Virtualisierung in sicherheitskritischen Systemen

Workshop:
Entwicklung zuverlässiger Software-Systeme

Stuttgart, 30. Juni 2011

Michael Tiedemann, SYSGO AG

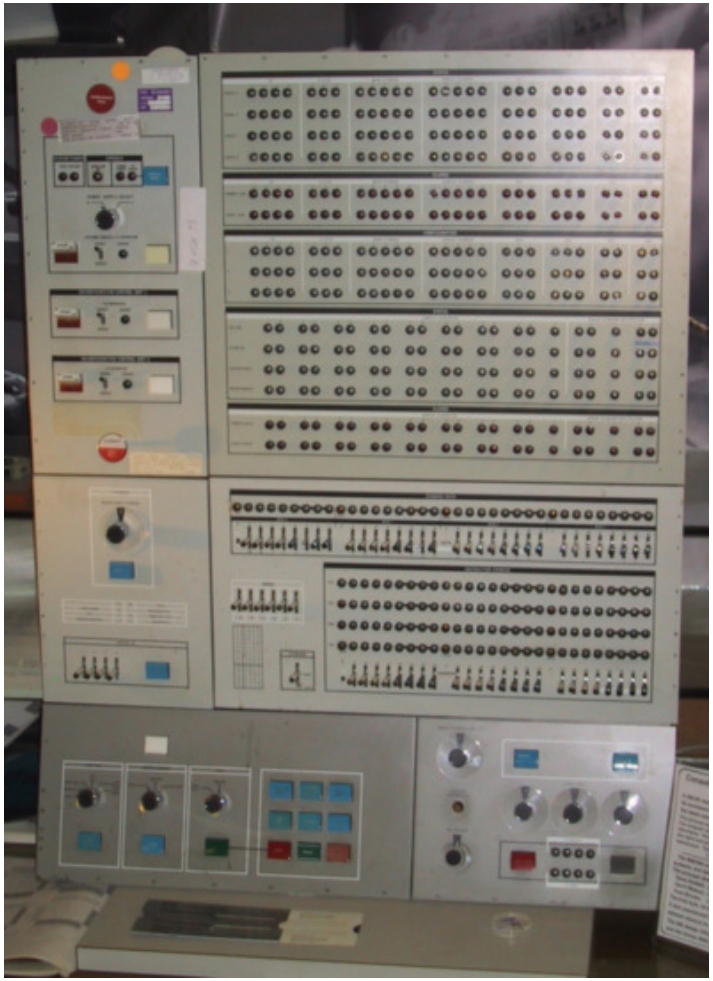
SYSGO
EMBEDDING INNOVATIONS

Agenda

- Begriff der Virtualisierung
- Innovation durch Elektronische Systeme
- Virtualisierungs-Konzepte
- Multi-Core basierte Systeme
- Fazit

Virtualisierung in der Vergangenheit

- Ursprünglich von IBM für die 360/67 Maschinen entwickelt
- Zielsetzung damals:
  Große Leistungsfähigkeit des Systems für verschiedene Anwendungen bereitzustellen
- => „Virtuelle Maschine“ vollständige isolierte Kopie des physischen Systems



Innovation durch elektronische Geräte

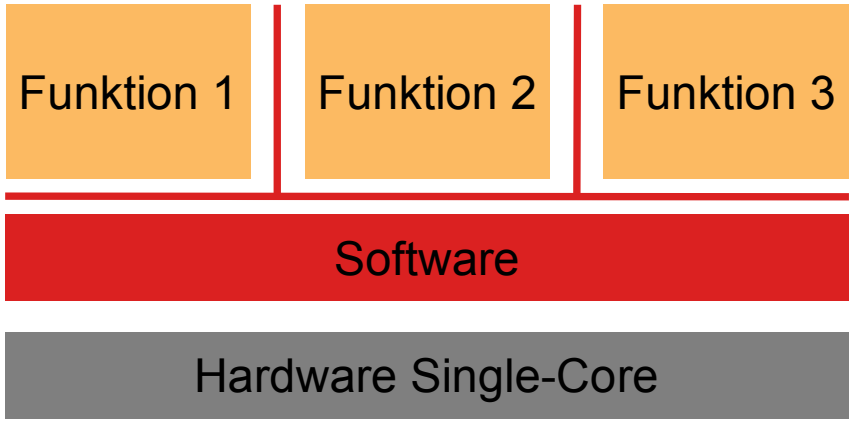
- **Ablösung mechanischer Kontroll- und Anzeigegeräte**
  - Funkgeräte für Kommunikation und Navigation (~ 70 Jahre)
  - Seit ~ 35 Jahren wird Mechanik durch Elektronik ersetzt
- **Elektronik übernimmt sicherheitskritische Funktionen**
  Entwicklung gemäß Zertifizierungsstandards
  - DO-178B für Software Entwicklung
  - DO-254 für Hardware Entwicklung
- **Innovationstreiber**
  - Erhöhung der Funktionalität und Informationsdichte
  - Einfachere Anpassung an verschiedene Konfigurationen
- **Architektur**
  - Ein Gerät pro Funktion (LRU)
  - Kopplung über Kommunikationskanäle
  - Vorteil: Fehlerisolierung, Zuständigkeit, Zertifizierungsprozess
  - Nachteil: Installation, Wartung, Raum, Gewicht, Anzahl Funktionen, Stromverbrauch, End of Life

SYSGO EMBEDDING INNOVATIONS

© SYSGO AG

Innovation durch elektronische Geräte

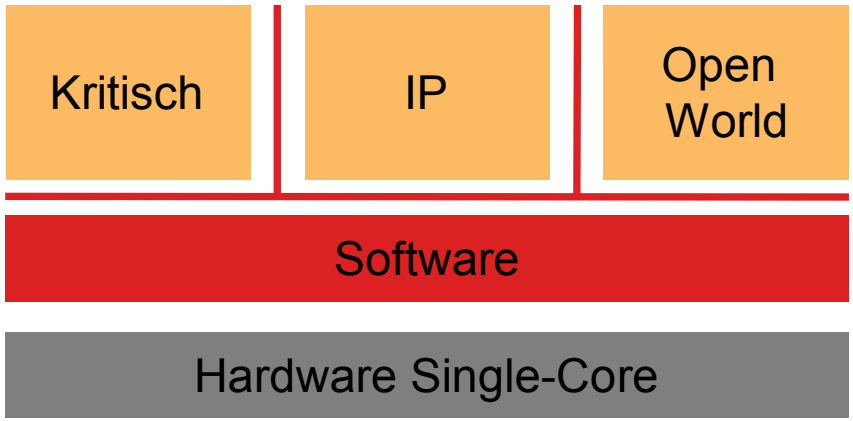
- **Integrated Modular Avionics (IMA)**
- **Erste Entwicklung von Honeywell für die Boeing 777**
  - Aufspaltung von Avionik-Geräten in Basis-Komponenten
    - Host, Stromversorgung, Prozessor, I/O Geräte
  - Operating System und Middleware
    - SW/HW-Trennung
    - SW/SW-Partitionierung (Zeit und Ressourcen)
  - Zuverlässigkeit steigt um eine Größenordnung verglichen mit bisherigen Lösungen



© SYSGO AG

Änderungen zu verteilten Systemen

- Definition von Standards (API)
- Integrationsprozess von HW und SW
- Update/Wartungsprozess
- Businessmodell (Zulieferer)
- IP-Schutz



© SYSGO AG

Zertifizierungsstandards

- **DO-178B**
  - Fokus Luftfahrt
  - Eingeführt 1992 durch RTCA (Radio Technical Commission for Aeronautics)
  - Verwendet durch FAA (US) und EASA (EU)
  - Ursprünglich nur für komplette Systeme
- **IEC-61508 / EN50128**
  - Zielmarkt industrielle Software (Kraftwerke, Chemische Industrie, Bahntechnik)
  - Eingeführt 1997 durch IEC (International Electrotechnical Commission)
  - Erlaubt Zertifizierung einzelner Komponenten
- **ISO 26262**
  - Standard für Automobilindustrie
  - Stark angelehnt an IEC-61508

© SYSGO AG

Innovation durch elektronische Geräte

Flight Deck Boeing 707 (1959)

Innovation durch elektronische Geräte

Flight Deck Boeing 747- 400

Innovation durch elektronische Geräte

Flight Deck Airbus A380

Entwicklung Prozessortechnologie

- **Prozessorleistung durch Erhöhung**
  - Prozessortaktes
  - Caches
- **Derzeit physikalische Grenzen erreicht (Frequenz, Halbleiter, Stromaufnahme)**
- **Entwicklung Dual-Core danach Multi-Core Prozessoren**
- **Homogene und heterogene Multi-Cores**
- **Einsatz leistungsfähiger komplexer Prozessorsysteme**
  - Virtualisierung der Hardware
  - Einhaltung von Echtzeitanforderungen
  - Berücksichtigung von Sicherheitsaspekten

© SYSGO AG

SYSGO EMBEDDING INNOVATIONS

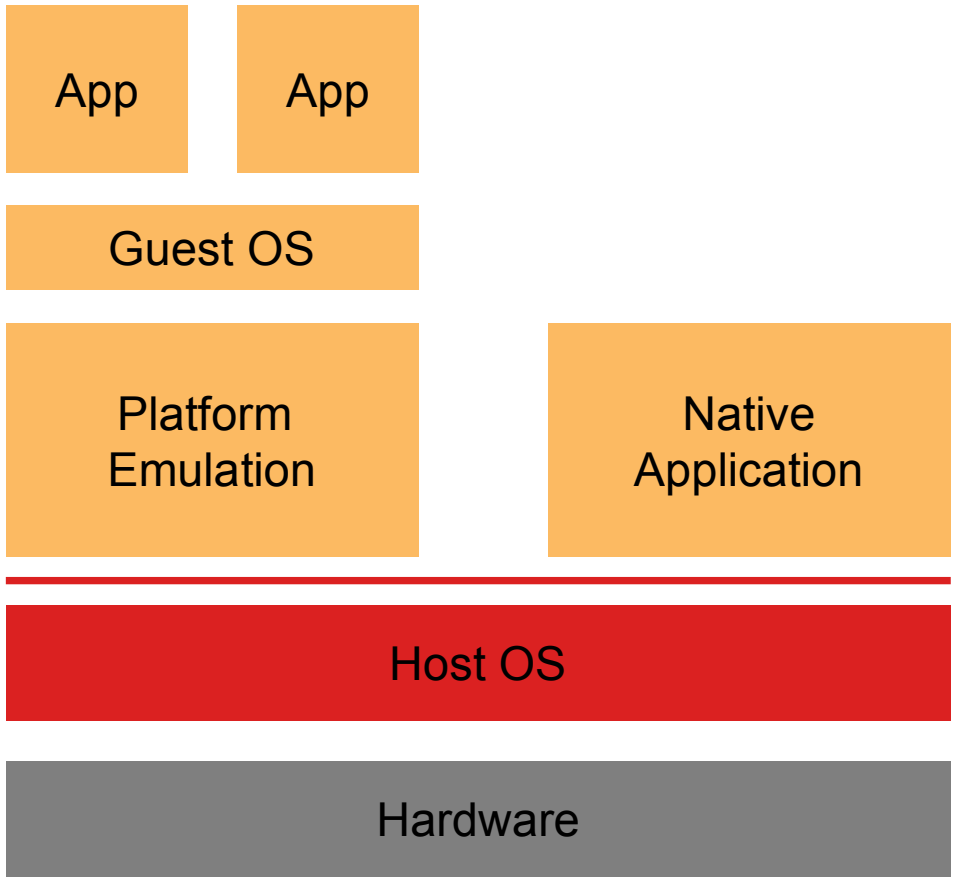
Virtualisierungs-Konzepte
Grundlagen

- **Abstraktion aller Ressourcen eines Computers durch einen Virtual Machine Monitor (Hypervisor)**
  - Kontrolle über alle Betriebsmittel des Prozessors
  - Kontrolle sämtlicher Zugriffe auf physische Betriebsmittel (Ein-/Ausgabe)
  - Isolierung aller VM voneinander
- **Prozessorientierte Virtual Machine**
  - Plattform- und Betriebssystem-unabhängige Ausführungsumgebung
  - Beispiel ‚Java Virtual Machine‘

© SYSGO AG

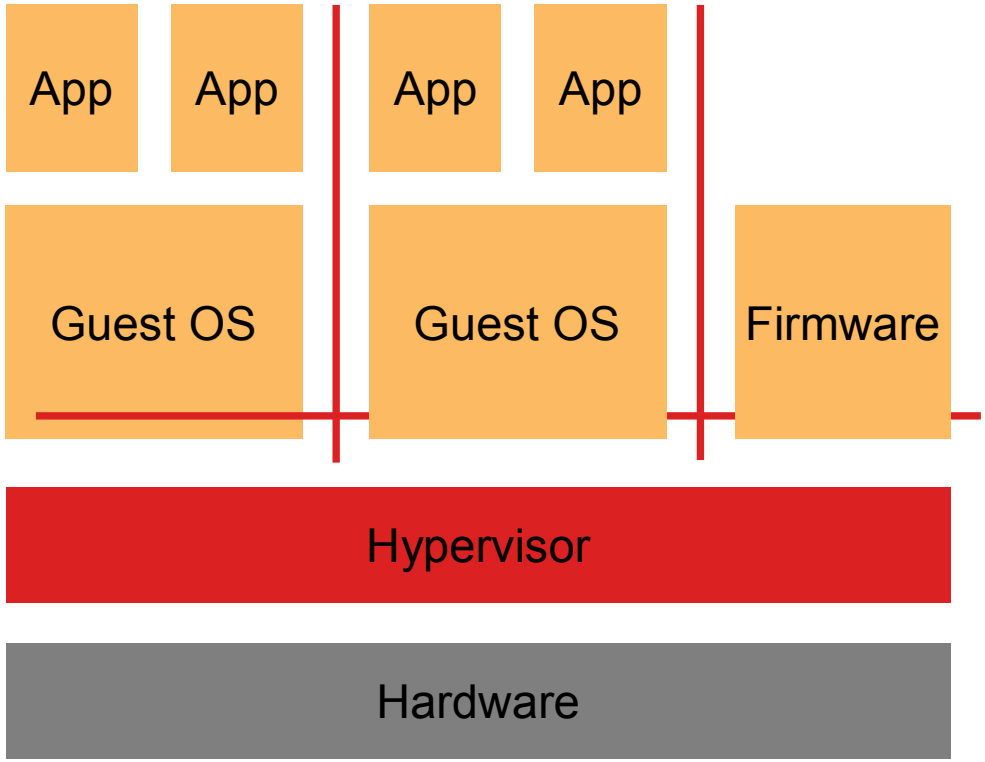
Virtualisierungs Konzepte
User-Mode / SW Virtualisierung - Emulation

- Emulation der Plattform bis auf Register Ebene
- Ausführung des unveränderten Gast-OS und der Anwendungen im User Space (Binärkompatibel)
- Ein Programm, das als Gast unter einem Virtual Machine Monitor zeigt identisches Verhalten (Ausnahme: Zeitverhalten)
- CPU Simulation
- Anwendungsentwicklung
- Cross-Plattform x86 -> PPC
- Beispiele: Qemu, Bochs, GXemul
- NT: Zeitverhalten



Virtualisierungs Konzepte
VM Implementierung - Vollvirtualisierung

- **VMM Ausführung im privilegierten Plattform-Mode**
  - Emulation privilegierter Instruktionen
- **Ausführung des unveränderten Gastbetriebssystems und der Anwendungen im User Space**
- **Intel VT, QorIQ**
- **Kritisch für Zertifizierung**
  - Echtzeitverhalten durch Emulation von Instruktionen (z. B. TLB misses)
  - Black-Box Partitionierung (Hardware-basiert)
  - Einsatz komplexer Hardware Untersuchung hinsichtlich CRI-F08 (CRI = Certification Review Item)

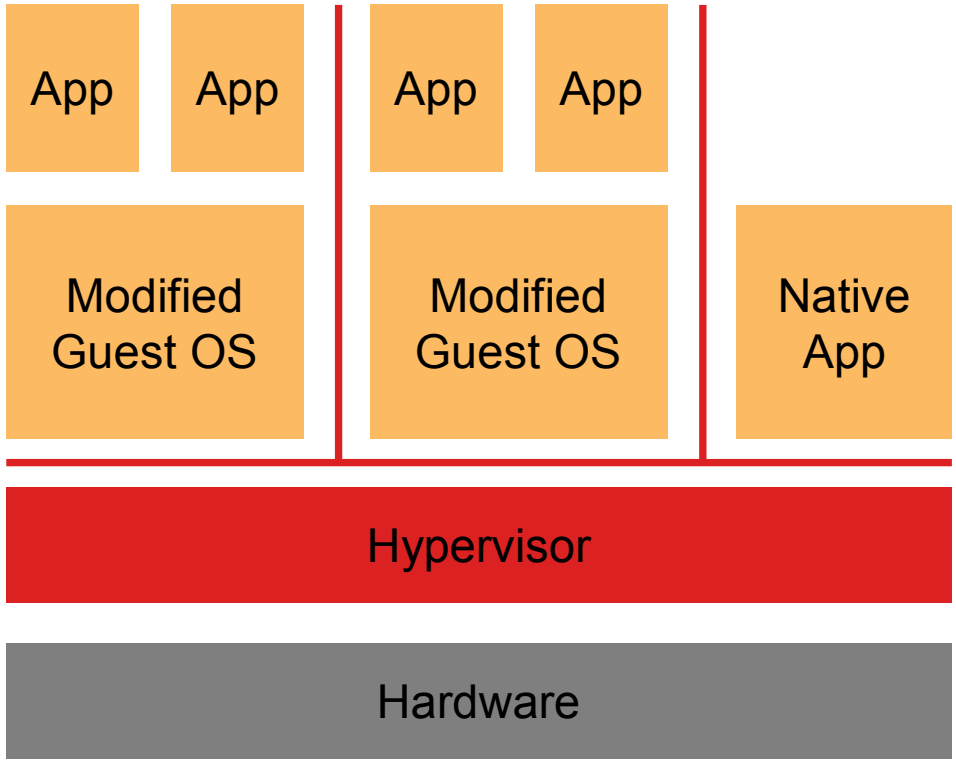


© SYSGO AG

SYSGO EMBEDDING INNOVATIONS

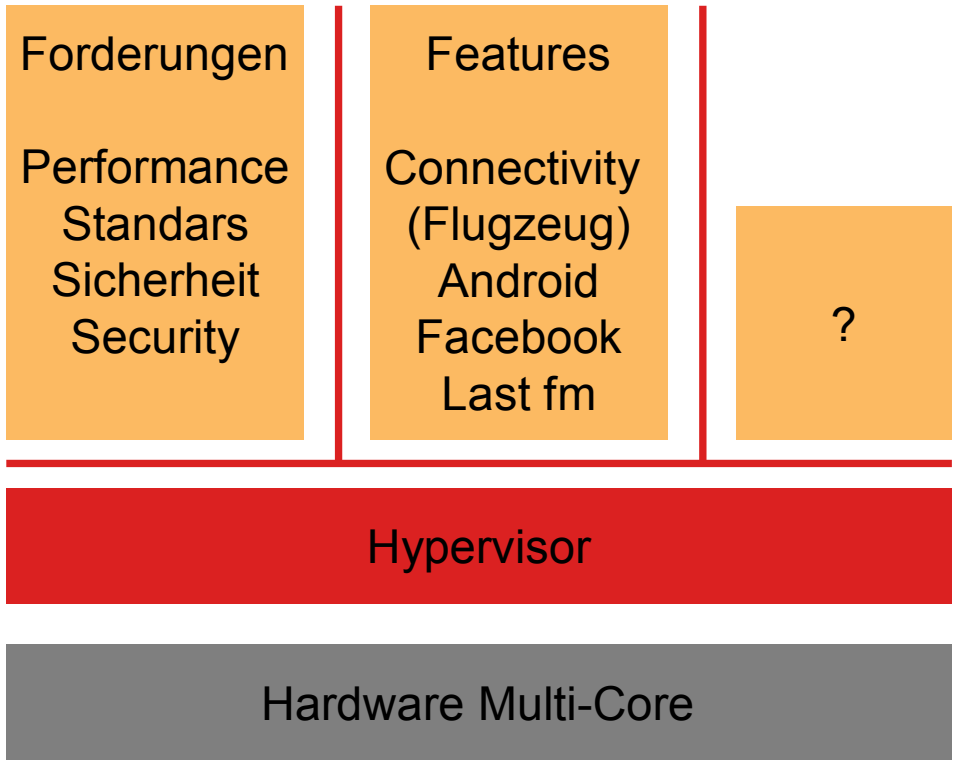
Virtualisierungs Konzepte
VM Implementierung - Paravirtualisierung

- **Keine Binärkompatibiltät des OS**
- **Modifiziertes Gast OS wird auf dem VMM ausgeführt**
- **Ausführung der VMM im privilegierten Plattform-Mode**
  - Abstraktion der Plattform durch OS
  - Teile des Gast OS müssen im Source-Code vorliegen
  - Sensitive Instruktionen wie Memory-Management, Exception-Handling, Interrupt-Handling werden in Hypercalls abgebildet
- **Verwendbar für sicherheitskritische Echtzeit-Anwendungen**
- **NT: Änderungen Gast OS müssen gepflegt werden**



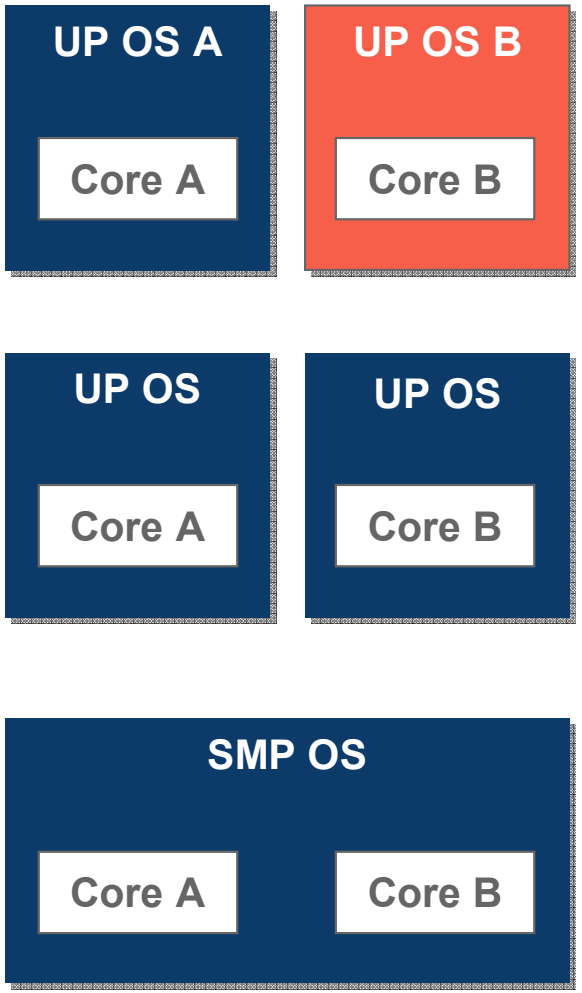
Innovationsdruck

- Innovationszyklus durch Software beschleunigt
- Innovation kommt nicht zwangsweise aus dem Markt
- Markt muss ständig reagieren
- Virtualisierung von Multi-Core Prozessoren



Multi-Core Operating System Modelle

- **Asymmetric Multi Processing**
  - Auf jedem Core läuft ein unterschiedliches OS
  - Lose gekoppelt über IPI und Shared Memory
  - Beide OS müssen sicher sein
- **Semi Symmetric Multi Processing**
  - Auf jedem Core läuft eine Instanz des gleicher OS
  - Gekoppelt über IPI (Inter Prozessor Interrupt) und Shared Memory
  - OS muss sicher sein
- **Symmetric Multi Processing**
  - All Cores werden durch ein einzelnes SMP OS kontrolliert
  - Enge Kopplung durch “resource locks” and IPI Synchronisierung
  - Multi-Prozessor Unterstützung auf Applikations-Ebene
  - OS muss sicher sein



Vergleich AMP und SMP

AMP

Pro

- Einfaches Design der System Software
- Parallele Ausführung der verschiedenen Uni-Prozessor OS

Contra

- Alles OS müssen sicher sein
- Externe Synchronisierung bei Zugriff auf gemeinsame Ressourcen
- Keine Unterstützung für Multi-Core innerhalb einer Applikation
- Schwierig mehr als 2 Cores zu bedienen
- Verteilte Konfiguration

SMP

Pro

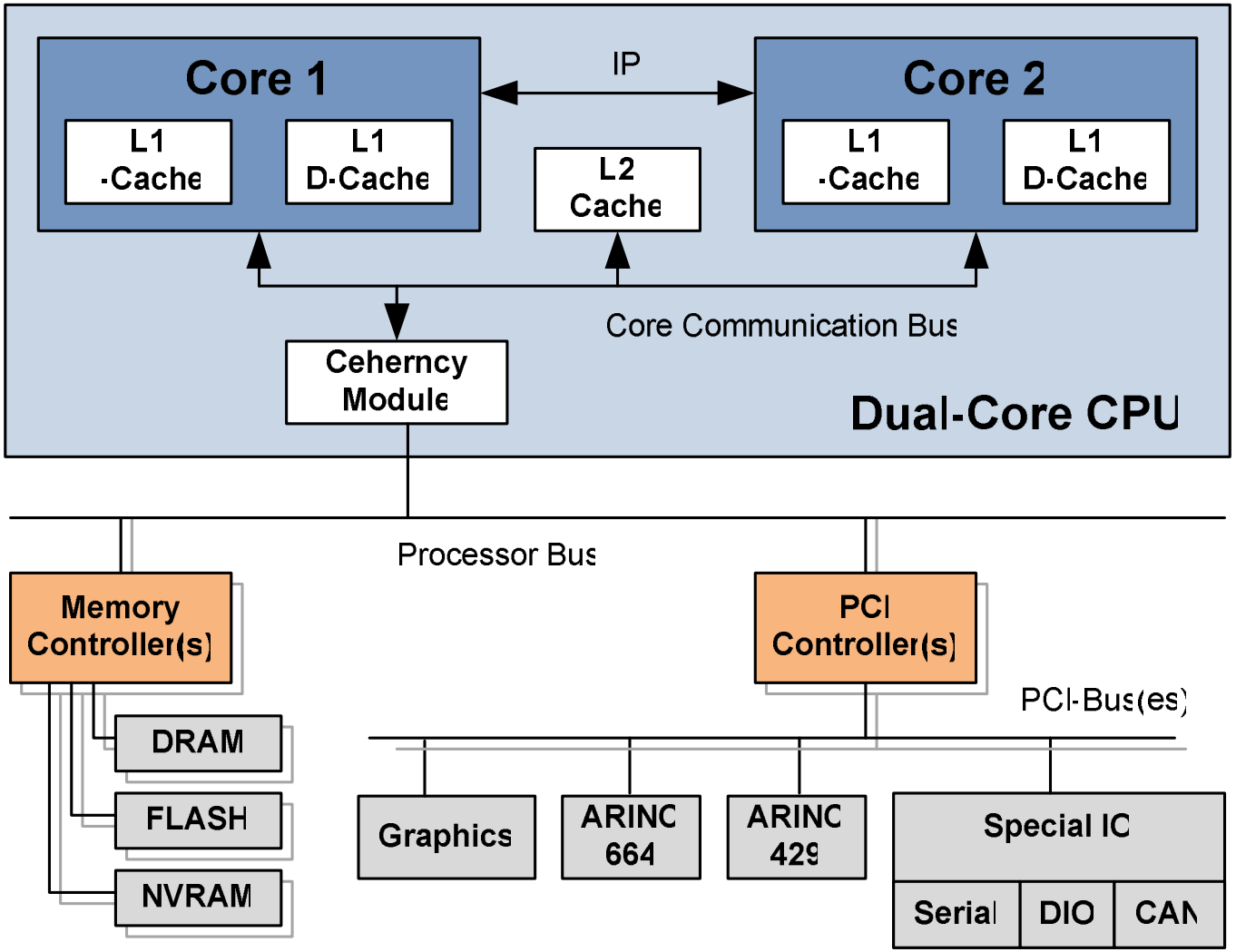
- Nur eine sichere System-Software
- Bessere Kontrolle der CPU-Aktivitäten
- Unterstützung von Multi-Core Processing auf Applikations-Ebene
- Einfachere Synchronisierung zwischen Partitionen
- Homogene Konfiguration

Contra

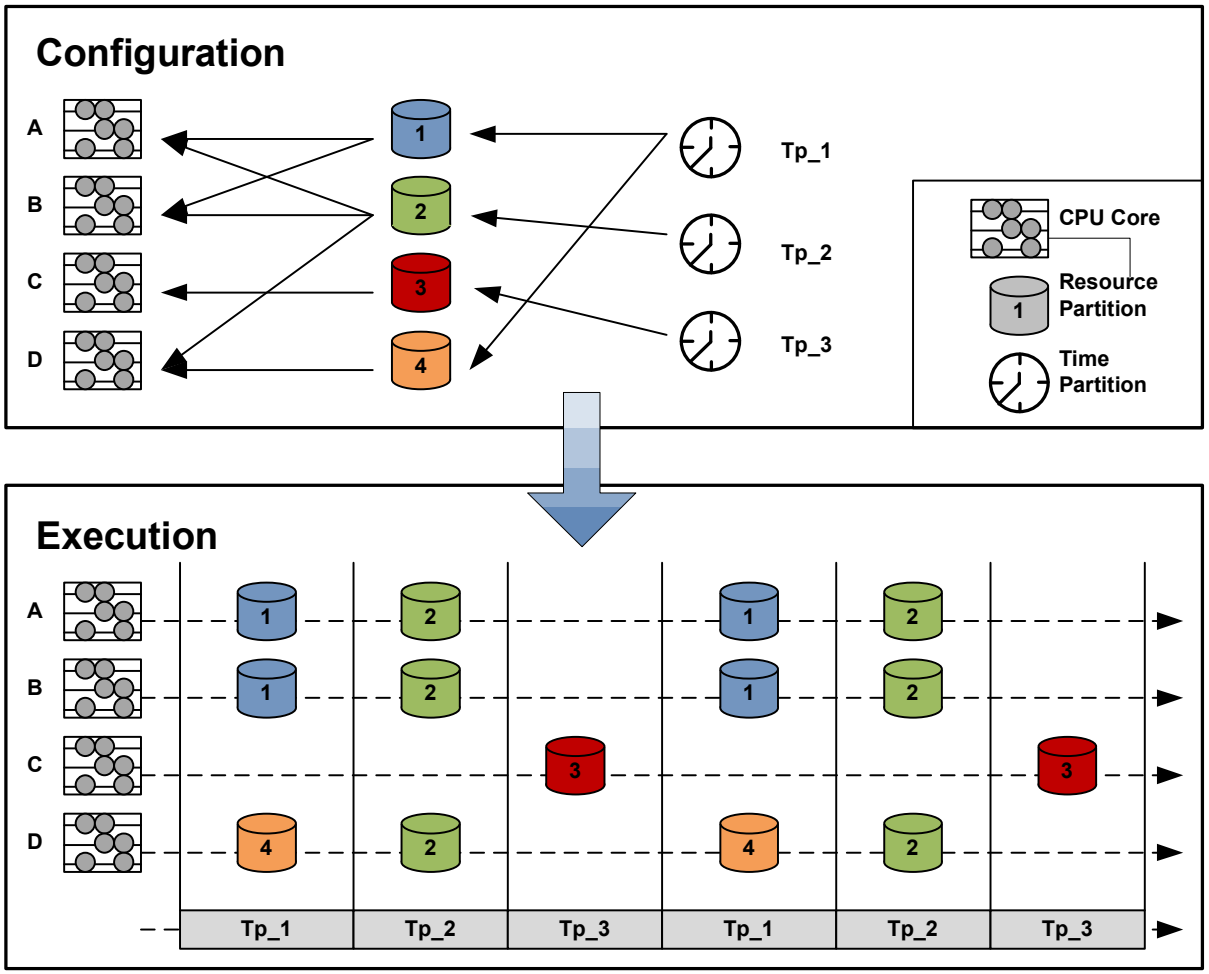
- Größere Komplexität des SMP OS
- Schlechtere Performance verglichen mit AMP bei lose gekoppelten Applikationen

© SYSGO AG

Multi-Core basierte Plattform



Beispiel für SMP basierte Lösung



- Hohe Rechenleistung erforderlich

- Sicherheitskritisch
- Isolierung läuft nicht parallel mit unkritischen Anwendungen

Sicherheitsbetrachtung Single-Core PPC

- **„Einfache“ Cache- und Pipeline-Architekturen**
- **ECC (Error Correction Codes) erkennen und korrigieren bei Single- oder Multi-Bit-Fehlern in Registern und Caches**
- **Unterstützt Lock-Step-Modus, der die Verbindung von zwei oder mehreren Prozessoren mit einem Voter erlaubt (2 oo 3)**
- **Sehr resistent gegen Single-Event-Upset (SEU)**
- **Design-Dokumente zur Unterstützung des Zertifizierungsprozesses sind verfügbar**
- **Häufige Verwendung in Avionik-Plattformen**
  - Sehr große Erfahrung durch Wartung und Updates vorhanden
  - Systematische Fehler mit sehr viel geringerer Wahrscheinlichkeit

© SYSGO AG

Sicherheitsbetrachtung Multi-Core PPC

- **Ein Multi-Core Prozessor ist häufig Nachfolger eines hochentwickelten Single-Core Prozessors**
- **Komplexe Instruktions-Pipelines, Multi-Level-Caches (Daten-Kohärenz)**
- **Gemeinsame Ressourcen wie Timer etc.**
- **Nichtdeterministisches Verhalten z. B.**
  - Refill Cache, TLB
  - Bus-Arbitrierung
- **Neueste Technologie,**
  Erhalt der Design-Dokumente höchst unwahrscheinlich

© SYSGO AG

Fazit

- Multi-Core CPU werden u. U. zur Erreichung zukünftiger Anforderungen unumgänglich
- System-on-Chip Lösungen mit fehlender Sicherheitseinstufung erschweren den Zertifizierungsprozess
- Lösung mit Multi-Core Plattform muss die störungsanfälligen Kanäle je nach Anforderung lösen
- Parallele Ausführung verschieden kritischer Anwendungen ist problematisch
- Auf Kosten nicht verwendeter Bandbreiten kann Determinismus von Single-Core Prozessoren auch durch Multi-Core Prozessoren erreicht werden
- Durch steigende Processing-Bandbreite können zusätzlich Redundanzkonzepte die Safety erhöhen
- Gemeinsame durch BMBF geförderte Anstrengung von Luftfahrt, Automobilindustrie, Bahntechnik und Halbleiterherstellern zur Lösung der Herausforderung

SYSGO EMBEDDING INNOVATIONS

© SYSGO AG

Vielen Dank für Ihre Aufmerksamkeit