



HS Technische Informatik

Erlangen Slot Machine Architektur eines dynamisch rekonfigurierbaren Rechnersystems

Johannes Görner

Johannes.Goerner@mailbox.tu-dresden.de

Dresden, 02.07.2008

ESM

- Einführung
- Architektur
- Ausblick

- entwickelt an der Friedrich-Alexander-Universität Erlangen Nürnberg
- FPGA-basierende Plattform
- Testplattform im Rahmen des DFG SPP „Rekonfigurierbare Rechnersysteme“
 - Sprachen-/Modelle
 - Analyseverfahren
 - Architekturen und Anwendungen
 - Entwurfsverfahren
 - Compiler, Echtzeitbetriebssysteme, Synthesewerkzeuge
 - Methoden zum Entwurf und zur Werkzeugentwicklung
 - Mechanismen der Selbstrekonfiguration/Adaption
 - Verifikation

- FPGAs bestehen aus:
 - Konfigurationsspeicher
 - Logik-Blöcken / Funktions-Blöcken
 - Verdrahtungsleitungen
- Inhalt der Konfigurationsspeicherzellen des Konfigurationsspeichers bestimmt:
 - Funktion der Logik-Blöcke
 - Verdrahtung der Logik-Blöcke/Funktions-Blöcke
- Nahezu jede digitale Schaltung lässt sich auf diese Struktur abbilden

kleinstes Element

Ebene

Verdrahtungsknoten

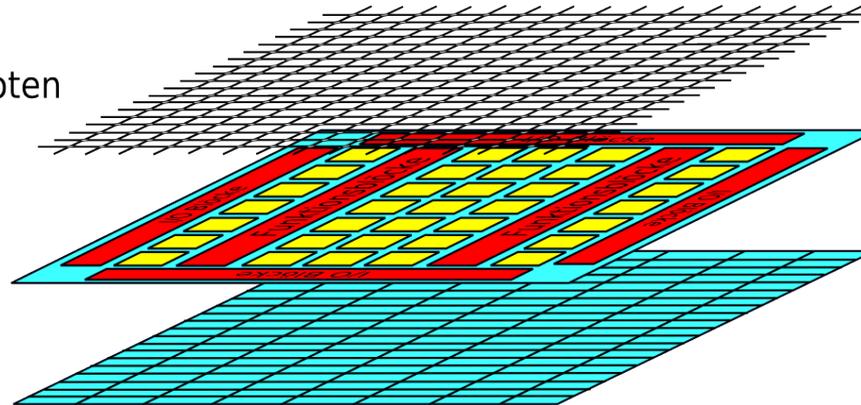
Verdrahtung

Logik-Block/
Funktions-Block

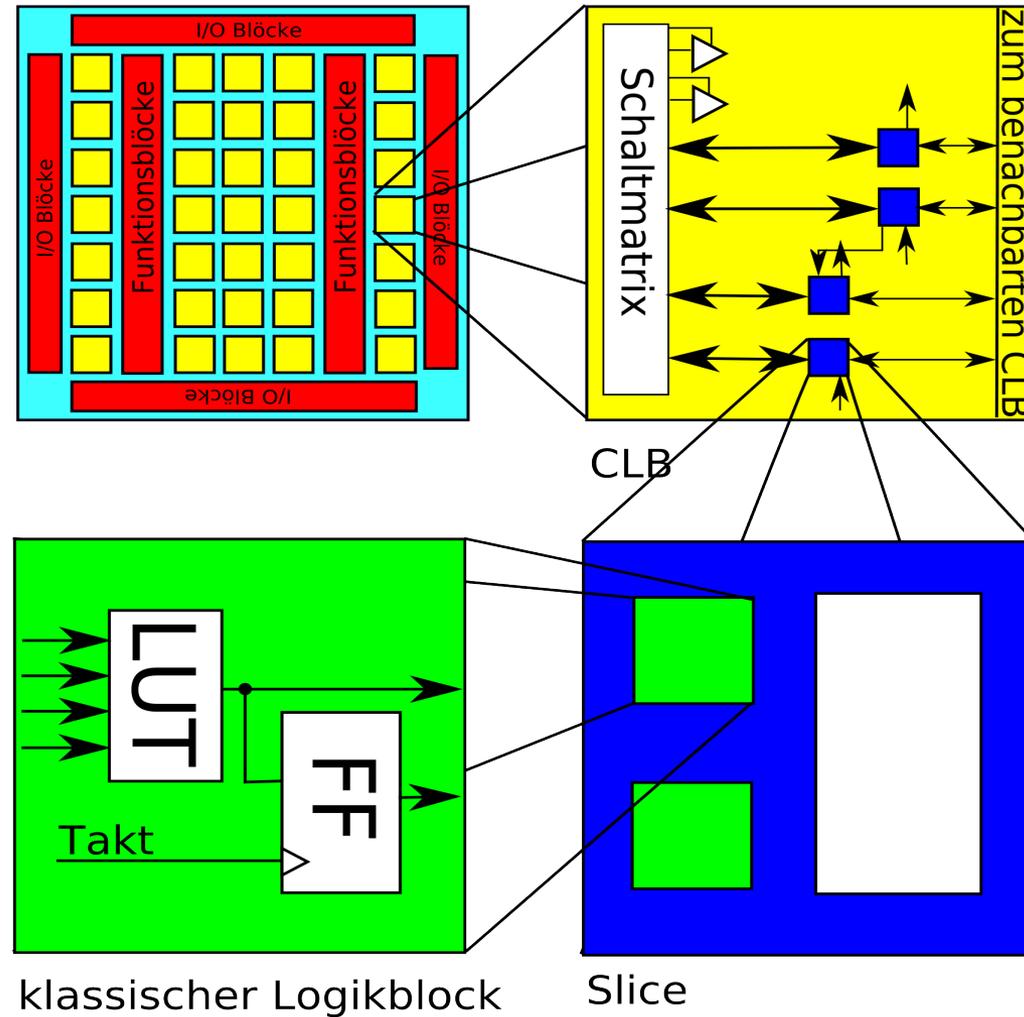
Logik

Speicherzelle

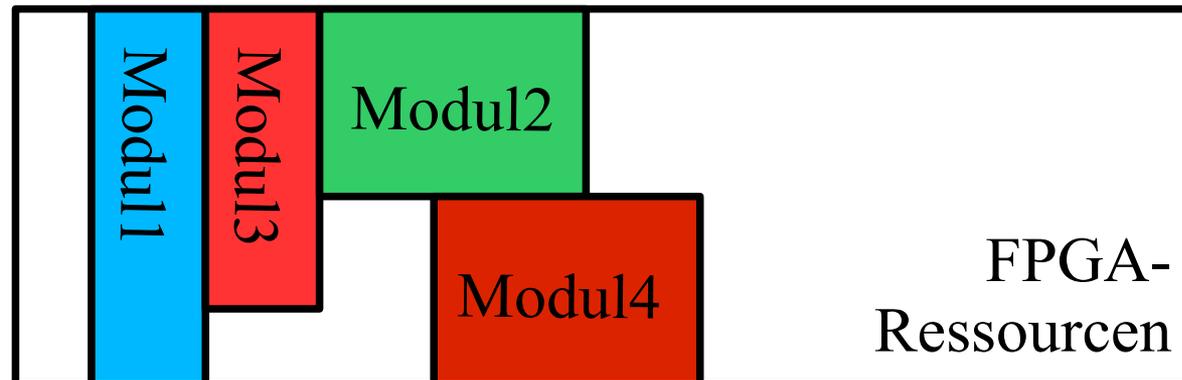
Konfiguration



Architektur von FPGAs



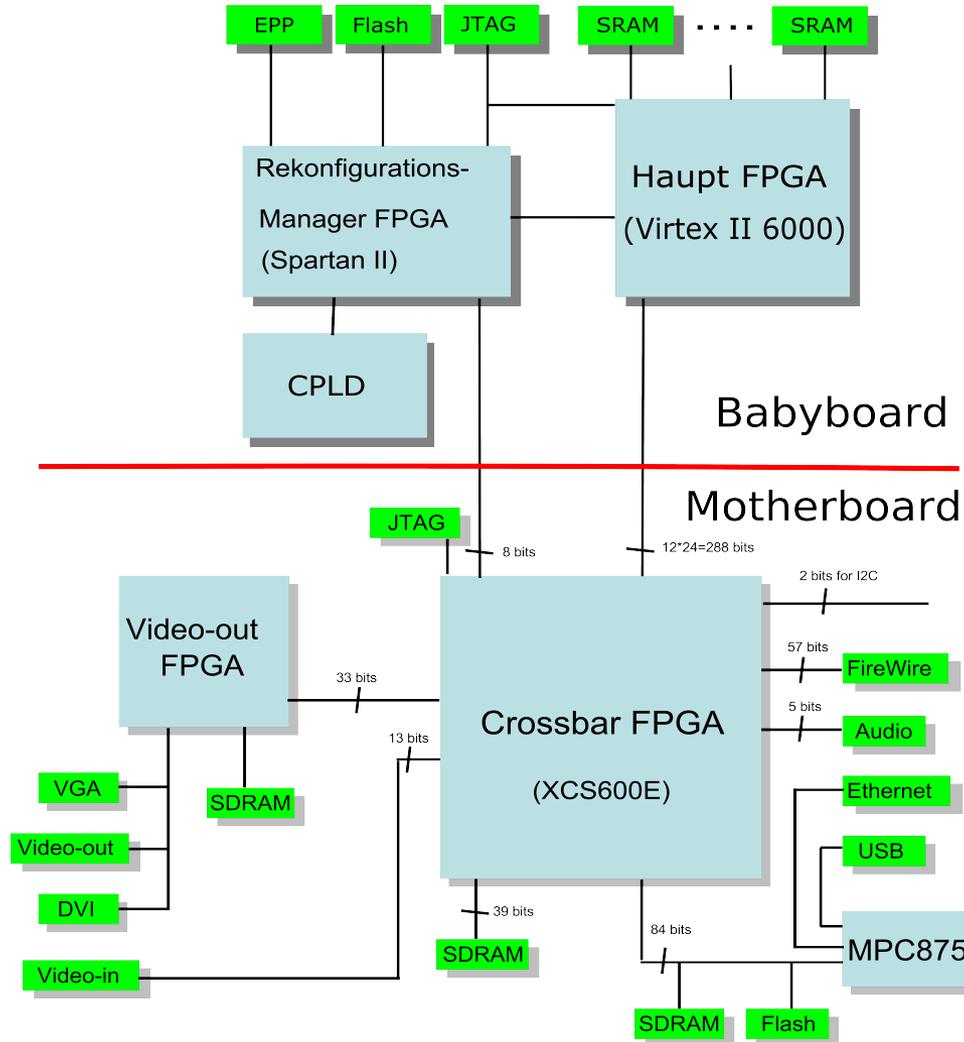
- Auf einem FPGA sollen verschiedene Aufgaben gelöst werden.
- Jede Aufgabe ist einem Modul zugeordnet.
- Jedes Modul benötigt bestimmte Menge an Ressourcen des FPGAs.
- Viele Aufgaben benötigen zur Abarbeitung nur endlich viel Zeit.
- -> **effizientes Nutzen der FPGA Ressourcen** durch laden / entfernen von Modulen zur Laufzeit



- Vorteile von FPGAs
 - weites Anwendungsfeld
 - kurze Implementierungszeiten
 - geringes Designrisiko – keine Silicon Bugs
 - beliebige Systemarchitekturen sollen sich abbilden lassen
 - hohes Maß an Parallelität möglich
- + Vorteile der dynamischen partiellen Rekonfigurierbarkeit:
 - Time-Sharing für Hardware-Ressourcen
 - FPGA Ressourcen können besser ausgenutzt werden
 - Schaltung kann zur Laufzeit an Anforderungen angepasst werden
 - Flexibilität bleibt zur Laufzeit erhalten

- **Anforderungen an die Plattform**
 - dynamisch partiell rekonfigurierbare Plattform
 - Flexibilität bei:
 - Berechnungen
 - Verbindung zur Peripherie
 - Platzierung von Modulen
 - Kommunikation zwischen den Modulen
- **Ableitbare Zielstellungen**
 - Aufzeigen von Realisierungsmöglichkeiten für derartige Systeme
 - Validieren von Forschungsprojekte durch Anwendungen

- **feste I/O Pins**
 - ✗ beliebiges Platzieren von Modulen
 - ✗ Signal Routing durch andere Module hindurch
- **beliebiges Platzieren von Modulen zur Laufzeit**
 - nur ein Bitfile für jede Platzierungsmöglichkeit eines Moduls
- **Kommunikation**
 - Unterbrechungsfrei während Rekonfiguration
 - zwischen Modulen
 - mit Peripherie
- **Speicherzugriff**
 - schnell
 - skalierbar
- **Werkzeugunterstützung**

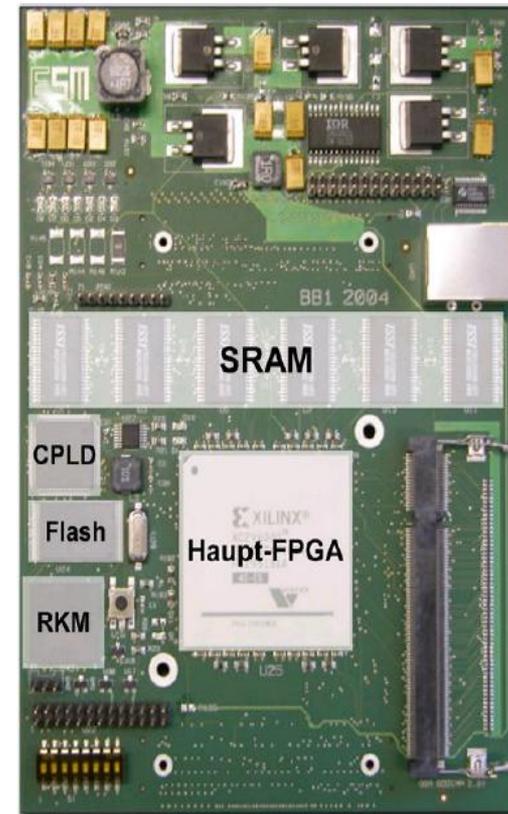


Motherboard & Babyboard



TU Dresden, 02.07.2008

Erlangen Slot Machine
Johannes Görner



[1]

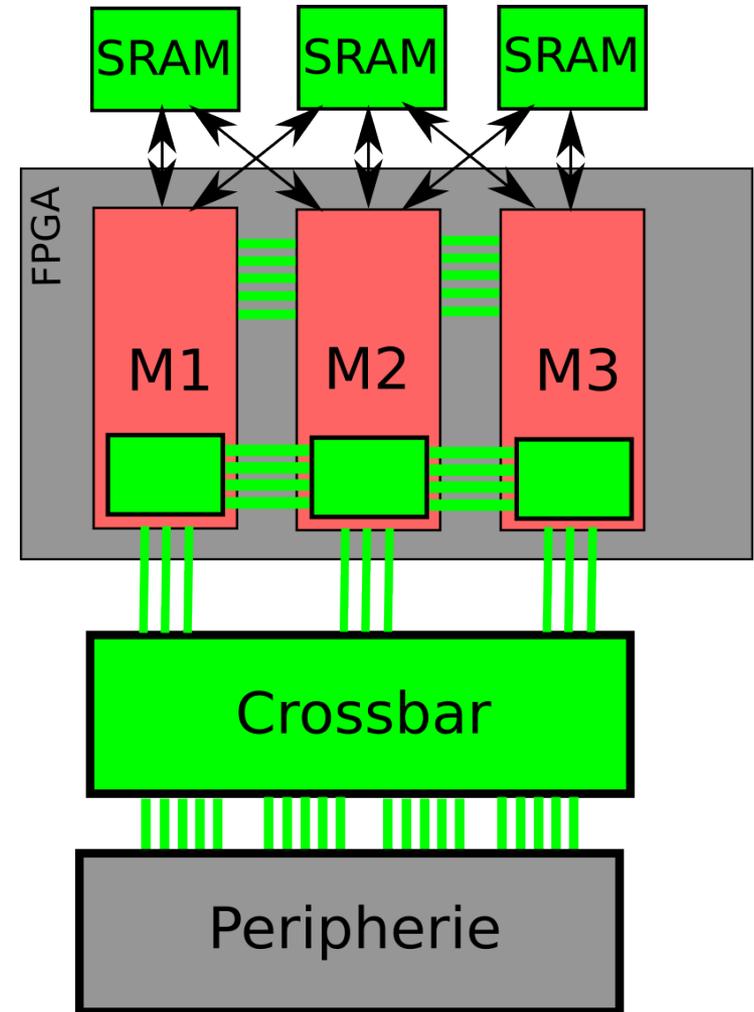
Folie 11 von 27

Intermodul-Kommunikation

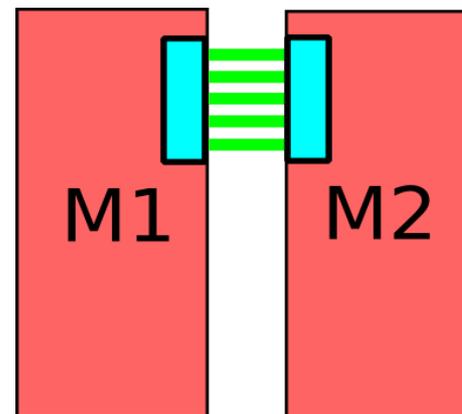
- Bus-Makros
- Gem. Speicher: SRAM, Block-RAM
- Crossbar
- RMB

Peripherie-Kommunikation

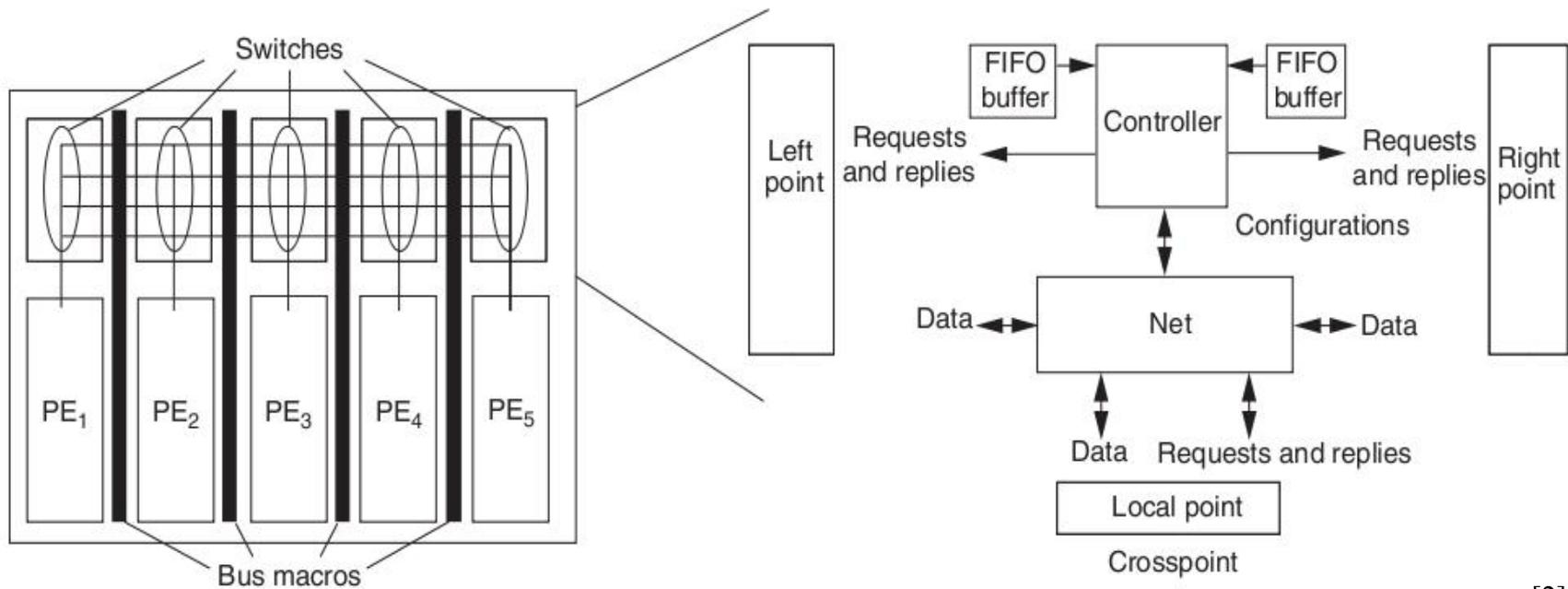
- Crossbar



- stellt feste Verbindungsstruktur zwischen benachbarten Modulen bereit
- bei allen Modulen exakt identisch und an der gleichen Stelle
- wird von FPGA-Hersteller Geräte-spezifisch angeboten
- Fixe Abbildung auf FPGA-Ressourcen

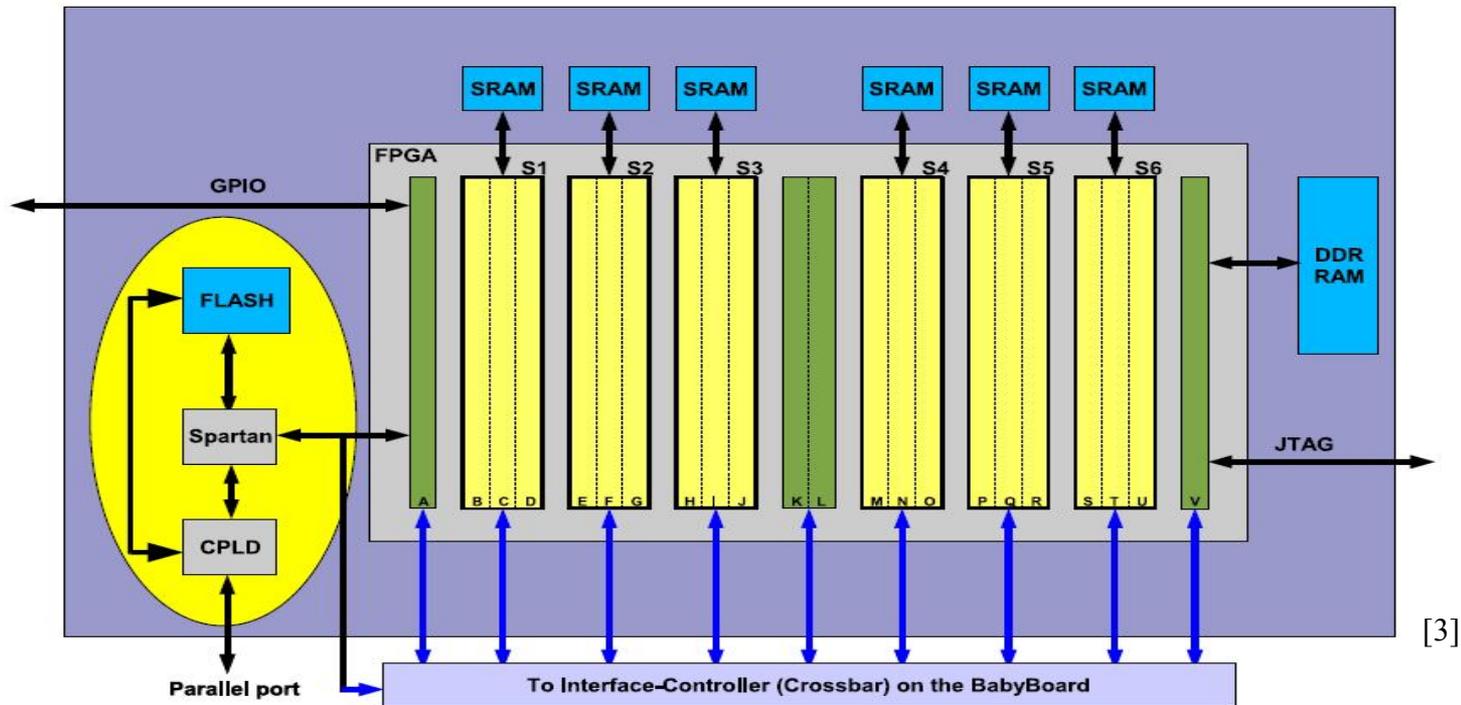


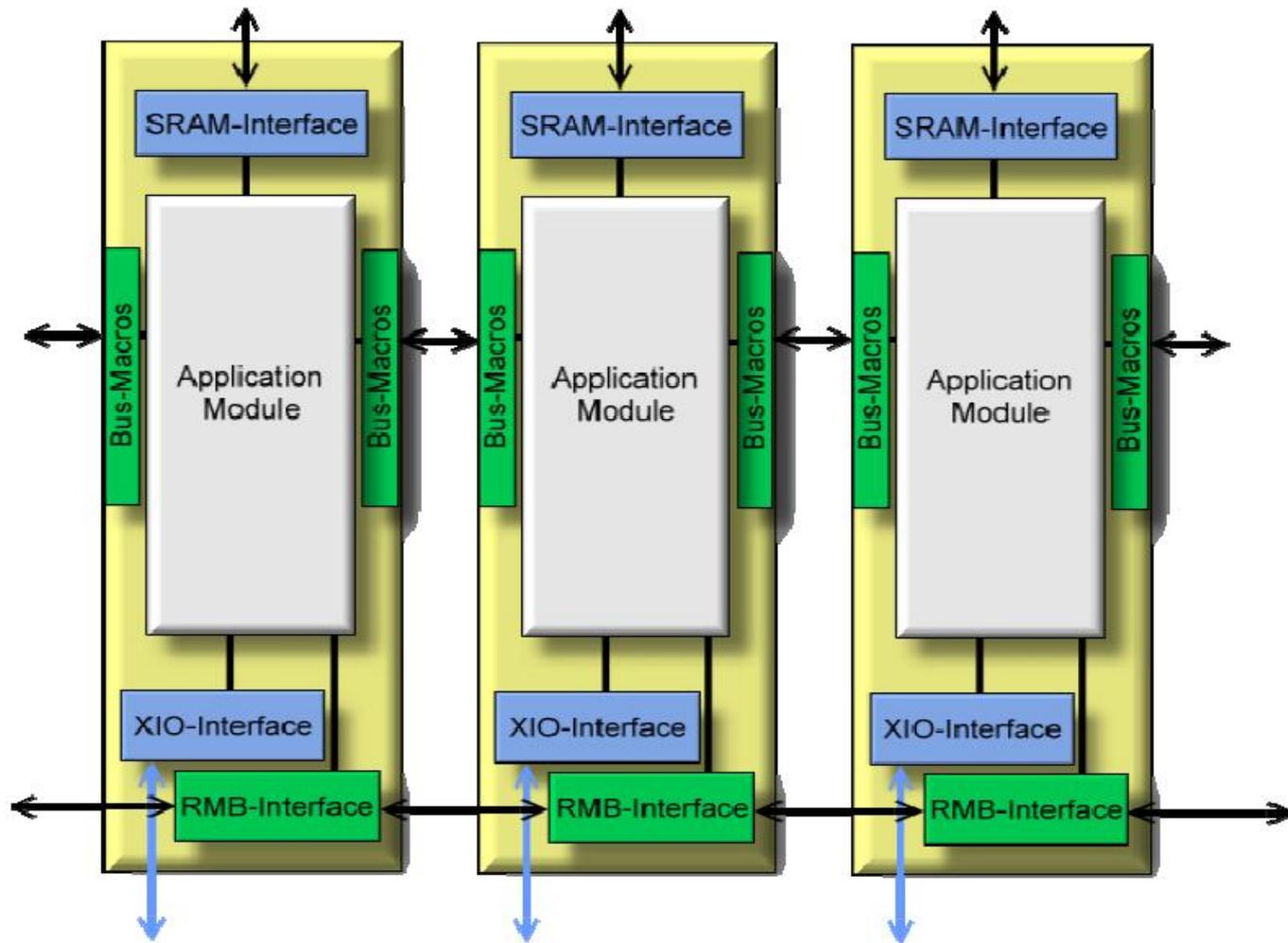
- Reconfigurable **M**ultiple **B**us **o**n **C**hip
- Circuit Routing
- Sender sendet Befehle zum auf-/abbauen von Verbindungen



[2]

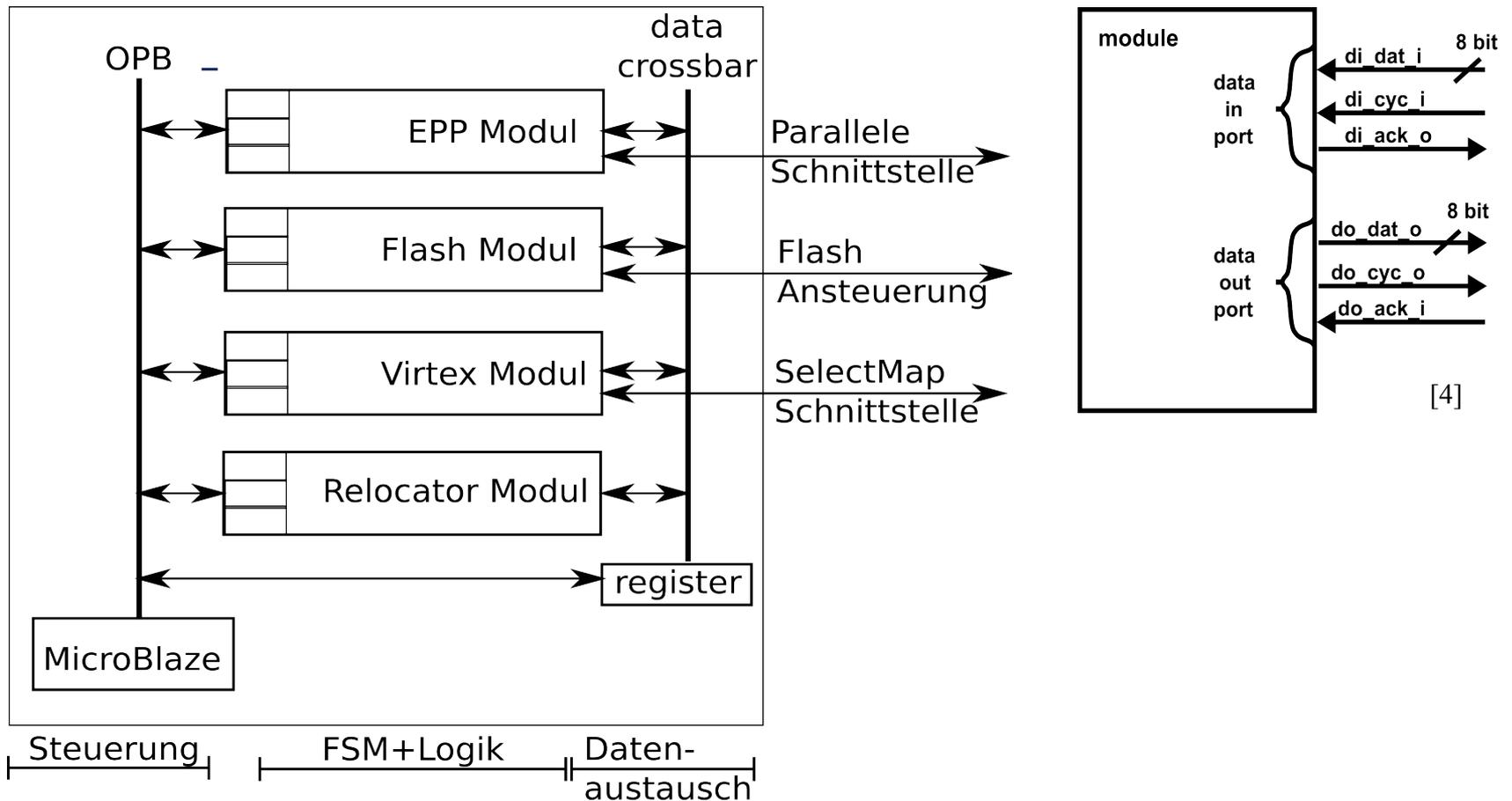
- 1-D spaltenweises dynamisches rekonfigurieren möglich (ganzer CLB)
- homogene Ressourcenstruktur
- „glitchless“ Rekonfiguration





- Zentrales Element: **Rekonfigurationsmanager**
 - Anweisungen/Daten von PowerPC
 - verwalten der freien Slots
 - konfigurieren der Crossbar
 - konfigurieren der Haupt-FPGA / laden der Module
 - Bitstream-Verwaltung
 - Manipulation des Bitstreams zur Laufzeit
 - partielles Bitfile muss nur einmal erstellt werden
 - speicherplatzeffizient

RCM - Rekonfigurationsmanager



- **FPGAs**
 - Schnittstellen wie ICAP ermöglichen dynamische sich selbst rekonfigurierende Architekturen
 - 2-Dimensionale dynamische Rekonfiguration ?
 - 2-Dimensionales dynamisches Routen von Informationsflüssen ?
- **Verfahren**
 - Nutzung der Ressourcen im Zeitscheibenverfahren nach Bedarf
 - Module als Coprozessoren für SW-Task
- **Entwurfsmethoden**
 - bessere Werkzeugunterstützung durch Hersteller
 - Sprachunterstützung für dynamisches Laden von Modulen

- mit heutigen FPGA-Plattformen ist dynamisch partielle Rekonfiguration möglich
- Herausforderungen:
 - Kommunikation
 - Platzierung der Module
 - Design Flow
- zahlreiche Veröffentlichung zum Thema verfügbar

Literatur:

- **ESM-Wiki**
<http://www12.informatik.uni-erlangen.de/esmwiki/>
- **DFG-Schwerpunktprogramm 1148**
<http://www12.informatik.uni-erlangen.de/spprr/abstract.php>
- **The Erlangen slot machine: increasing flexibility in FPGA-based reconfigurable platforms**
Bobda, C.; Majer, A.; Ahmadiania, A.; Haller, T.; Linarth, A.; Teich, J., 2005
- **The Erlangen Slot Machine: a highly flexible FPGA-based reconfigurable platform**
Christophe Bobda, C.; Majer, M.; Ahmadiania, A.; Haller, T.; Linarth, A.; Teich, J., 2005
- **An FPGA-Based Dynamically Reconfigurable Platform: From Concept to Realization**
Mateusz, M., 2006
- **Minimizing Communication Cost for Reconfigurable Slot Modules**
Fekete, S.P.; van der Veen, J.C.; Majer, M; Teich, J, 2006
- **RMB-a reconfigurable multiple bus network**
ElGindy, H.; Schroder, H.; Spray, A.; Somani, A.K.; Schmeck, H., 1996
- **Dynamic interconnection of reconfigurable modules on reconfigurable devices**
Bobda, C.; Ahmadiania, A., 2005
- **XILINX II Data Sheet**
http://www.xilinx.com/support/documentation/data_sheets/ds031.pdf

- **Abbildungen:**
- [1] ESM_Tutorial_1_Intro.pdf
- [2] **Dynamic interconnection of reconfigurable modules on reconfigurable devices**
Bobda, C.; Ahmadiania, A.;, 2005
- [3] **The Erlangen Slot Machine: a highly flexible FPGA-based reconfigurable platform**
Christophe Bobda, C.; Majer, M.; Ahmadiania, A.; Haller, T.; Linarth, A.; Teich, J.;, 2005
- [4] http://www12.informatik.uni-erlangen.de/esmwiki/images/8/89/ESM_RCM_data_crossbar.svg
- [5] http://www12.informatik.uni-erlangen.de/esmwiki/images/f/ff/SlotComposer_block_diagram1.png
- [6] http://www12.informatik.uni-erlangen.de/esmwiki/images/7/78/Partial_flow.png

Vielen Dank für
Ihre
Aufmerksamkeit!