



Fakultät für Elektrotechnik
Professur für Technische Informatik
Helmut Schmidt Universität
Universität der Bundeswehr, Hamburg
Dipl. Inf. Dominik Meyer

Multicore Reconfiguration Platform

Architektur und Implementierung

Inhalt

- 1 Einleitung
- 2 Multicore Reconfiguration Platform
- 3 Implementierung
- 4 Zusammenfassung

Einleitung

Motivation

- Mehrkern Prozessoren State of the Art
- Reconfigurierbare Prozessoren existieren
 - Convey HC1
 - Proteus
 - Arbeit von Daniel Hallmannseder
 - ...
- Verbindung: Reconfigurierbare Multikern Systeme

Einleitung

Eigenschaften Rekonfigurierbarer Mehrkern-Systeme

notwendig

- mehrere Kerne
- zur Laufzeit rekonfigurierbar
- Verbindungsnetzwerk konfigurierbar

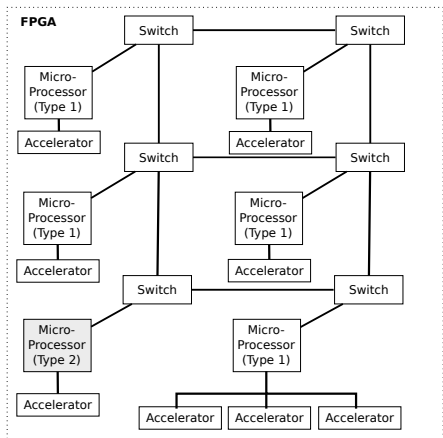
optional

- mehrere heterogene Kerne
- einzelne Kerne rekonfigurierbar
- keine initiale Beschränkung der Anzahl von Kernen

Einleitung

Beispiel: RampSoc

- Runtime Adaptive Multi-Processor System-on-Chip
- feste Anzahl von Kernen
- proprietäre Kerne (Microblaze)
- Accelerator Komponenten
- konfigurierbares Netzwerk



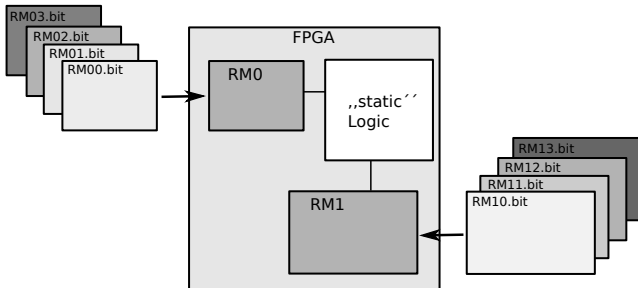
Einleitung

Einschränkungen RampSoc

- feste Anzahl von Kernen
- feste Größe von Kernen
- Kerne nicht reconfigurierbar
- immer maximal große Kerne \Rightarrow schlechte Platzausnutzung
- Beschleunigung nur über Accelerator Komponenten

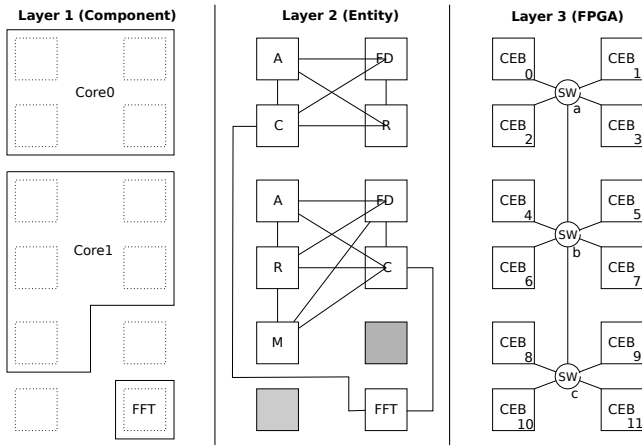
Einleitung

System Voraussetzung - Partial Reconfiguration



Multicore Reconfiguration Platform

Konzept



Multicore Reconfiguration Platform

Configurable Entity Block

- Reconfiguration Module des PR-Konzeptes
- alle die gleiche Größe \Rightarrow maximale Dichte auf dem FPGA
- enthält ein Berechnungselement (Teil eines Prozessorkerns, FSM, etc.)
- festes Signal Interface
- mehrere über einen Switch verbunden

Multicore Reconfiguration Platform

Switch

- Reconfiguration Module des PR-Konzeptes
- programmierbare Routing Tabellen
- Programmiernetzwerk I^2C/SPI
- Implementierung von NOC, PTP, Ring, ...

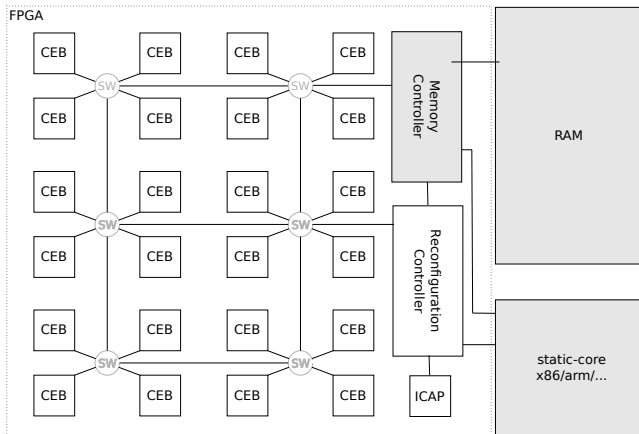
Multicore Reconfiguration Platform

Vorteile

- keine festen Kerngrößen
- keine feste Anzahl von Kernen
- Rekonfiguration innerhalb eines Kernes
- dynamisch anpassbares Verbindungsnetz

Multicore Reconfiguration Platform

Architektur



Implementierung

Hardware

- Convey HC1
- Kombination aus X86 und FPGA
- 4 x Virtex5 FPGAs
- genaueres im nächsten Vortrag

Implementierung

Software

Basisstruktur

- Erstellung Reconfigurable Modules
- CEB Switch Vernetzung
- Platzierung aller Komponenten auf dem FPGA
- Routing aller Signale zur Vernetzung
- Perl Script

Implementierung

Software

CEB Entities

- Erstellung in VHDL
- Hinzufügen zum MRP durch Perl Script

Switch Entities

- Erstellung in VHDL
- Hinzufügen zum MRP durch Perl Script

Implementierung

Synthese

- jede Komponente muß ein mal synthetisiert werden
- Jede CEB/Switch Komponente muß für jede CEB Position plaziert und geroutet werden
- für y CEB Komponenten und z CEBs $\Rightarrow y \times z$ Synthese + Plazierungsschritte
- Verwendung: paralleles Makefile für PR

Implementierung

paralleles Makefile

Abarbeitung

- 1 statische Komponente synthetisieren
- 2 parallel: alle CEB/Switch Komponenten synthetisieren
- 3 parallel: alle CEB/Switch Komponenten für alle RM mappen
- 4 parallel: alle CEB/Switch Komponenten für alle RM plazieren
- 5 parallel: alle CEB/Switch Komponenten für alle RM routen

Ergebnis

mindestens doppelt so schnell wie standard Xilinx Design Flow !

Zusammenfassung

- MRP - plattformunabhängiges reconfiguration Framework
- Benutzung im Baukastenprinzip
- flexible Aufteilung des FPGA \Rightarrow gute Platzausnutzung
- nicht nur für Prozessoren einsetzbar