

Rekonfigurierbare System-on-Chip Architekturen



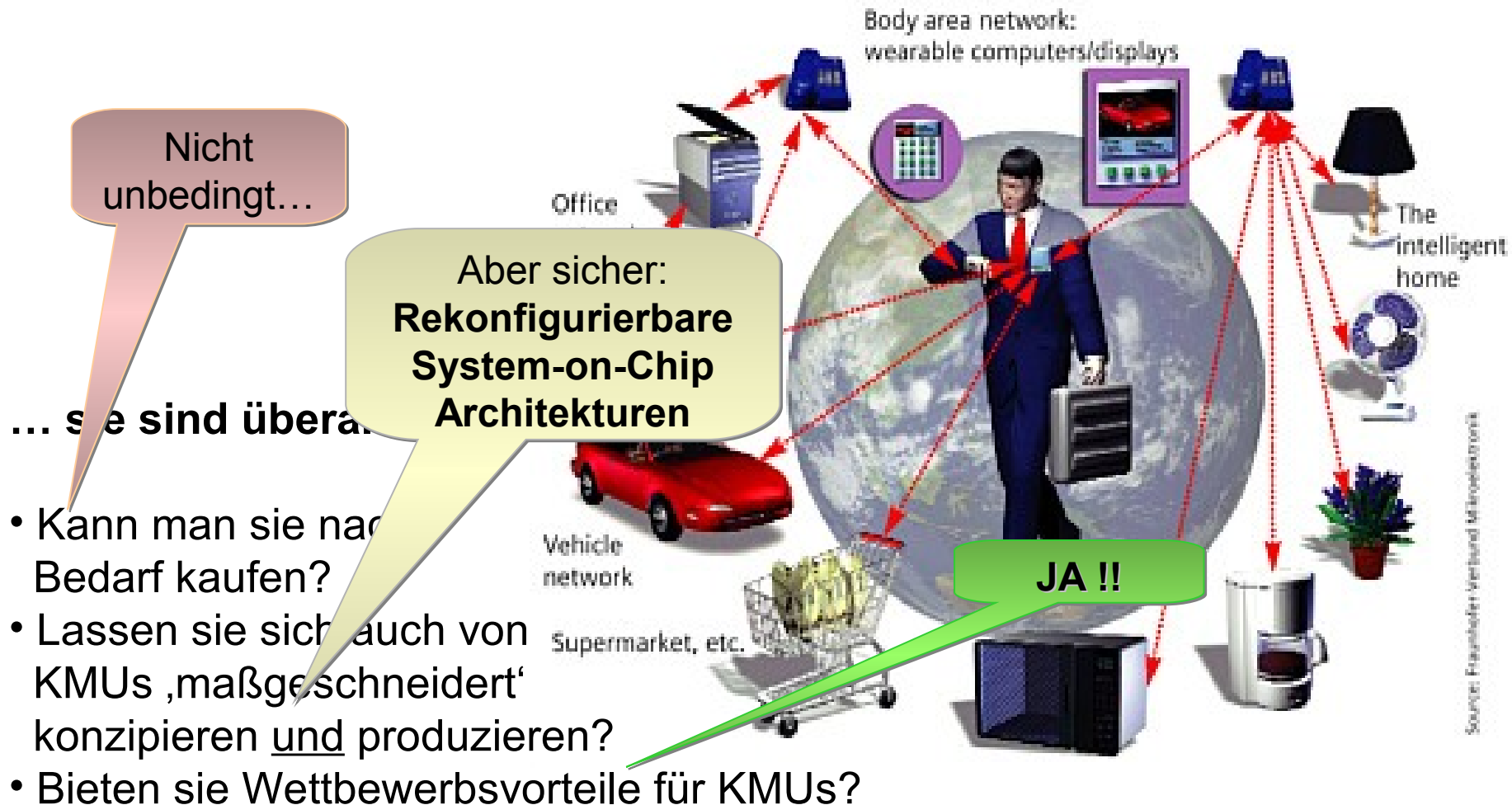
TECHNISCHE
UNIVERSITÄT
DARMSTADT

Chancen für mittelständische Unternehmen

Sorin A. Huss

Integrierte Schaltungen und Systeme
Fachbereich Informatik
Fachbereich Elektrotechnik und
Informationstechnik

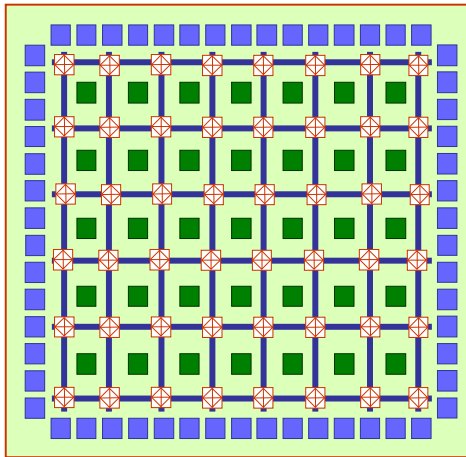
Technische Universität Darmstadt



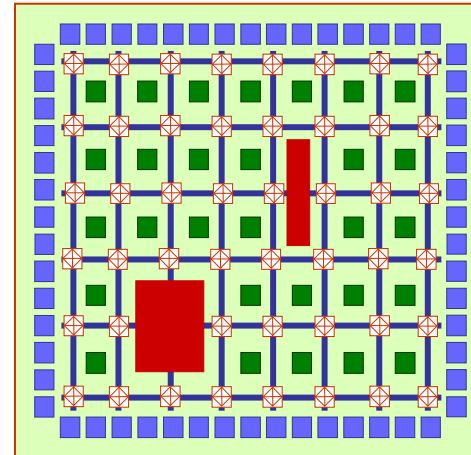
Was wird rekonfiguriert?



Feingranulares FPGA



Grobgranulares FPGA

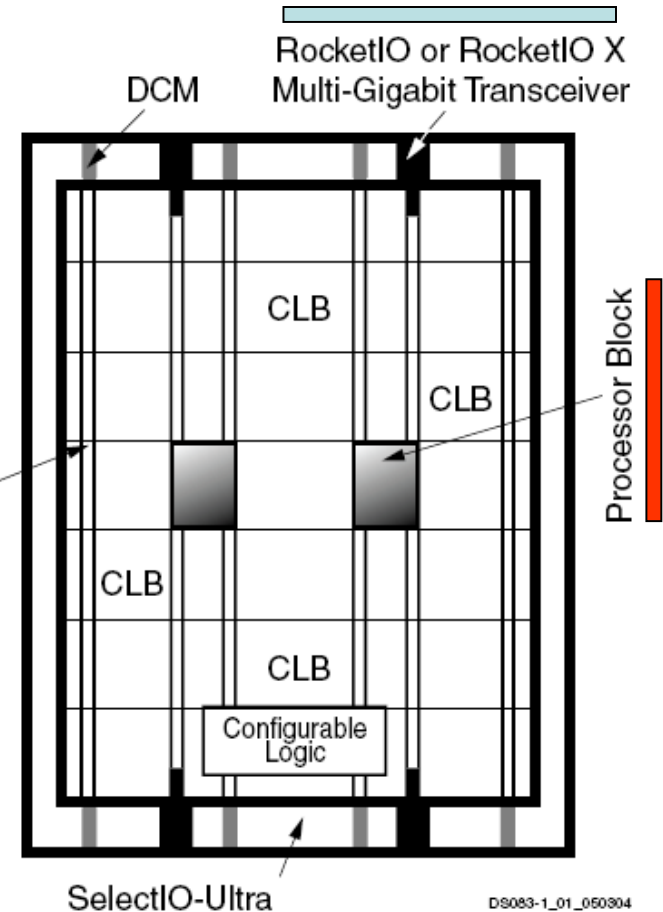
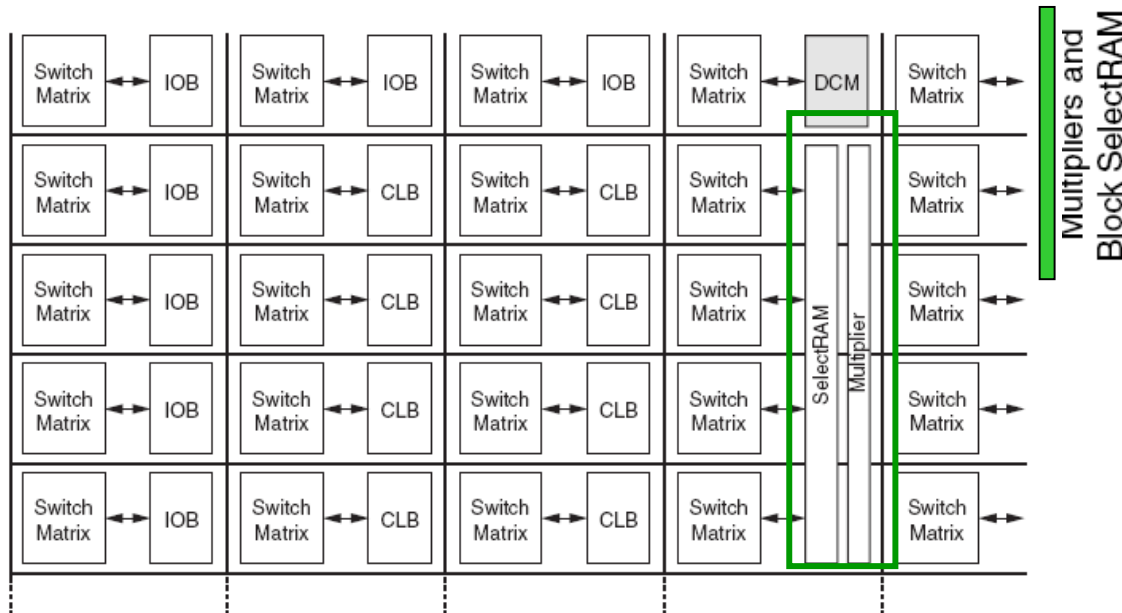


Zur Realisierung von *System-on-Chip* (SoC) Architekturen stellen moderne FPGAs grobgranulare Module zur Verfügung, die mit dem Rest des FPGA über die Verdrahtungsressourcen verbunden werden:

- Eingebettete Prozessoren
- Speicherblöcke
- Schnittstellen-Controller
- Dedizierte Multiplizierer

Beispiel: Xilinx Virtex II Pro – High End RecSoC

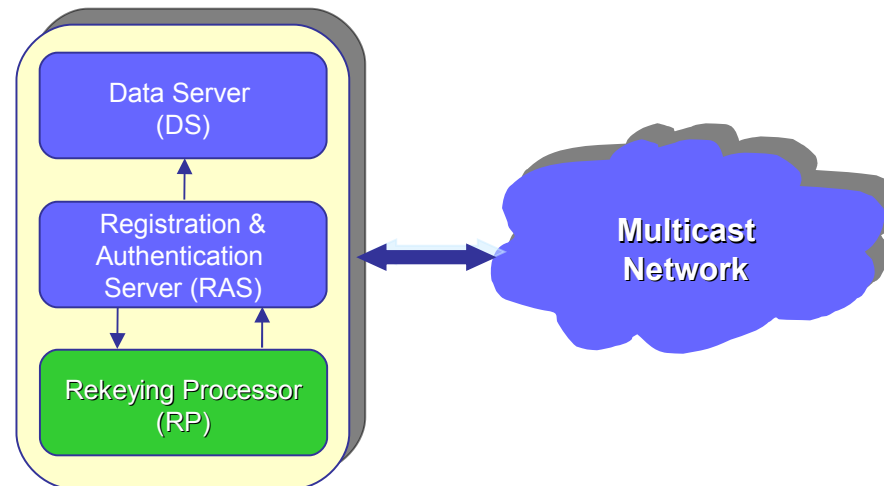
- *BlockSelectRAM*: 18 KBit
- *Multiplier*: 18*18 Bit
- *DCM*: Digital Clock Manager
- *RocketIO*: Multi-Gigabit Transceiver
- *ProcessorBlock*: IBM PowerPC 32 Bit (*Hard-Core*)



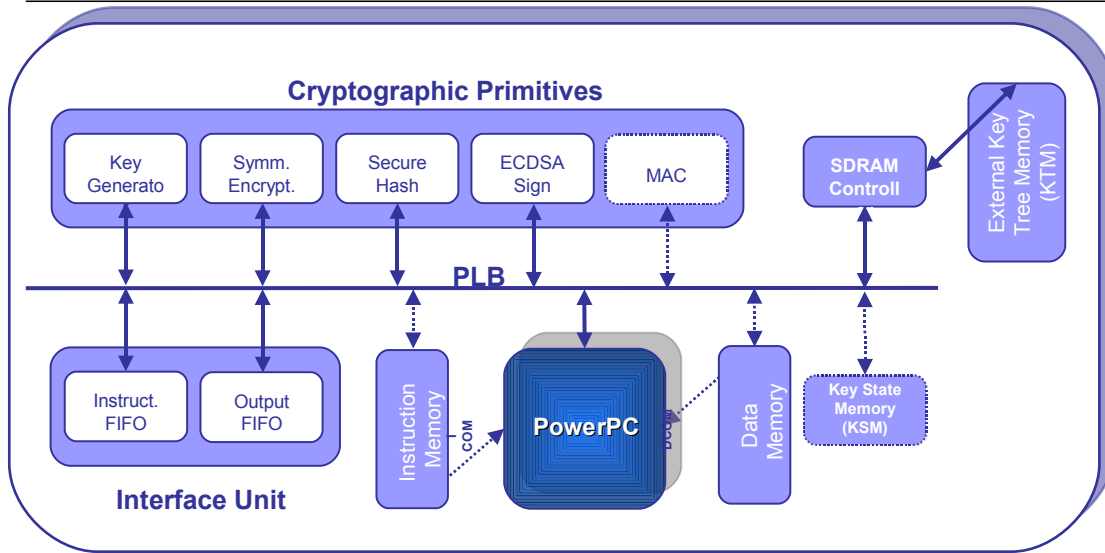
Anwendung 1: Secure Multicast beim Pay-TV



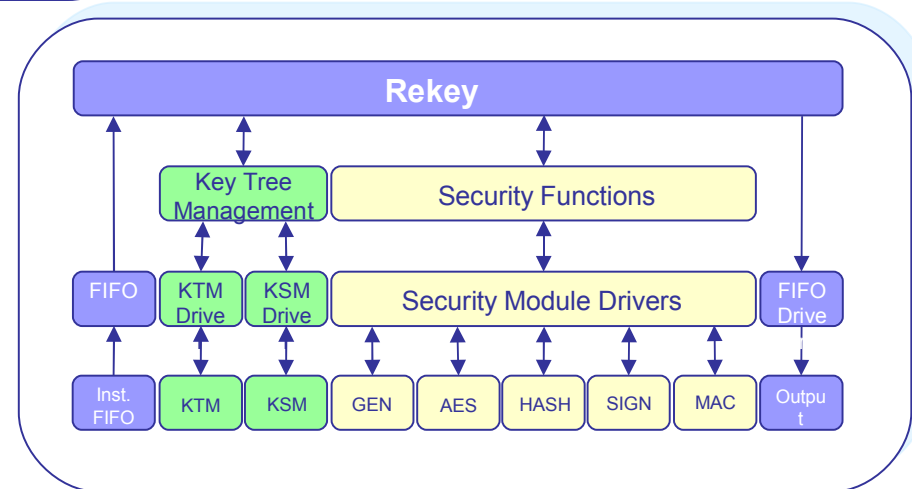
- Der *Rekeying Processor* (RP) ist ein rekonfigurierbarer Prozessor, der als *Co-processor* in einer Serverumgebung eingesetzt werden kann.
- Der RP empfängt Rekeying-Anfragen vom RAS, führt sie aus, generiert Rekeying-Nachrichten und liefert sie zurück.
- Der RAS übernimmt die Kommunikationsaufgabe und leitet die Rekeying-Nachrichten an die Internet Pay-TV Teilnehmer weiter.
- Der RP liefert den neuen Gruppenschlüssel an den DS, der diesen zur Verschlüsselung der Videodaten (*Content Encryption*) verwendet.



Rekeying Prozessor: HW- und SW-Architektur



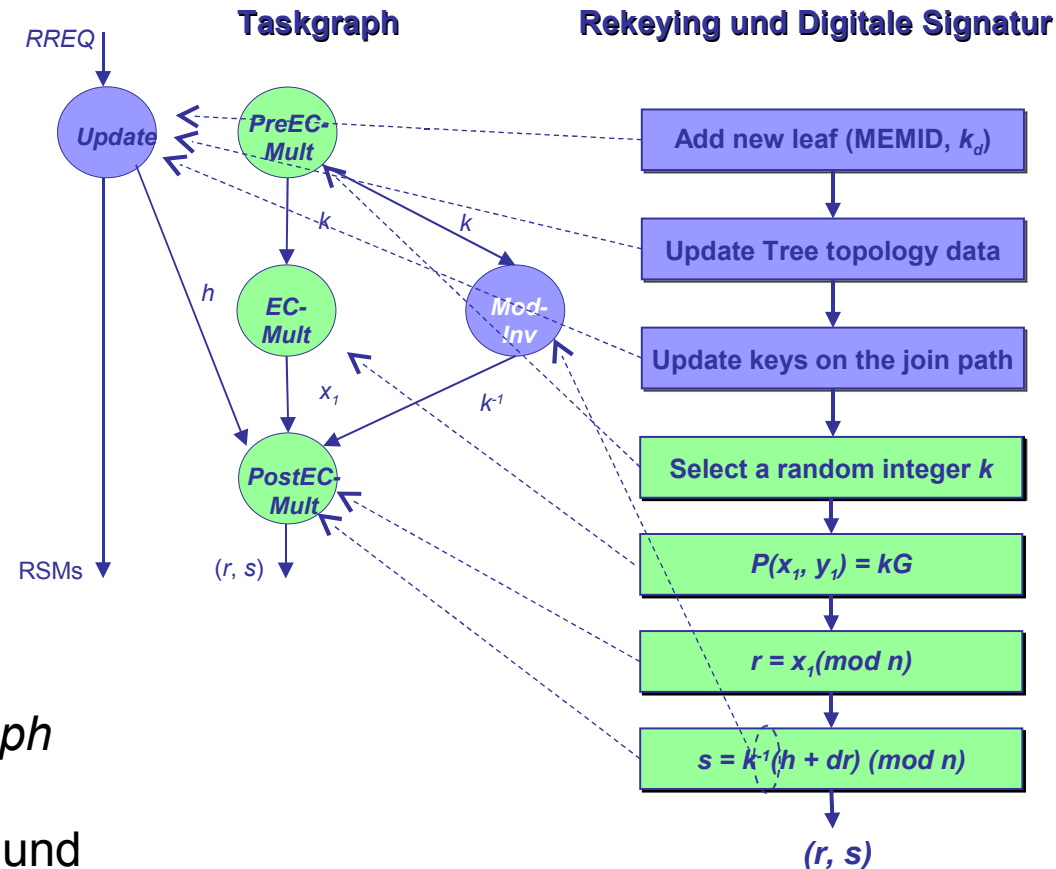
Softwarearchitektur



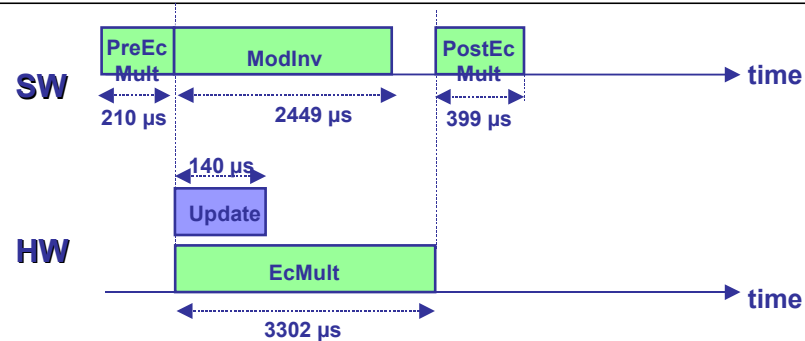
HW/SW Partitionierung der *join member* Operation

HW/SW Realisierung:

- Notation des Algorithmus, so dass die Datenabhängigkeiten und die Nebenläufigkeit erkennbar werden.
- Geeignete Repräsentation: *Taskgraph*
- *Bindung* der Tasks auf Ressourcen und deren *Ablaufplanung* lassen sich so bestimmen und evaluieren.



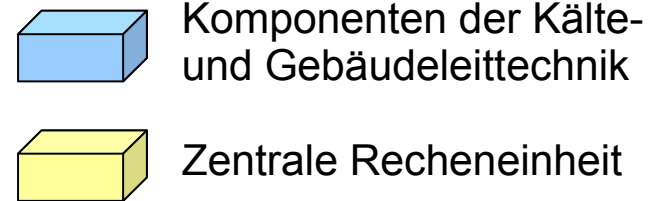
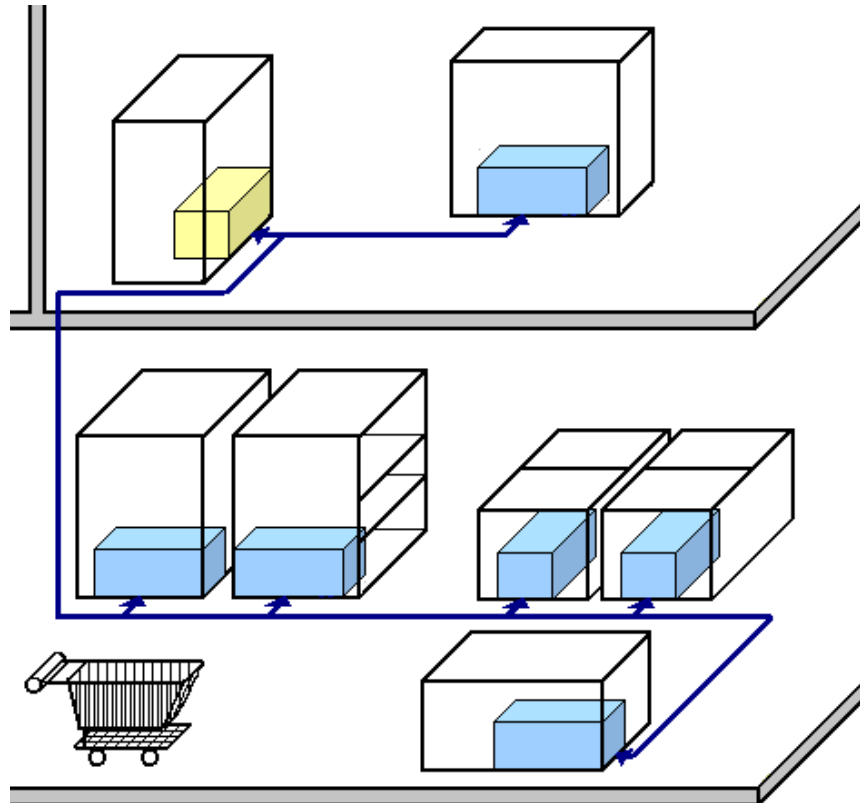
Ergebnisse



	[Am05]	[Wo00]	HiFlexRP (Virtex II Pro)
Encryption	n/a	DES-CBC	AES-128
Key generation	n/a	n/a	ANSI X9.17
Secure hashing	n/a	MD5	Meyer hash
Digital signing	RSA-1024	RSA-512	ECDSA-256
Group size	50	8,192	131,072
Execution time	up to 640 ms	12 – 16.2 ms	3.91 ms
Measurement conditions	Total elapsed time for disjoint	Average-case disjoint, server time only	Worst case disjoint, HiFlexRP processing time only

Anwendung 2: Leitechnik im Supermarkt

Supermarkt



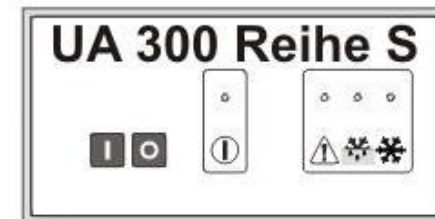
- **Ausgetauschte Daten**
 - Messwerte
 - Parameter
 - Alarmsignale
- **Kommunikationssystem:**
 - CAN-Bus

Y. Scholl: *Entwicklung und Realisierung eines CAN-Bus Protokollanalytors mit Hilfe von IP-Cores*. Diplomarbeit, FB Informatik, TU Darmstadt, 2007 (in Kooperation mit Fa. Eckelmann AG, Wiesbaden)

Steuerung von Kühlstellen

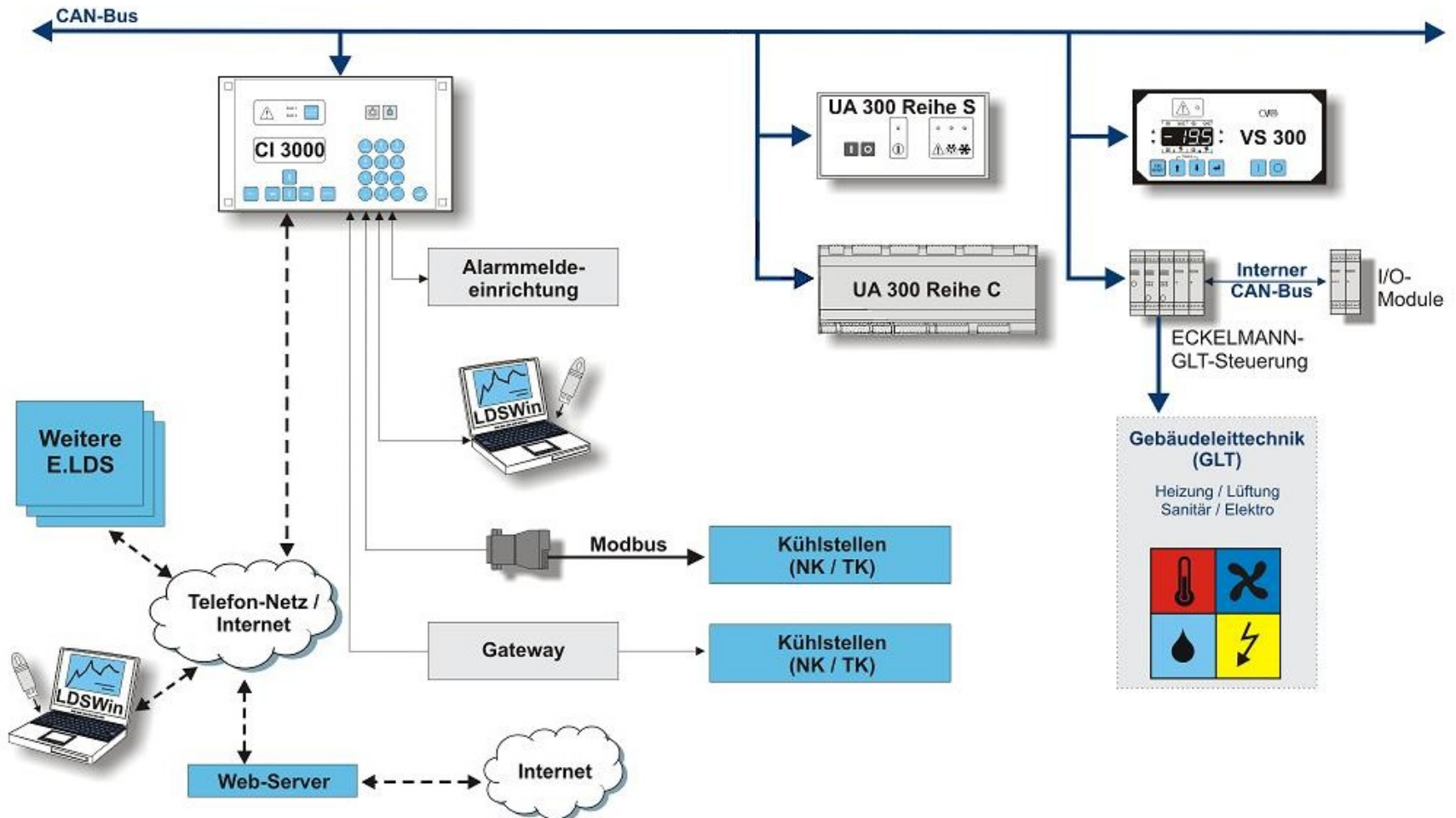


Kühlstellen

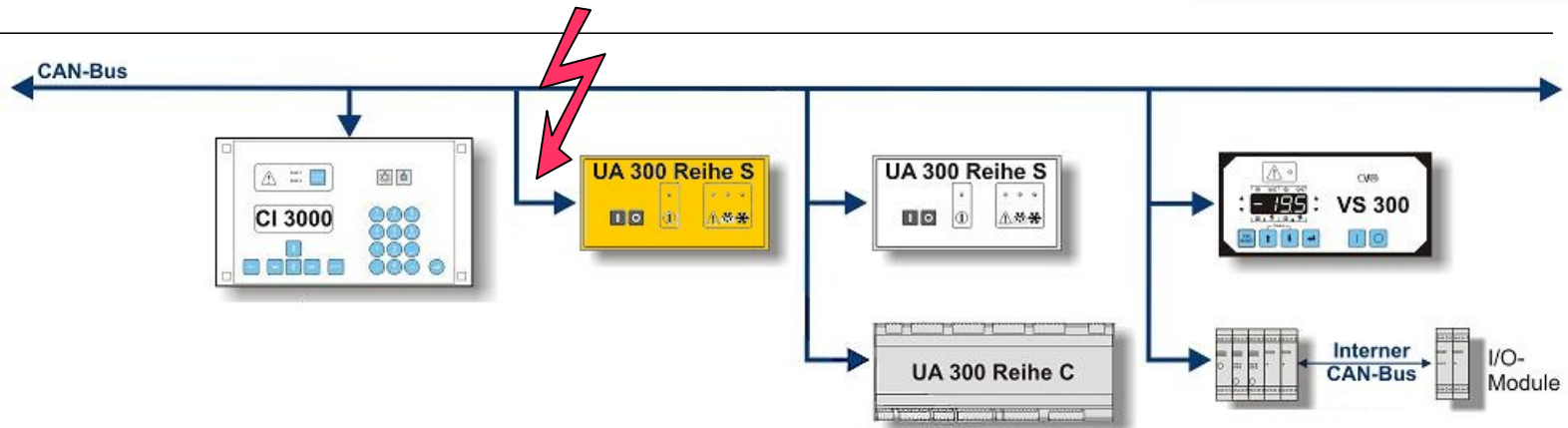


Kühlstellenregler

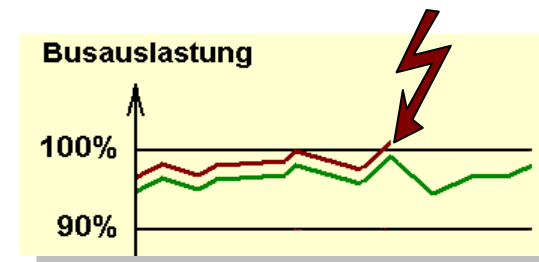
IT-Netzwerk für Gebäudeleittechnik



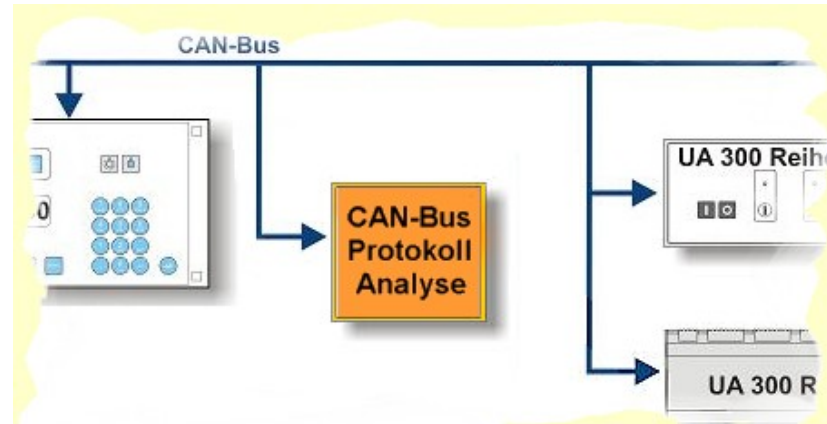
Mögliche Fehlerursachen im Netzwerk



- Neuer Busteilnehmer falsch konfiguriert?
- Bus vorher an Belastungsgrenze?
 - Busauslastung messen
- Viele Fehlernachrichten?
 - Fehlerrate messen
 - Quelle der Fehlernachrichten ermitteln

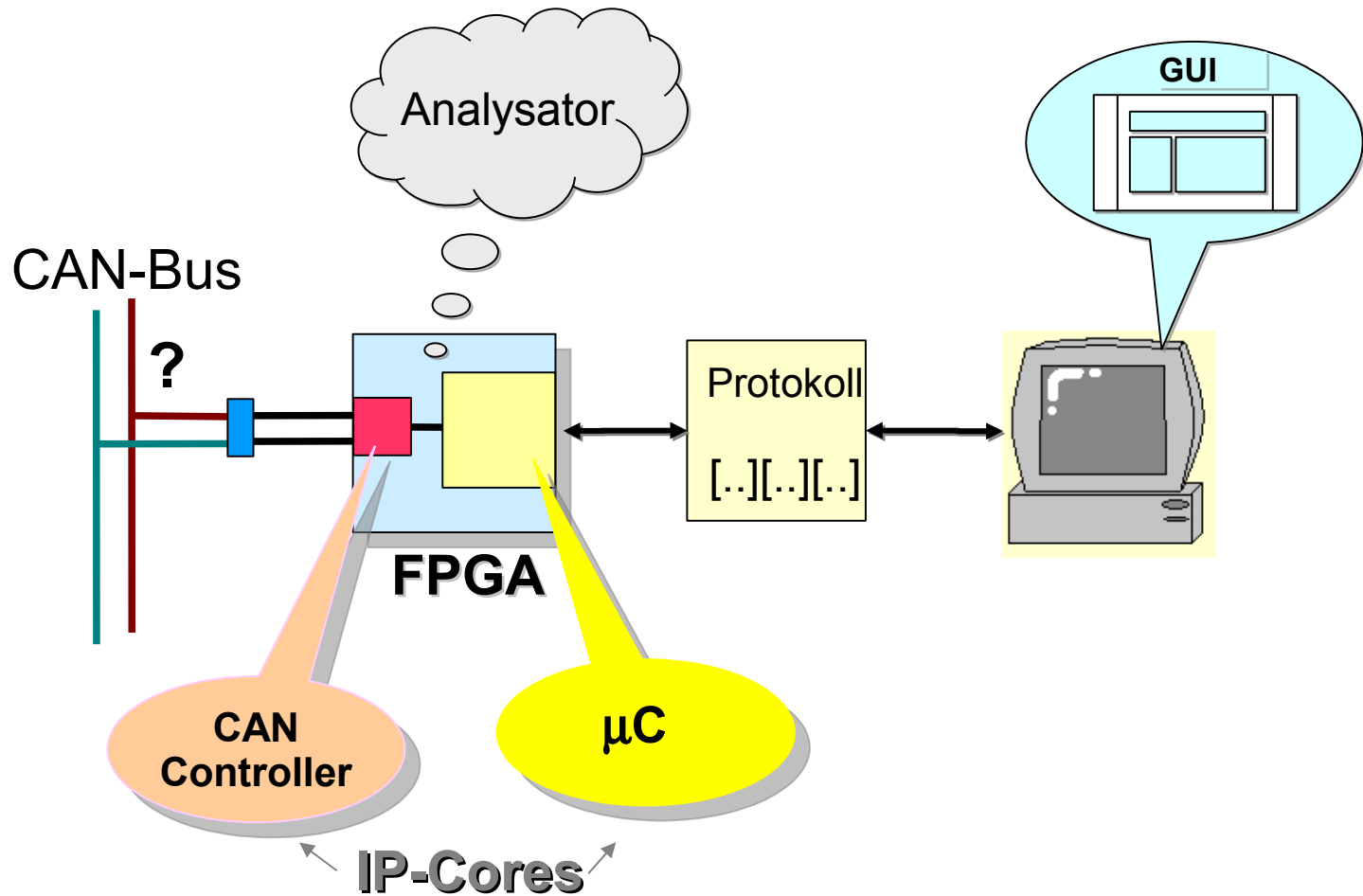


Lösung: Dedizierter CAN-Bus Analysator



- **Entwicklung eines CAN-Bus Protokoll-Analysators**
 - *Zweck:* Ermittlung von signifikanten Kenngrößen als Hinweis auf die Fehlerursachen
 - *Motivation:* Wesentlich geringere Kosten pro Gerät im Vergleich zu kommerziellen Produkten
- **Analyseumfang**
 - Fehlererkennung
 - Fehlerursachenbestimmung
 - Systemkenngößen

CAN-Bus Analysator - als System-on-Chip



IP: Intellectual Property



8051 μ C

- 4x8 I/O
- 1 UART
- 2 16 Bit Timer
- 2 ext. Interrupts
- 8 kB ROM
- 2 kB RAM

**Implementierung
als sog.
Soft-Core
Prozessor**



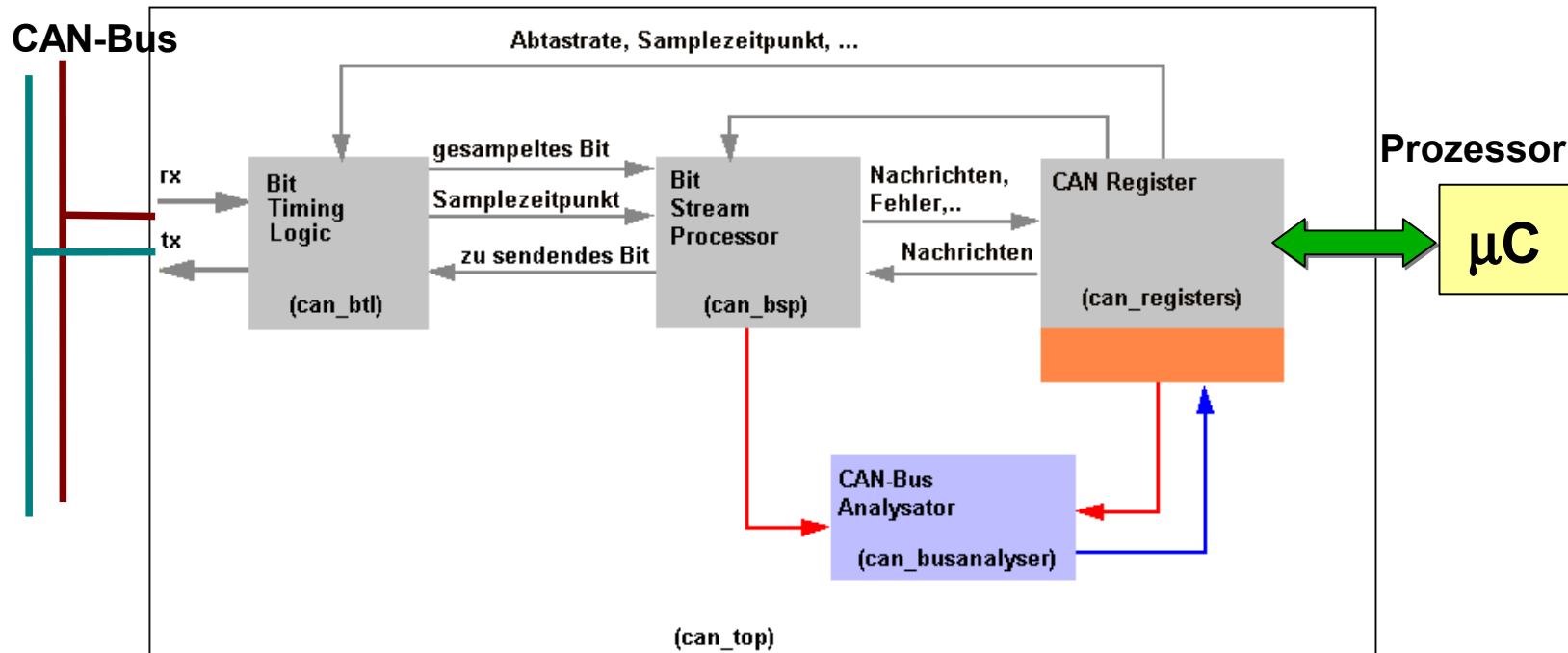
SJA 1000

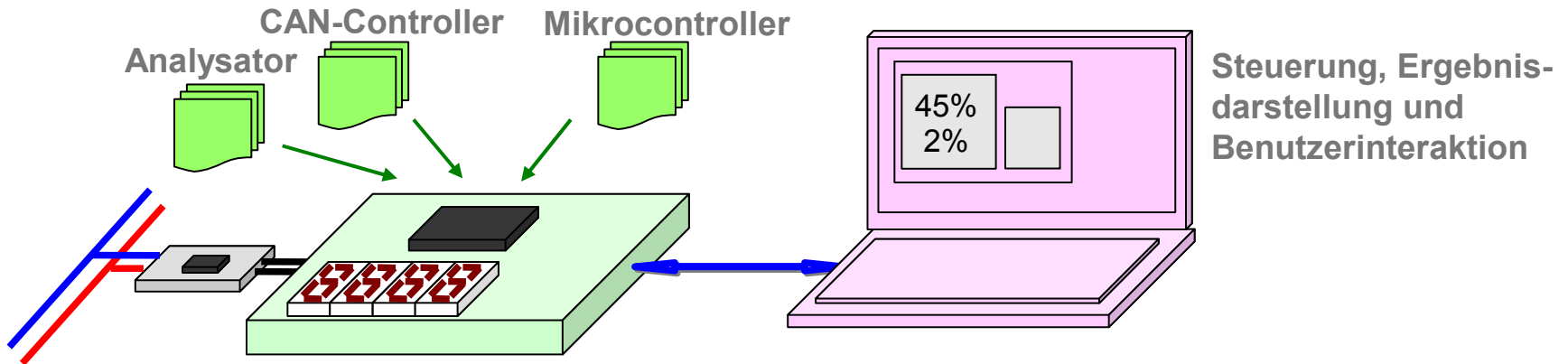
- **Standard CAN-Controller**
(Philips)
- **32 Registerbytes**
- **8 Interruptflags**
- **Nachrichtenfilter**
- **64 Byte Empfangs-FIFO**

CAN-IP Core



CAN-Controller





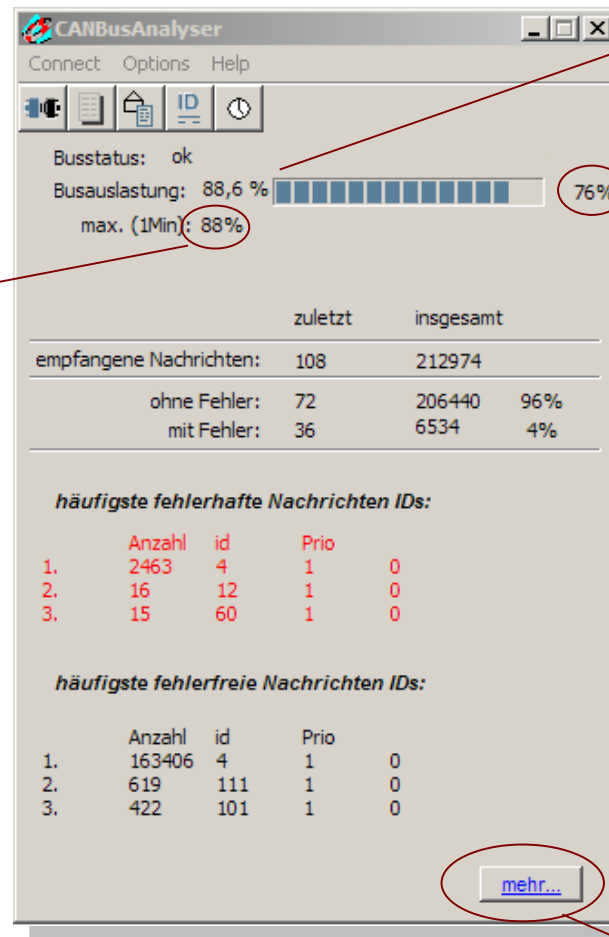
Systemkomponenten

- FPGA-Konfiguration aus CAN- und Mikrocontroller
- Erweiterung des CAN-Controllers zum Sammeln von Analysedaten
- Platine zur Anbindung an den CAN-Bus
- Protokoll zur Kommunikation zwischen Mikrocontroller und PC
- Software
 - Mikrocontroller
 - PC

Benutzungsoberfläche auf PC



Maximalwert,
letzte Minute



Aktueller Wert

Mittelwert,
letzte Minute

Nachrichtenzähler

Häufigste
Nachrichten-IDs
im Netzwerk

Aufrufen der
kompletten ID-Liste

- **Dediziertes CAN-Bus Diagnosewerkzeug**
 - **Qualitätsmerkmale**
 - **Mögliche Fehlerquellen**

- **Direkt darstellbare Qualitätsmerkmale**
 - **Busauslastung**
 - **Fehlerrate**

- **Lokalisierung möglicher Fehlerquellen**
 - **Gespeicherte IDs**
 - Fehlerfreie Nachrichten
 - Fehlerhafte Nachrichten
 - **Fehlerraten von einzelnen Bussegmenten**

Chancen für mittelständische Unternehmen:

- Schnelle Integration von HW- und SW-Komponenten
- Einheitlicher und dadurch verlässlicher Entwurfsablauf
- Nutzung günstiger, zum Teil kostenloser IP-Module
- Geringere Prototyp- und Produktionskosten
- Produktpflege durch Anpassungen sowohl der SW als auch der HW nach Auslieferung beim Kunden
- Hohe Nachbausicherheit der Produkte