Selbstrekonfiguration des Funkprototypingsystems HaLo

Marcel Köhler

marcel.koehler@inf.tu-dresden.de

Technische Universität Dresden
Institut für Technische Informatik

Gliederung

- Aufgabenstellung
- Motivation
- Problemanalyse
- Vorstellung der HaLo Plattform
- Rekonfigurationsflow des Gesamtsystems
- Lösungsansatz
  - Datenformat
  - Prototypingaufbau
- Implementierung
  - State Machine
  - Prozessorkern
- Zusammenfassung

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Schwerpunkte der Aufgabenstellung

- Einarbeitung/ Literaturstudium zu:
  Funkprototypingsystem HaLo
  JTAG-basierte Rekonfiguration von HW-Bausteinen

- Untersuchung und Evaluierung von Möglichkeiten für:
  Steuerung zur Rekonfiguration des FPGAs

- Implementation von Lösungsansätzen auf dem HaLo-System

- Beispiele zur Validation der entwickelten Ansätze
- Effizienzbetrachtungen der Implementationsvarianten

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Motivation

- Erwerb neuer Hardware/ Software

- Kunde erwartet möglichst guten Support

  → hier speziell Updatefähigkeit
  → Beseitigung von Programmiermängeln (Bugs)
  → Steigerung der Funktionalität
  → auf Kundenwünsche reagieren

- Kundenzufriedenheit als wichtiges Kriterium für Unternehmen

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

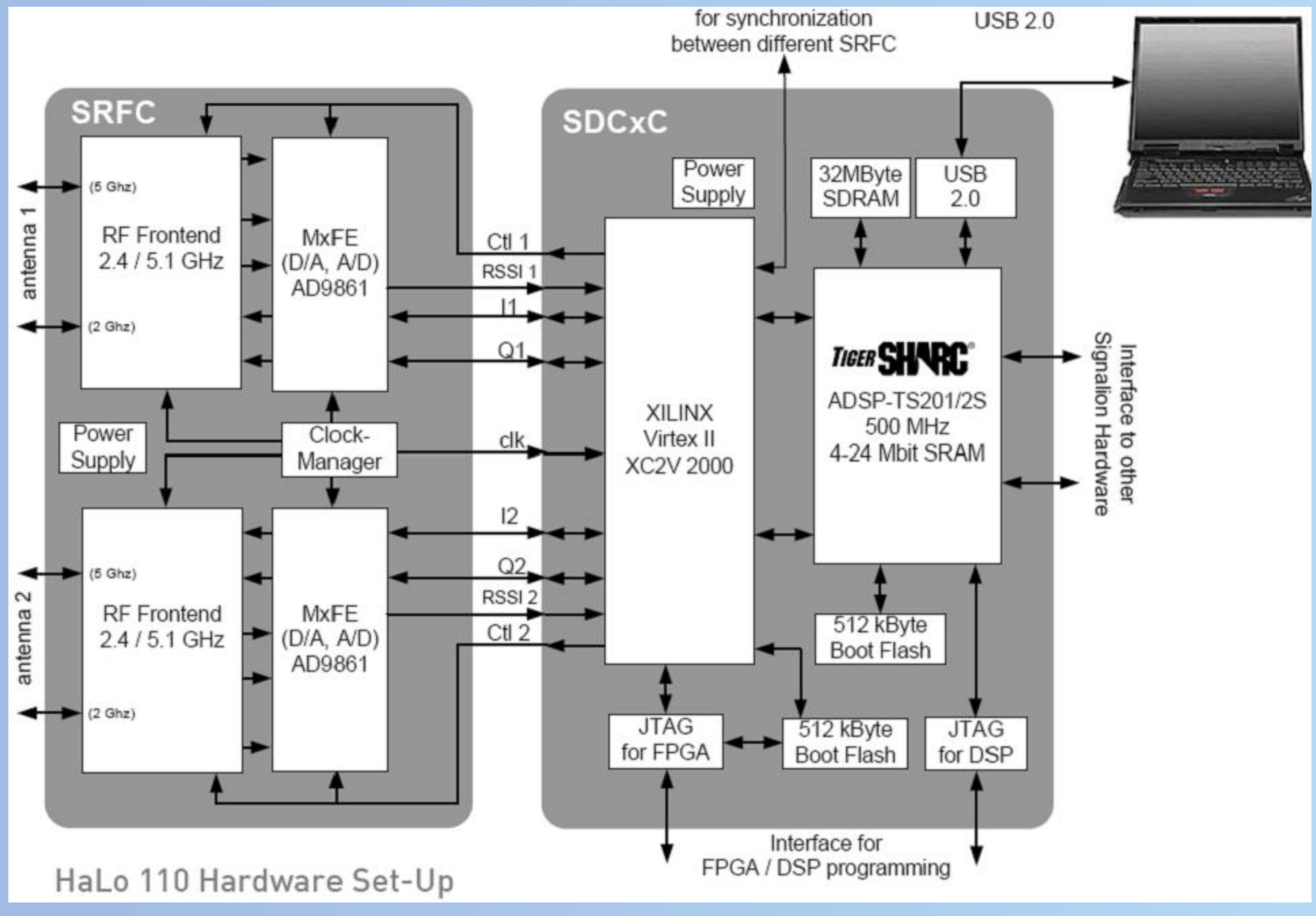
Problemanalyse

- Lösung industrierelevanter Problemstellungen durch Zusammenarbeit von Signalion GmbH und dem Lehrstuhl für VLSI-Entwurfssysteme, Diagnostik und Architektur
- Verbesserung der Updatefähigkeit für bestehendes System
- Möglichkeiten zur Rekonfiguration eines FPGAs für das Funkprototypingsystem HaLo über Rekonfigurationsspeicher
- Problem: möglichst kein zusätzlicher HW-Aufwand, da bereits beim Kunden eingesetzt

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

HaLo Prototypingsystem

- HaLo: „**Ha**rdware in the **Lo**op“



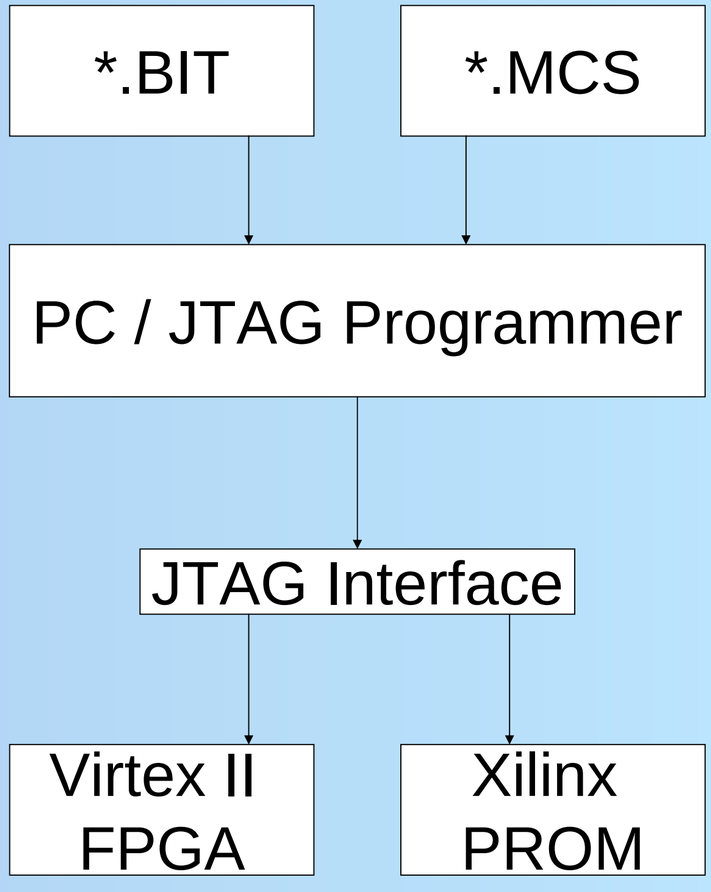
- Test neuer Drahtlosgeräte
- Kanalmodell durch realen Funkkanal ersetzt
- Anbindung an Matlab über USB-Interface

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Rekonfigurationsflow (IST)

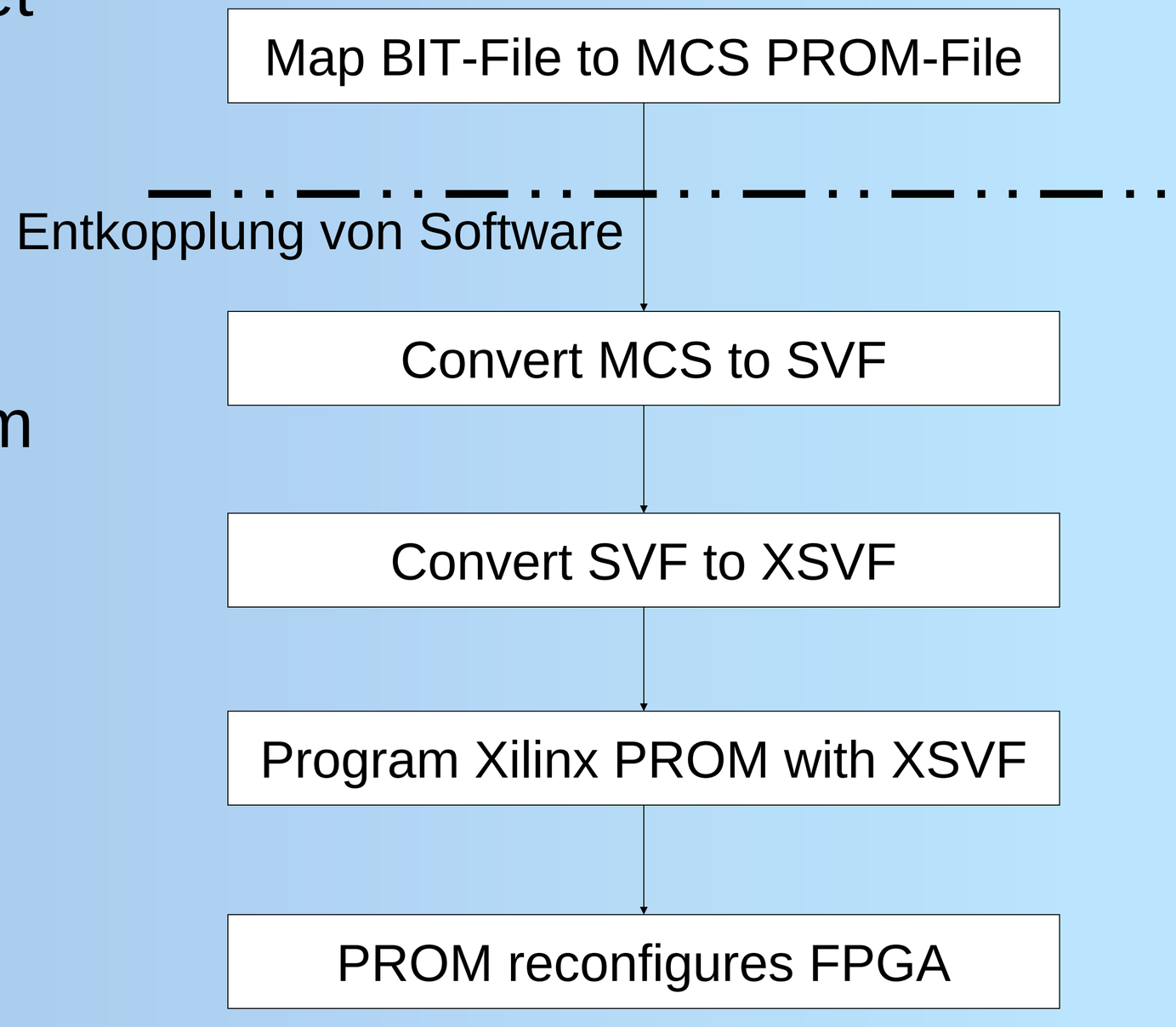
- DSP über USB2.0 rekonfiguriert
- für FPGA derzeit noch keine Updatefähigkeit via USB2.0
- Designflow (jetzt)
  - für Testzwecke BIT-File
  - für dauerhafte Speicherung MCS-File für PROM
  - Download erfolgt mit Xilinx iMPACT oderJTAG Programmer
- wünschenswert ohne Software Rekonfiguration durchzuführen
- Wegfall Vor-Ort-Service beim Kunden



Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Rekonfigurationflow (SOLL)

- SVF Format beinhaltet Daten und Boundary Scan Befehle
- XSVF Format kompakte binäre Form des SVF Inhalts
- Einsatz des FPGAs zur PROM Programmierung



Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Datenformate

- SVF Format zur Beschreibung der IEEE.1149.1 Operationen von Texas Instrument entwickelt
- Industriestandard zur Programmierung über JTAG
  → unabhängig von eingesetztem Test-Equipment und verwendeter Programmiersoftware
- ASCII codiert - für eingebettete Anwendungen zu groß
- deshalb angepasstes Format: XSVF Dateiformat
  - Vorteile gegenüber SVF Format:
    - Bessere Datenkompression (ca.2:1), kleinere Files
    - durch Modifikation des Formats evtl. weiterer Kompressionsgewinn (muss noch untersucht werden)

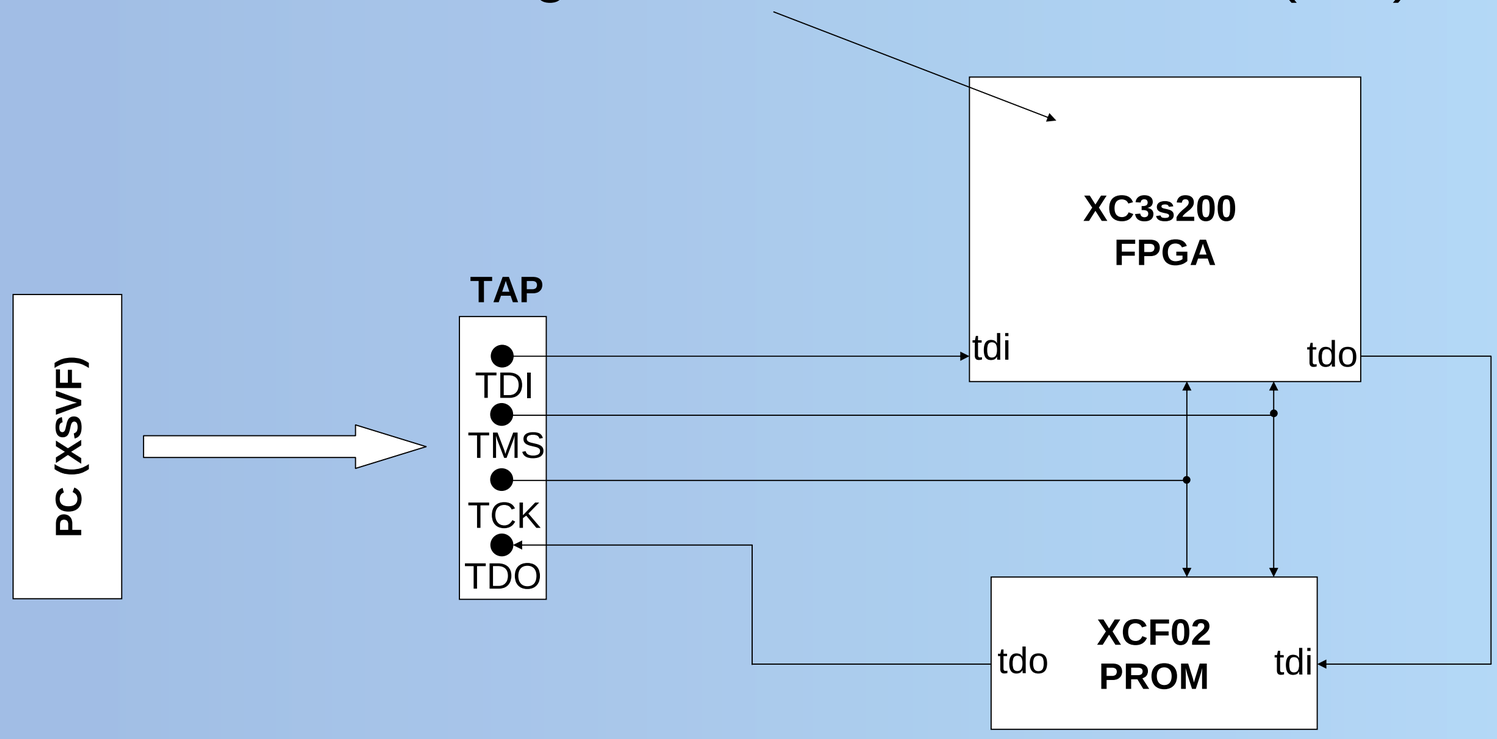
Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Versuch

- Versuch mit Praktikumsboard (Spartan FPGA)
- 2 Testentwürfe erstellt

I.Schritt: Konfiguration 1 auf FPGA laden (BIT)



II.Schritt: Programmierung PROM mit Konfiguration 2 (XSVF)
III.Schritt: neue Konfiguration aus PROM nach Reset/Neustart

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Folgerung aus Versuch

- Erkenntnis: während der Serial PROM via JTAG neu programmiert wird, arbeitet FPGA in Normalmodus

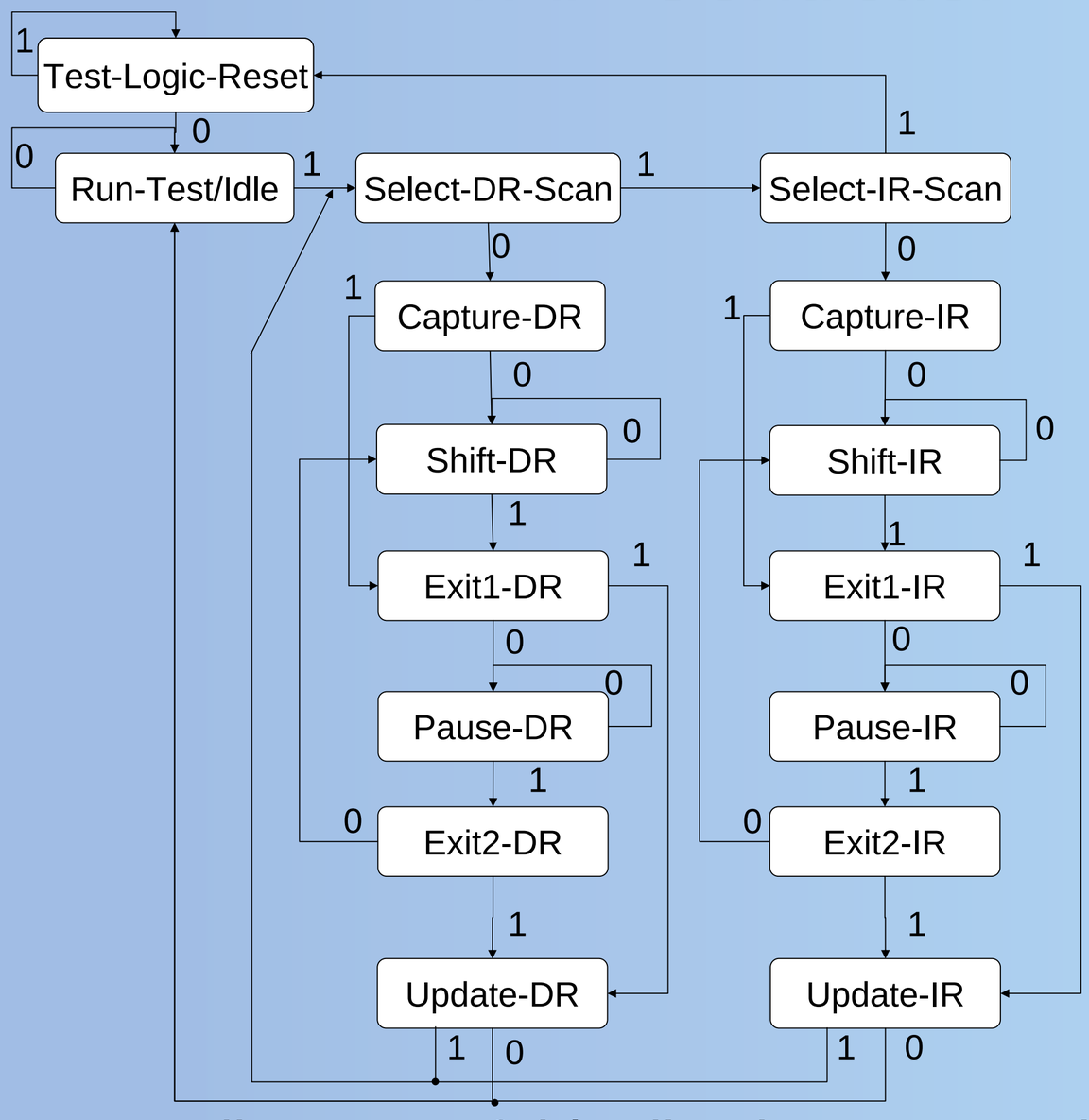
  - Bedeutung: während Rekonfiguration(PROM) kann FPGA eigenständige Aufgaben übernehmen
  - Gedanke die Steuerung (PC) in JTAG-Kette zu verlagern

1. FPGA erhält Streaming (XSVF)
2. Implementierung eines Entwurfs, der Ansteuersignale TDI,TDO,TMS, TCK generiert
3. FPGA stimuliert über Rückkopplung (Kabel) TAP
4. Daten über TDI in PROM

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Zusammenhang TAP controller - XSVF

TAP Controller



XSVF Befehle

XREPEAT 0x20

XSTATE 0x00

XSTATE 0x01

XRUNTEST 0x00000000

XSIR 0x0E 0x3ffe

XSDRSIZE 0x00000021

XTDOMASK 0x000fffffff

XSDRTDO 0x0000000000

0x00f5045093

Es gibt : Befehle die einen Zustandswechsel einleiten

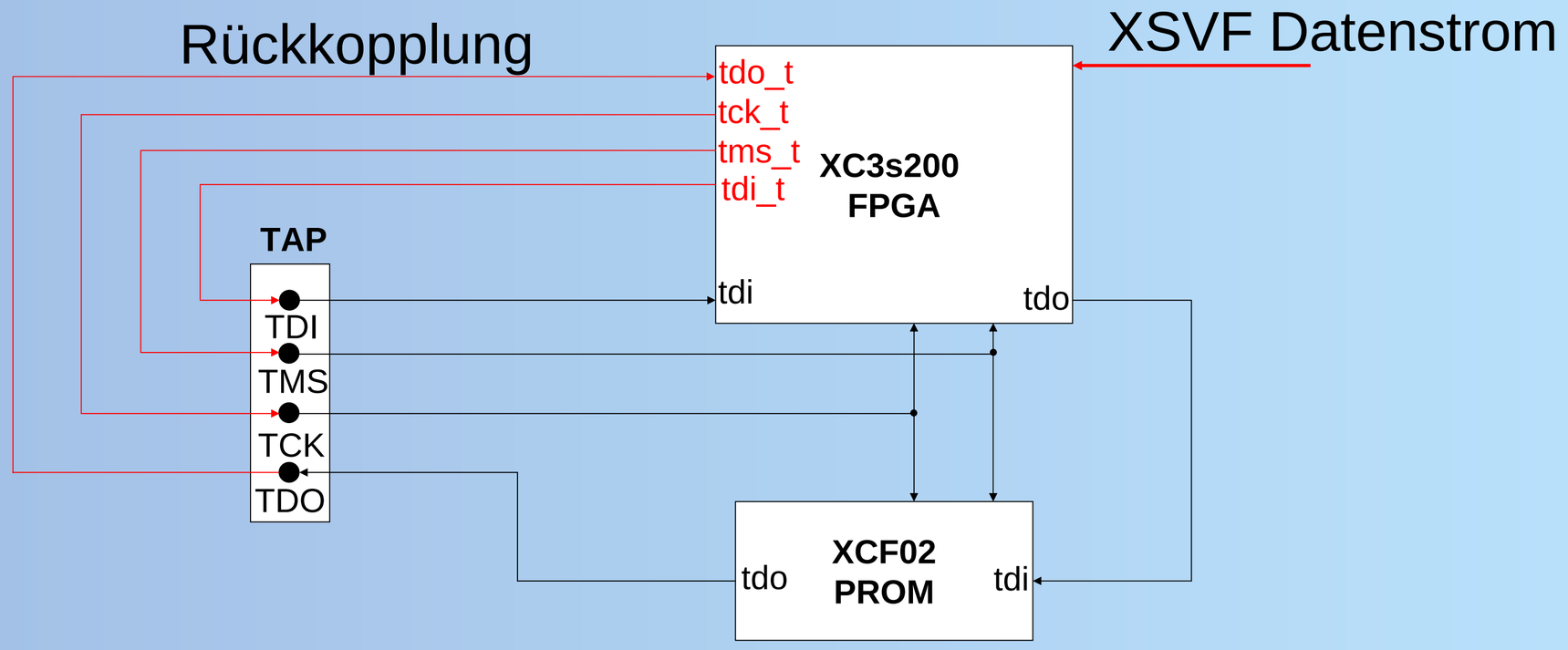
Befehle für Statusinformationen

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Prototypingumgebung

- Umgebung zum Prototypen



- JTAG standardisiertes Interface
- zunächst Anstrebung einer „Stand-Alone“-Lösung
  → universelle Integration in andere Systeme
  → adaptierbar auf HaLo-Systemkomponenten

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Implementierung(1)

- Realisierung einer State Machine
  - Analyse XSVF –File ergab Notwendigkeit folgender HW-Elemente:
    - Zustandsautomat (8 von 21 XSVF-Befehlen)
    - zusätzlich Menge von Auffangregistern zur Speicherung von Informationsdaten für Folgebefehle:

      Wiederholversuche 8-Bit (XREPEAT)

      Wartezyklen 32-Bit (XRUNTEST)

      Argumentlänge ‚length‘ 32-Bit (XSDRSIZE)

      Ready-Signalisierung 1-Bit (XCOMPLETE)

      ‚length‘-Bit Auffangregister für ‚tdoCaptured‘ und ‚tdoExpected‘ (XSDRTDO / XSIR)

      ‚length‘-Bit Auffangregister für ‚tdoMask‘ (XTDOMASK)

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

Implementierung(2)

- Microprozessorcore als Steuerung
  - Design auf FPGA
    - umfangreicher als FSM-Design
    - flexibler in Anpassung an veränderte Algorithmen

- alternative Lösungsansätze
  - Einsatz der dynamischen Rekonfiguration
  - Möglichkeit andere Aufgaben mit µC zu erledigen
  - Steuerung von System- und Testfunktionen

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Zusammenfassung

- erledigte Aufgaben:
  - Literaturstudium
  - Verständnis zu den Formaten
  - erster Teil der schriftlichen Arbeit fertiggestellt
  - Überlegung zu einer geeigneten Protoyping–Umgebung
  - Aufbau des Prototypingsystems
- TODO:
  - Implementation der Lösungsansätze (aktuell in Arbeit)
  - Effizienzbetrachtungen
  - Schnittstelle DSP – FPGA
  - Dokumentation der Ergebnisse

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de

SIGNALION

Vielen Dank für Ihre Aufmerksamkeit!

Marcel Köhler
Technische Universität Dresden
Institut für Technische Informatik
marcel.koehler@inf.tu-dresden.de