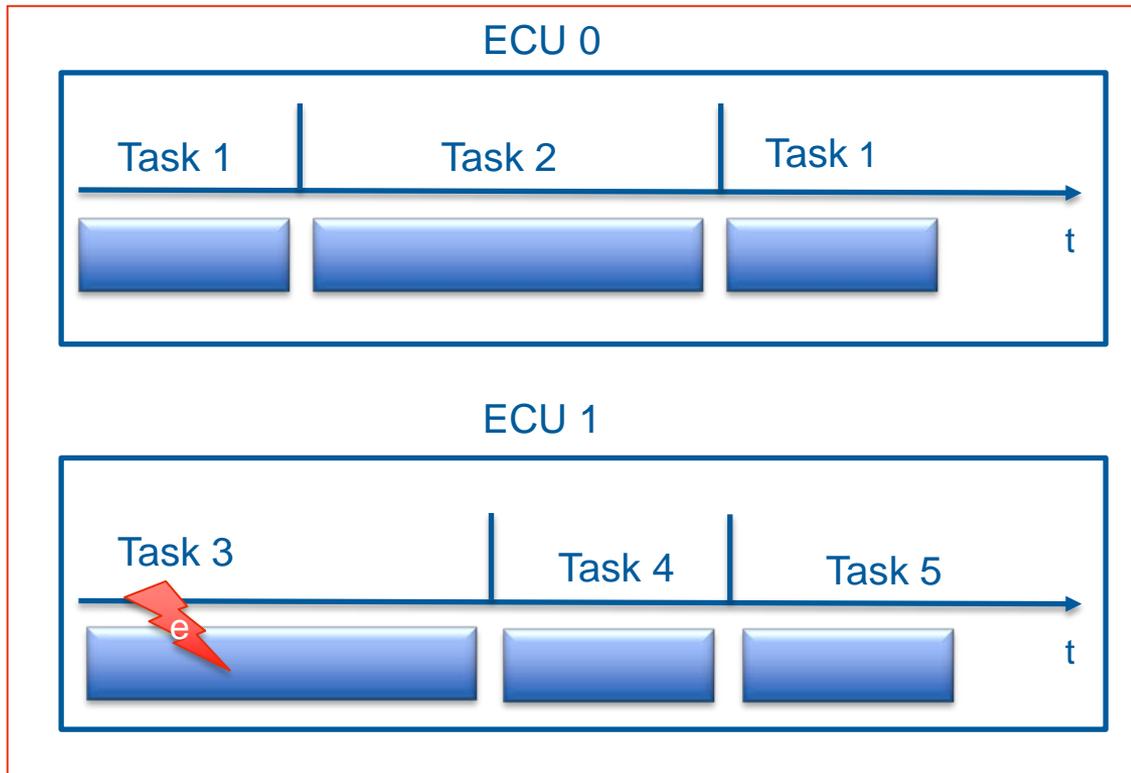
- Der Ressourcenverbrauch wird durch die momentane Situation definiert
- Die Situation verändert sich während der Laufzeit

Task-Migration in verteilten eingebetteten Mehrkernsystemen



Situationsbasierte Verteilung von Rechenkapazität in verteiltes Mehrkernsystem

Situation <20 km/h



Ereignisse betreffen alle Steuergeräte



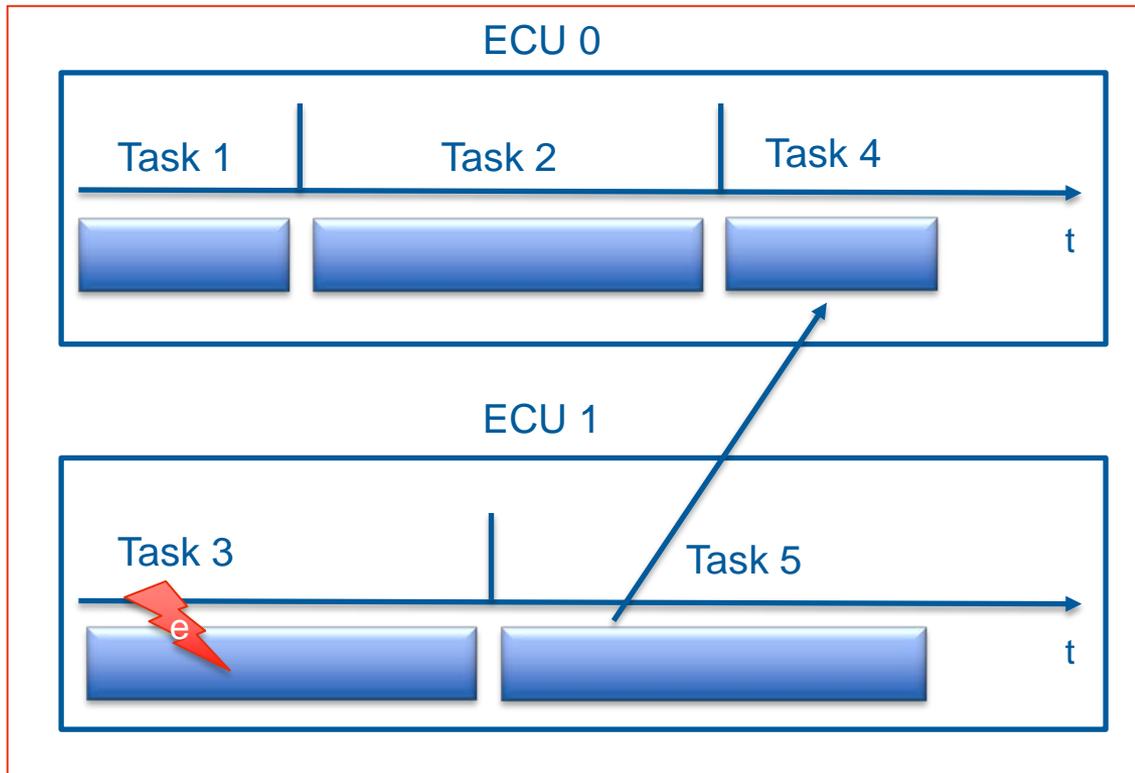
Reaktion mit einem sinnvollen
Verteilungsplan für alle Steuergeräte

Task-Migration in verteilten eingebetteten Mehrkernsystemen



Situationsbasierte Verteilung von Rechenkapazität in verteiltes Mehrkernsystem

Situation <20 km/h



Ereignisse betreffen alle Steuergeräte



Reaktion mit einem sinnvollen
Verteilungsplan für alle Steuergeräte



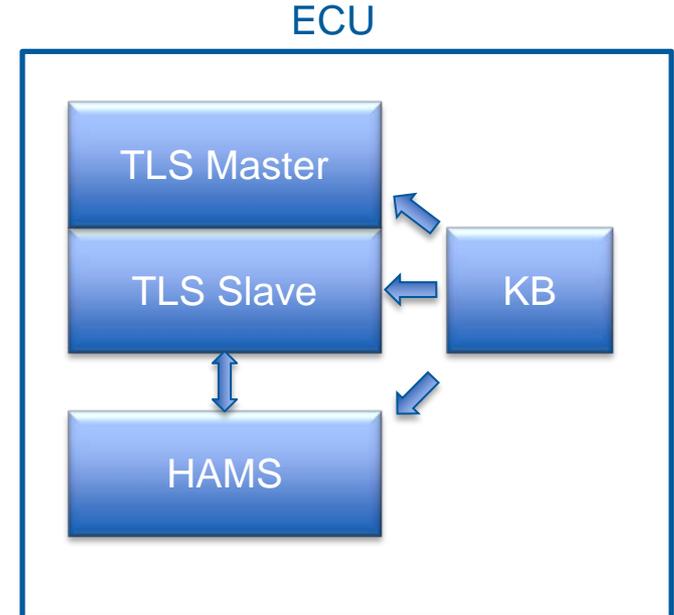
Durch die Rekonfiguration
entstehen freie Kapazitäten



Die Rekonfiguration erfordert eine
Verschiebung von Tasks zwischen
zwei Steuergeräten

Hierarchischer asynchroner Multi-Core Scheduler für verteilte Mehrkernsysteme (HAMS-MC)

- Weiterentwicklung von HAMS (situationsbasierter Multi-Core Scheduler)
- Erweiterung um den Third Level Scheduler (TLS)
- Vorberechnete Wissensdatenbank (KB)
- TLS Master: Determinismus im verteilten Mehrkernsystem
- TLS Slave: Rekonfiguration des Steuergeräts in Verbindung mit HAMS

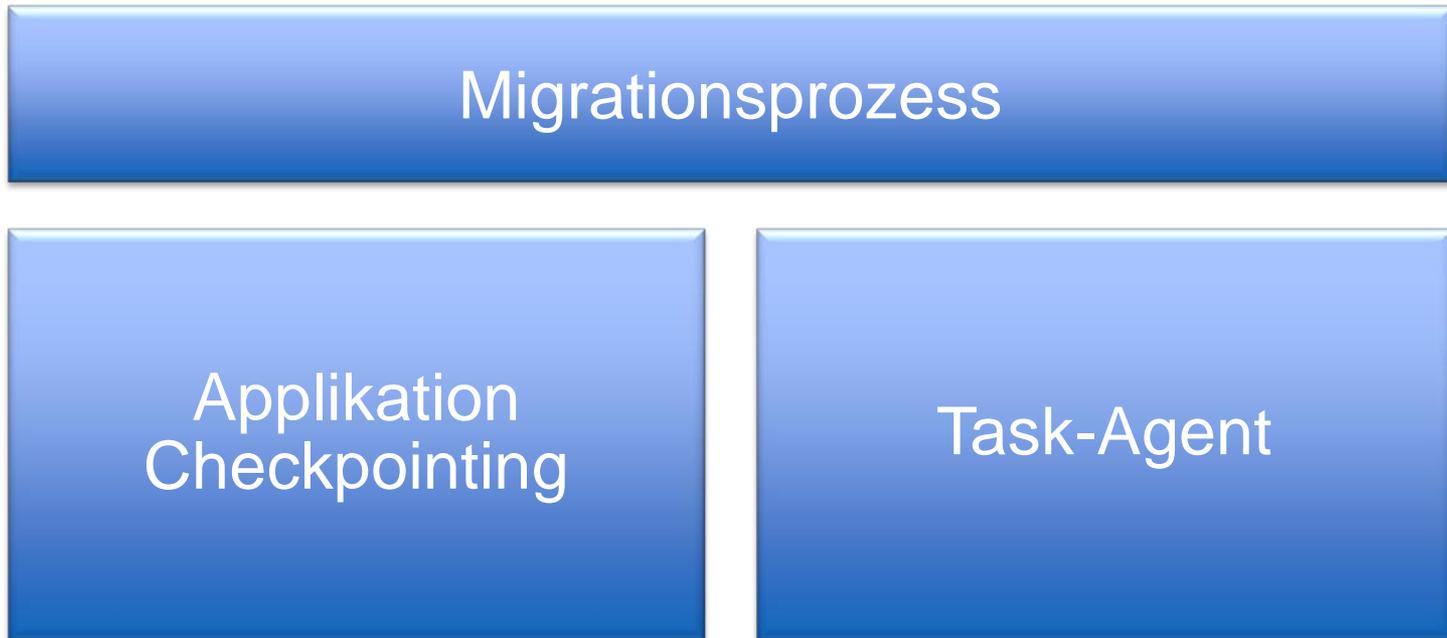


Task-Migration in verteilten eingebetteten Mehrkernsystemen



Situationsbasierte Verteilung von Rechenkapazität in verteiltes Mehrkernsystem





1. Applikation Checkpointing

Sicherung der wesentlich zur Wiederherstellung des Tasks benötigten Informationen und Zustände (Laufzeitinformationen)

- Geringer Umfang des Kontextes
- Keine Anpassung des Kontexts an Zielsystem notwendig
- Architektur unabhängig



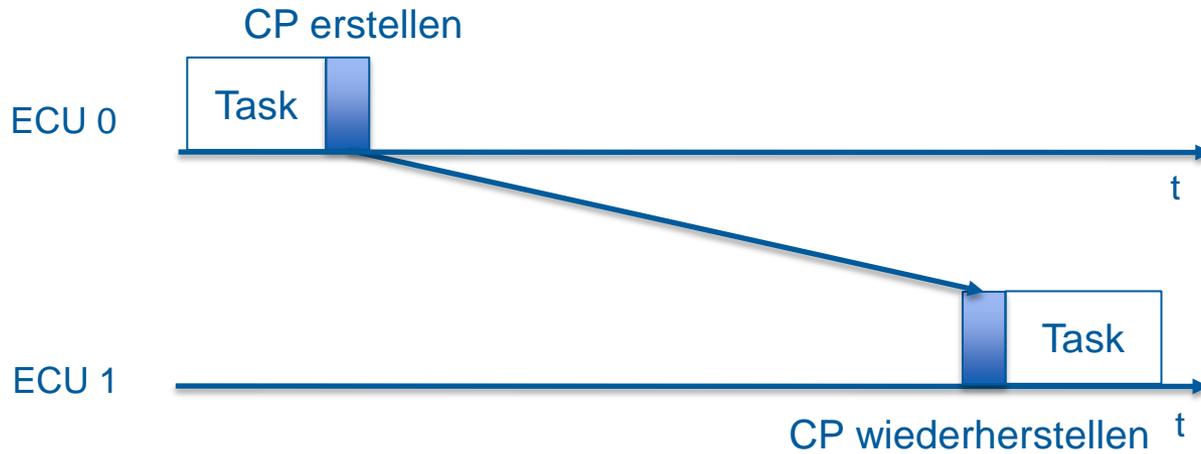
1. Applikation Checkpointing

Aufteilung der Laufzeit des Tasks und des Task-Agent in die Phasen
Vorbereitung und Ausführung





2. Task-Agent

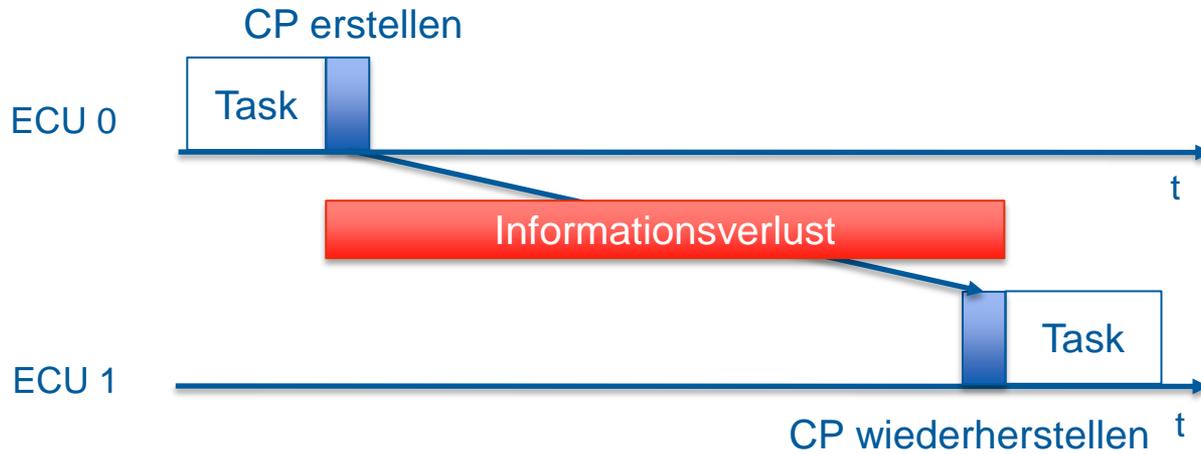


Erhöhter Zeitbedarf für Migration auf Mehrkernsystem

Task-Migration in verteilten eingebetteten Mehrkernsystemen



2. Task-Agent



Erhöhter Zeitbedarf für Migration auf Mehrkernsystem

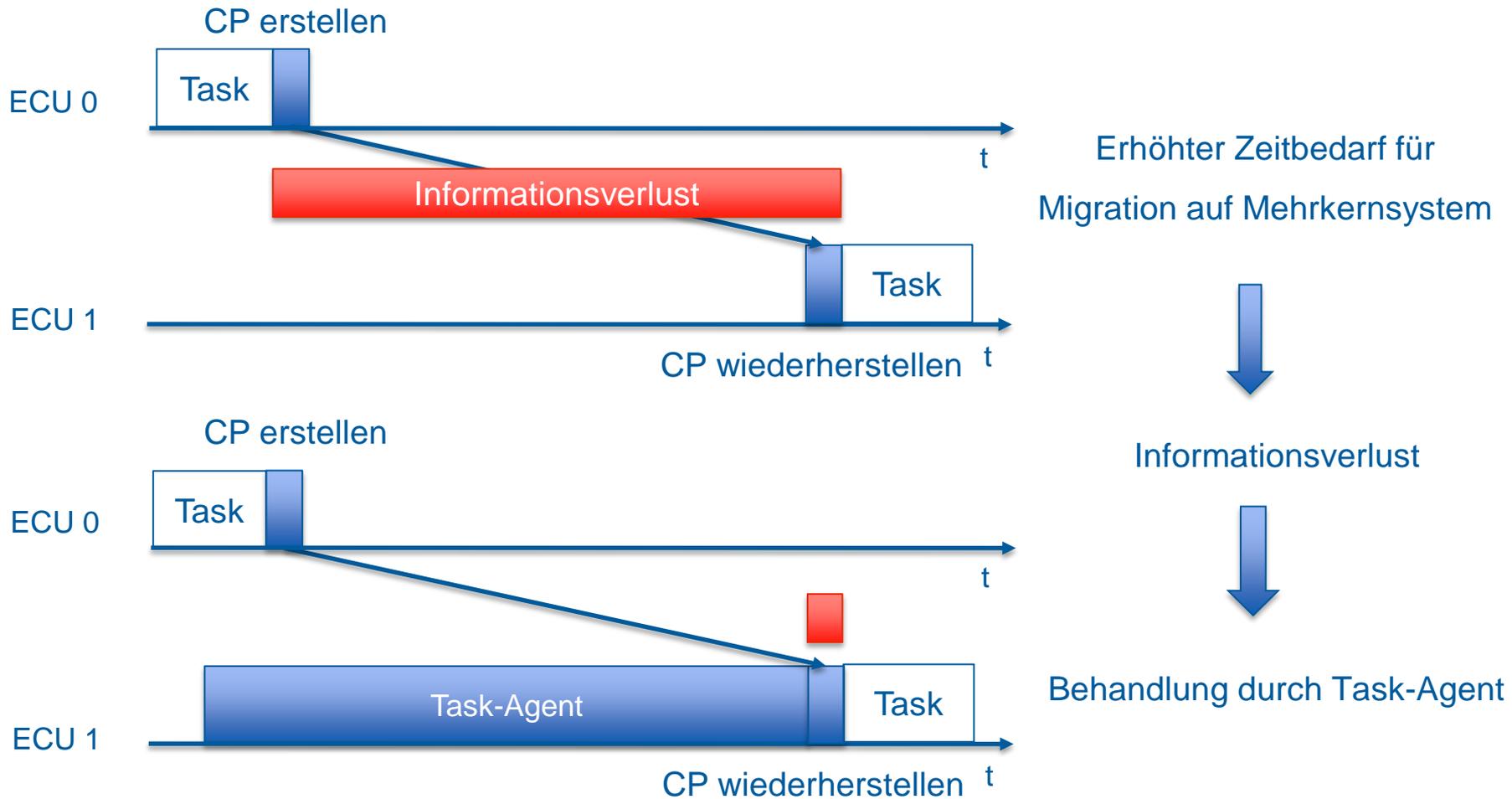


Informationsverlust

Task-Migration in verteilten eingebetteten Mehrkernsystemen

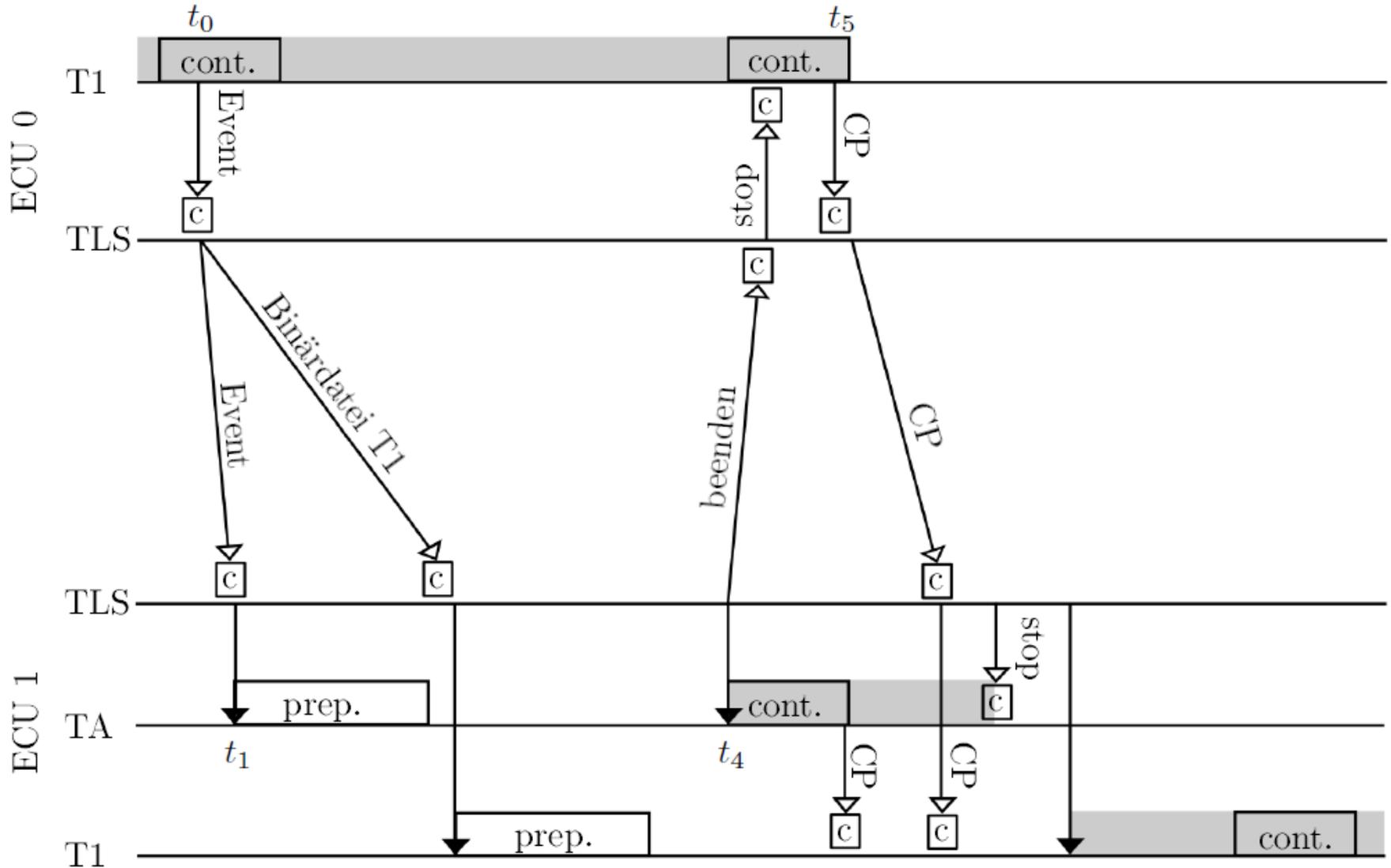


2. Task-Agent





Migrationsprozess





Evaluation

Ziele

Gesamtdauer der Migration

Informationsverlust wird durch Task-Agent vermieden

Testaufbau

2 * ARMv7 (Pandaboard)

Fast Ethernet

Linux mit RT-Patch

HAMS-MC (lite)

Szenario

Migration eines Tasks von ECU 0 auf ECU 1

Messmethode

GPIO Pins in Verbindung mit Oszilloskop

Scheduler Analyse mittels KernelShark

Evaluation

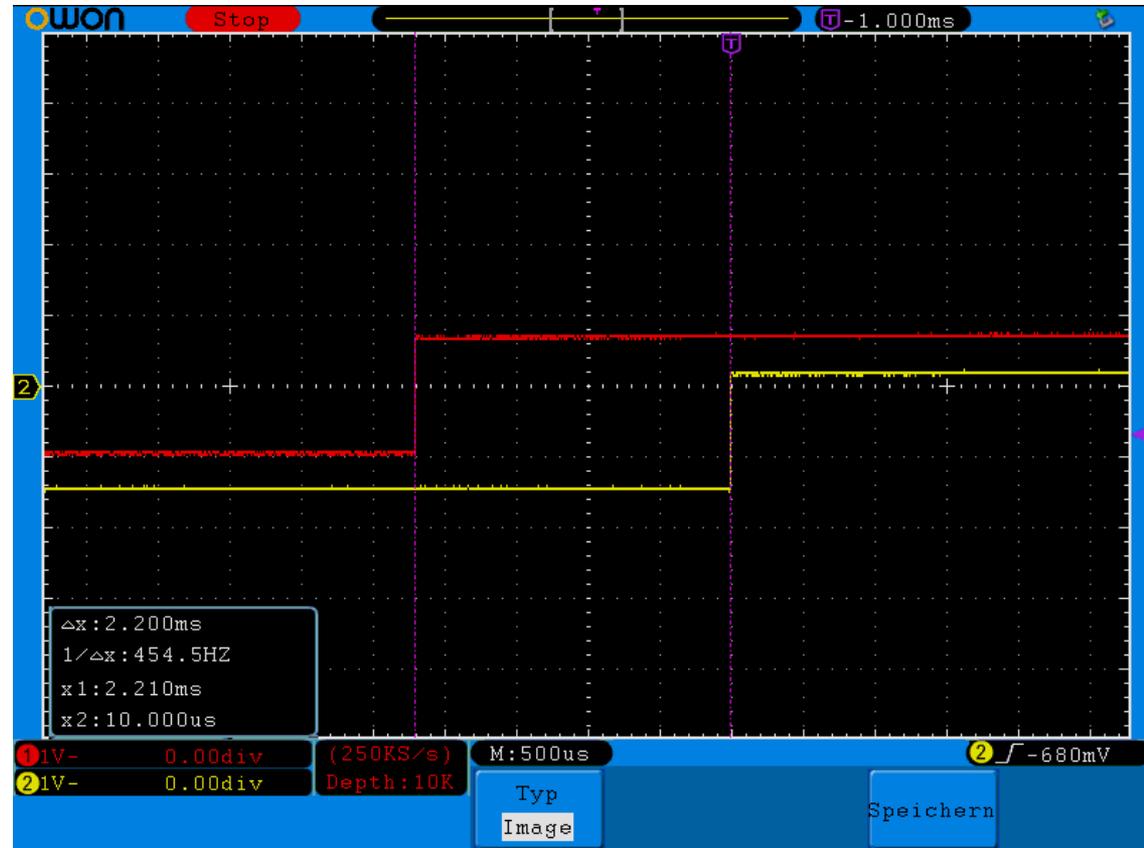
GPIO Pins als Trigger Punkte

- Kommunikationsdauer
- Synchronisation zweier Traces

geringer Einfluss auf die Messung

 PIN 138 auf ECU 0

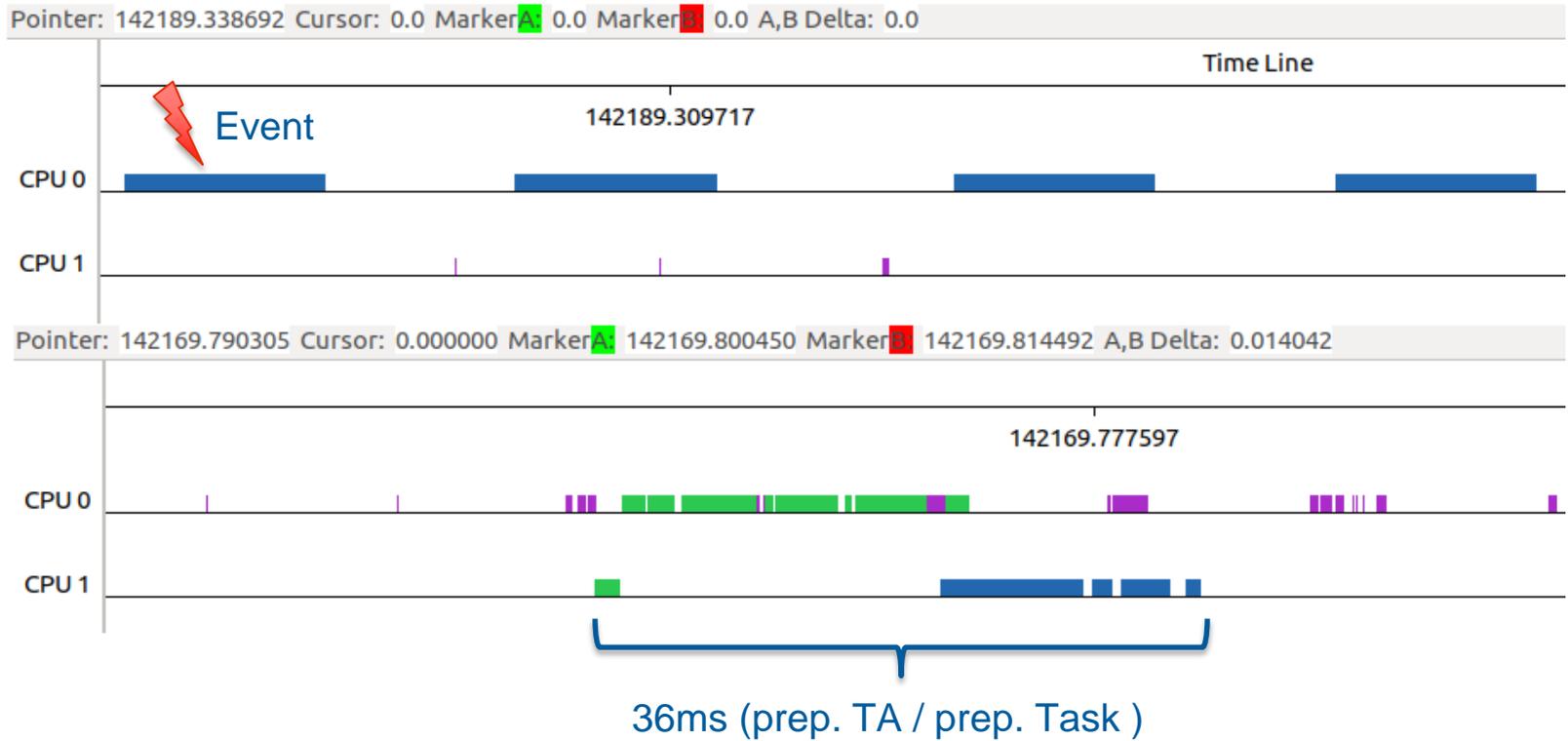
 PIN 138 auf ECU 1





Evaluation

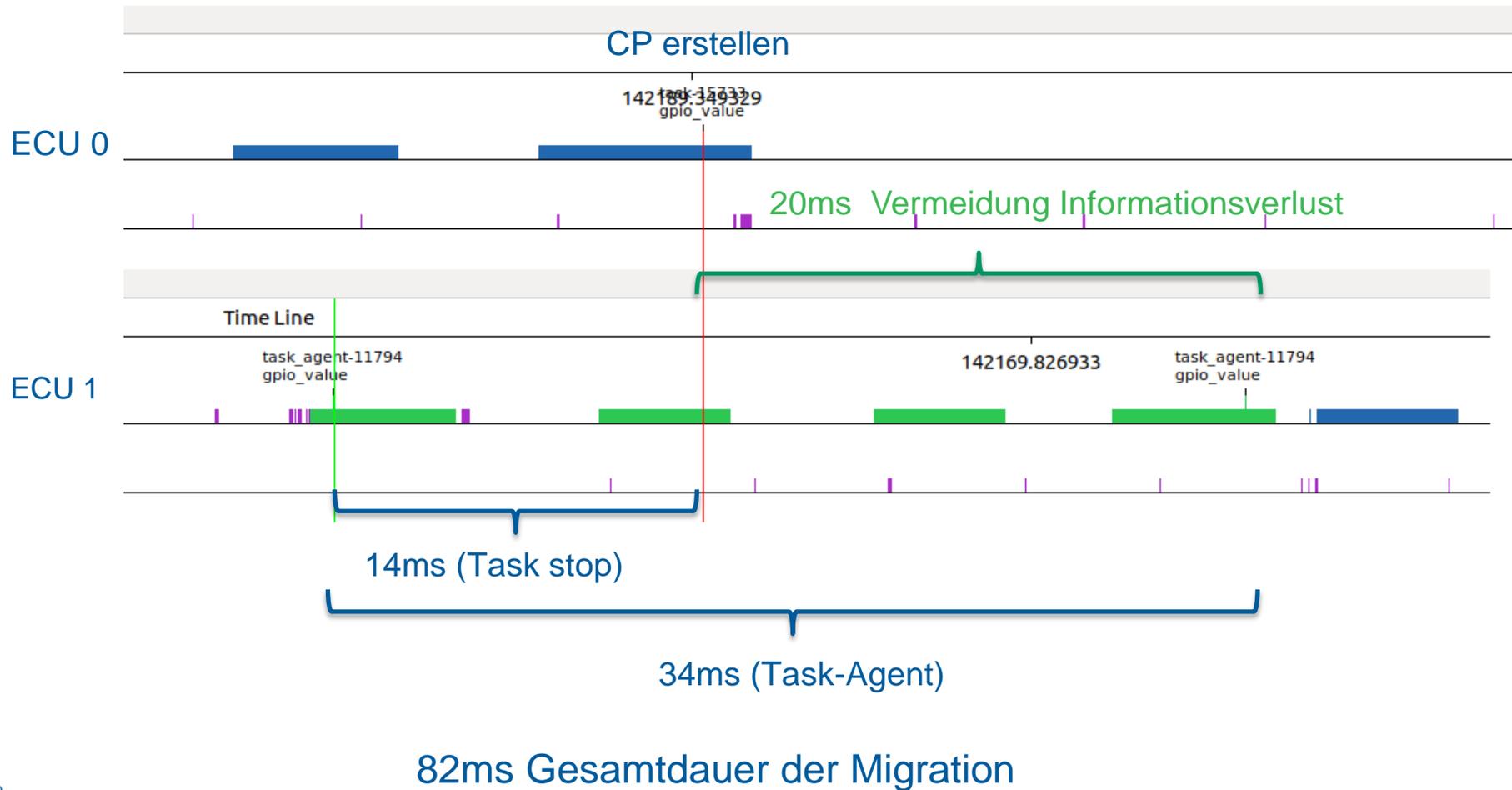
KernelShark trace 1





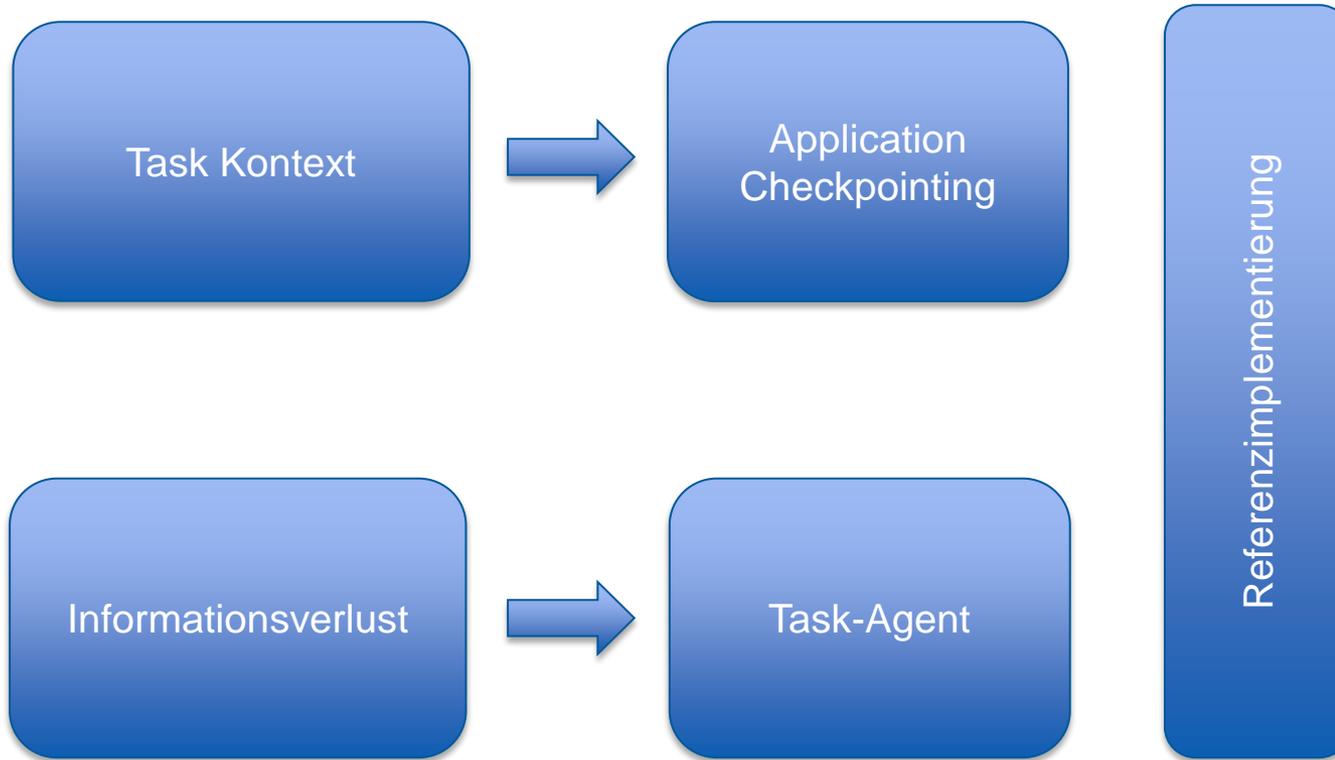
Evaluation

KernelShark trace 2

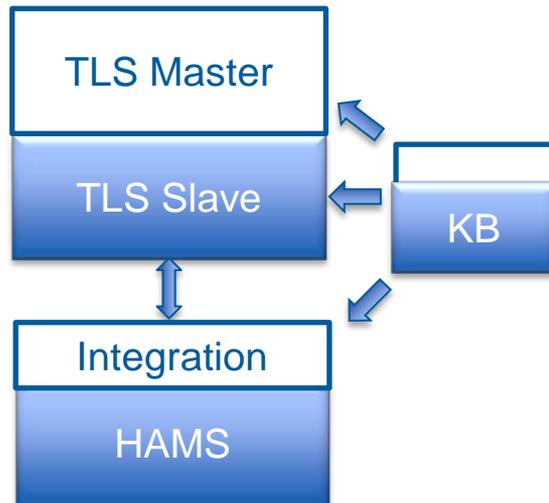




Zusammenfassung



Ausblick



- Implementierung TLS Master
- Erweiterung KB
- Integration in HAMS
- Evaluation der gesamten Migration



Fragen



Tobias Meier

M. Sc.

Wissenschaftlicher Mitarbeiter
Research Assistant

Tel +49 841 9348-6431
Fax +49 841 9348-996431

tobias.meier@thi.de
www.thi.de



**Technische
Hochschule Ingolstadt**

Zentrum für
Angewandte Forschung

Esplanade 10
D-85049 Ingolstadt