

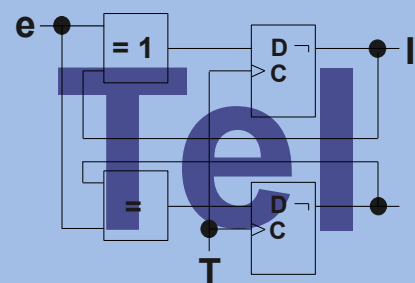
Diplomarbeit

Vortrag zum Diplom

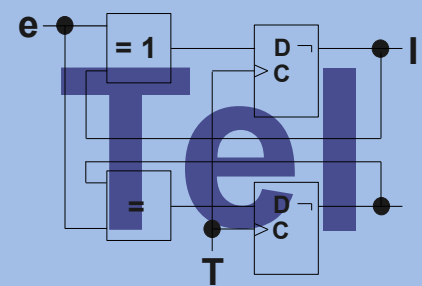
**Mikrocontrollergestützte  
Selbstorganisationsprinzipien  
rekonfigurierbarer  
Rechnersysteme am Beispiel der  
Xilinx FPGA-Architektur**

Falk Niederlein

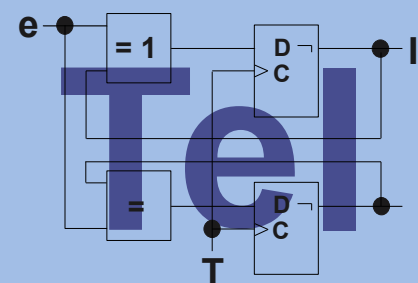
s6838029@mail.inf.tu-dresden.de



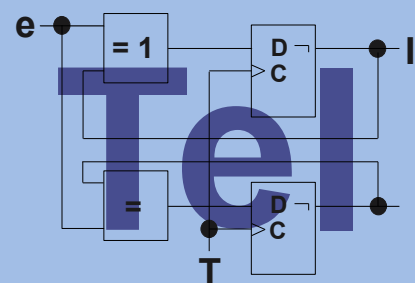
1. Motivation
2. Grundlagen
3. Lösungsansätze
4. Stand der Arbeiten
5. Zusammenfassung



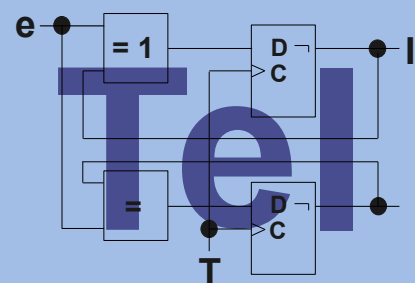
- Implementierung von Methoden, welche die Entwicklung von Systemen ermöglichen, die selbstorganisierende Eigenschaften besitzen
- Ziel der Selbstorganisation ist, ein sich selbst verwaltendes und ausführendes System
- Anwendung der Prinzipien des Organic Computing zur Realisierung der Selbstorganisation
- Entwicklung von Methoden, welche die Möglichkeit der Integration von Selbstorganisationprinzipien in zukünftigen Agentensystemen erlauben
- Einsatzgebiete:
  - Gebäude-Automation
  - Fahrzeugtechnik
  - Robotik
  - Medizintechnik



- Die Prinzipien der Selbstorganisation sollen in dieser Arbeit umgesetzt werden
- Spätere Anwendbarkeit in HW-Agenten, z.B. bestehend aus Mikrocontroller und FPGA  
Verwaltung/Scheduling  
Beschleunigung  
Mikrocontroller:  
FPGA: Hardware-
- Der Mikrocontroller übernimmt die Konfiguration des FPGAs (über JTAG-Schnittstelle; Ersatz des Parallelen Programmierkabels)
- Über die Netzwerk-Schnittstelle des Mikrocontrollers können Konfigurations-Dateien beschafft werden
- FPGA führt die rechenintensive Anwendung aus
- Welche Vorteile?
  - flexibel
  - ausfallsicher
  - leistungsfähig

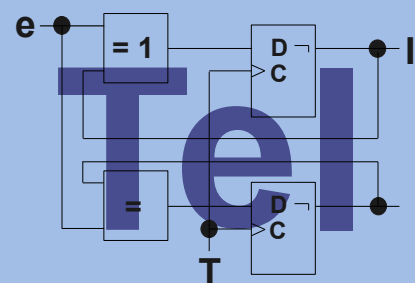


- Agenten sind autonome Systeme, die komplexe Aufgaben selbstständig lösen können
- Man unterscheidet zwischen Software- und Hardware-Agenten  
SW-Agent: Informationsbeschaffung, Buchungssysteme  
HW-Agent: klassischer Roboter, intelligentes eingebettetes System
- Gemeinsame Eigenschaften: autonom, zielorientiert, reaktiv, proaktiv, sozial, mobil und lernend
- Hauptunterschiede entstehen durch unterschiedliche Welten und unterschiedliche Interaktionsmöglichkeiten  
SW-Agent: virtuelle Welt; Interaktion durch Informationen  
HW-Agent: physische Welt; Sensoren  $\leftrightarrow$  Aktoren
- Zusätzlich zu der Hardwareseite, benötigt der HW-Agent, für die Umsetzung der Agenten-Eigenschaften Software



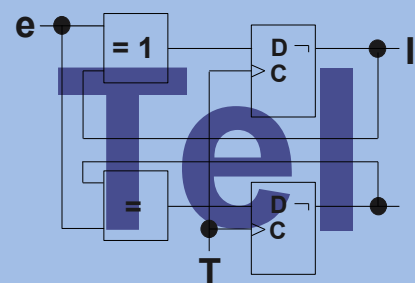


- Ist ein neuer Ansatz für ein (Selbst-) Organisationsprinzip
- Hat die Natur als Vorbild, mit dem Ziel biologische Prozesse zu verstehen und auf technische Systeme zu übertragen
- Lebewesen können sich wechselnden Bedingungen anpassen, ohne dass sich ihre Funktionalität ändert
- „Ein organischer Computer ist als ein selbst-organisierendes System definiert, das sich den jeweiligen Umgebungsbedürfnissen dynamisch anpasst“ – *Gesellschaft für Informatik e.V.*
- Self-x-Eigenschaften: selbst-konfigurierend, selbst-optimierend, selbst-heilend, selbst-erklärend und selbst-schützend
- Die Ideen und Konzepte des OCs gleichen somit in vieler Hinsicht denen der agentenbasierten Systeme → Kombination

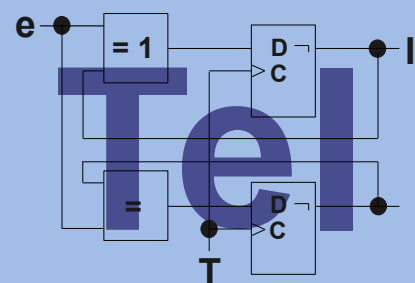


# HW-Unterstützung für Selbstorganisation

- FPGA ist Teil des Agenten (HW-Beschleunigung)
- FPGA stellt die Eigenschaft der Rekonfiguration zur Verfügung
- Die FPGAs der Firma Xilinx basieren auf SRAM-Technologie und sind dadurch rekonfigurierbar
- Statische und dynamische Rekonfiguration einsetzen
- (Re-) Konfiguration kann über unterschiedliche Schnittstellen erfolgen
- Die (Re-) Konfiguration über die JTAG-Schnittstelle ist die Universellste
- Durch Rekonfigurierbarkeit des FPGAs erhält der Agent ein Instrument zur Verwirklichung der Selbstorganisation

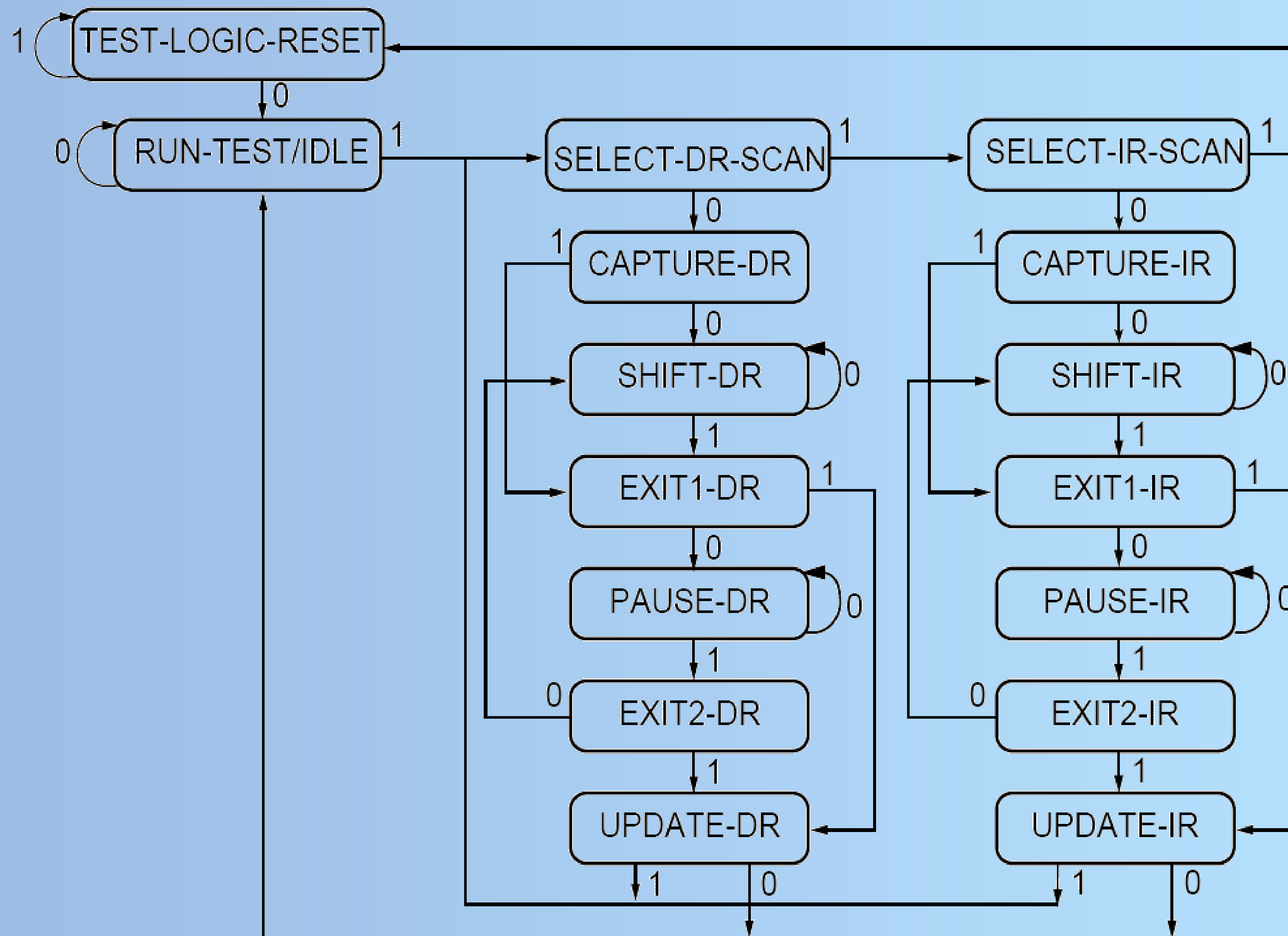


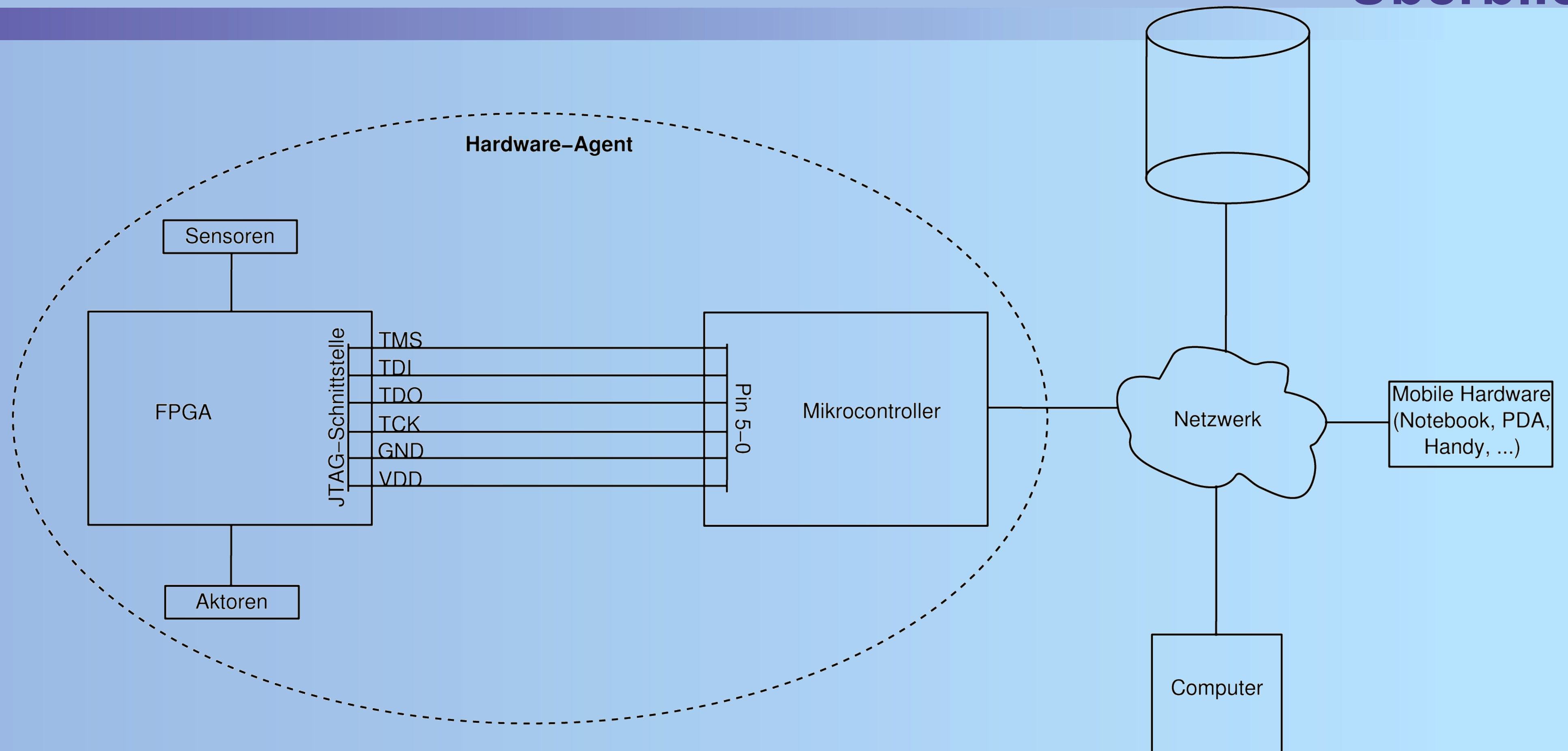
- JTAG (Joint Test Action Group) bzw. Boundary-Scan ist ein bekanntes Testverfahren und heute auch eine Schnittstelle für die bequeme Konfiguration von PLDs
- externe Logik wird verwendet, um die JTAG-Pins anzusteuern
- Die Daten werden seriell ausgetauscht
- Alle JTAG-Operationen werden über die Pins des Test-Access-Port (TAP) kontrolliert, bestehend aus vier obligatorischen Pins (TDI, TDO, TMS, TCK)
- Die vier TAP-Pins kontrollieren, unter Anwendung des TAP-Controllers, wie die Befehle bzw. Daten in die verschiedenen Register eingelesen werden
- Der TAP-Controller ist ein Steuerautomat mit 16 Zuständen





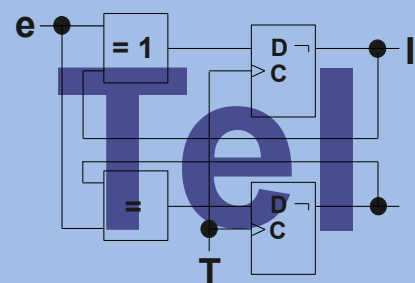
# HW-Unterstützung für Selbstorganisation



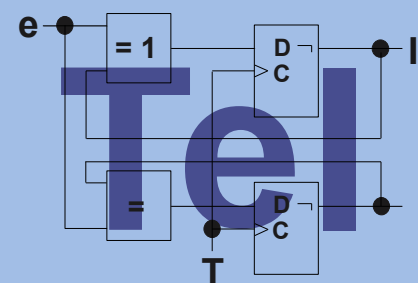


- HW-Agent setzt Selbstorganisationsprinzipien um
- Mikrocontroller verwaltet FPGA
- FPGA ist HW-Unterstützung für verschiedene Aufgaben
- Agent ist netzwerkfähig

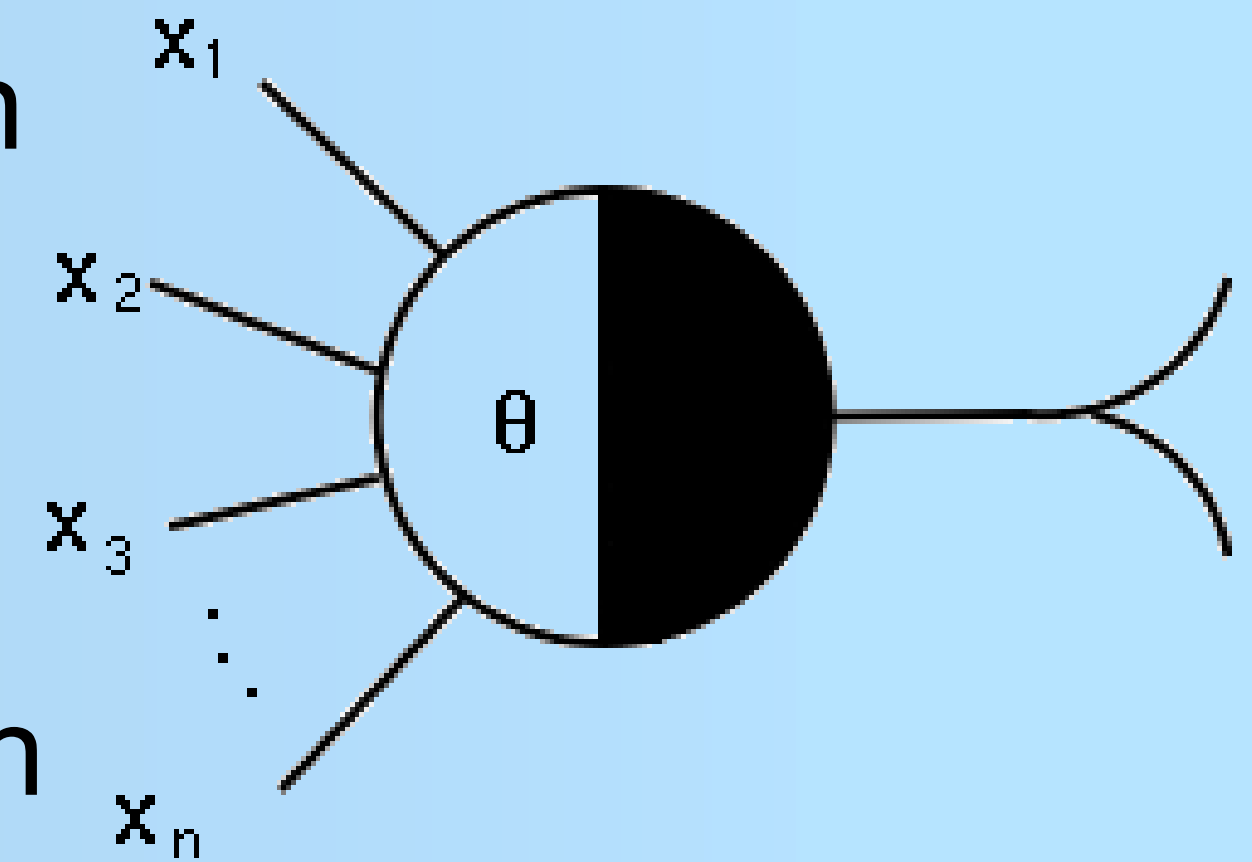
- Die Konfiguration wird durch Befehle gesteuert, die in das Befehlsregister geladen werden
- Die JTAG-Konfiguration bedarf der Konvertierung der Bitstream-Datei in das SVF- und XSVF-Format
- Das SVF-Dateiformat ist ein industrieller Standard und wird benutzt, um die JTAG-Befehle in eine kompakte und portable Form zu bringen
- Das Format überführt hersteller- und bausteinspezifische Abweichungen in allgemeine SVF-Befehle
- Das XSVF-Format ist die kompakte und binäre Form des SVF-Formats, optimiert für Xilinx-FPGAs



- Das Scheduling steuert den Ablauf des Agenten
- Das Scheduling übernimmt der Mikrocontroller
- Das Scheduling umfasst einerseits die Verwaltung der Konfigurationsdateien und andererseits die Rekonfiguration des FPGAs mit der entsprechenden Datei
- Beispielsweise holt sich der MC Konfigurationsdateien aus dem Netzwerk und speichert sie lokal ab, aber entscheidet auch, mit welchem Modul der FPGAs rekonfiguriert werden soll
- Ablauf kann durch einen Graphen visualisiert werden
- Unter Berücksichtigung des Organic Computing soll ein Graph mit neuronalen Eigenschaften entstehen



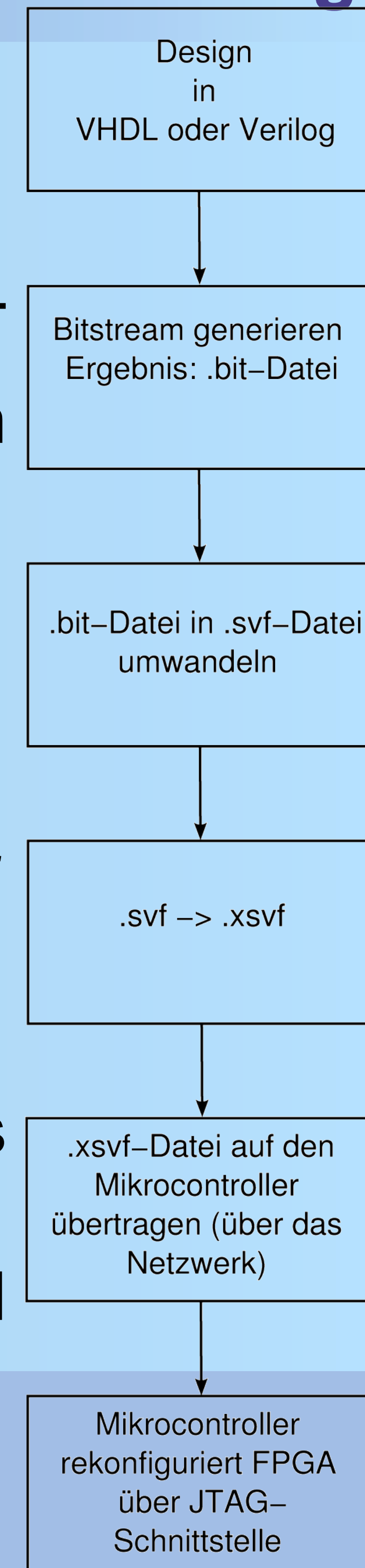
- Eine McCulloch-Pitts-Zelle ist ein von Warren McCulloch und Walter Pitts im Jahr 1943 vorgeschlagenes Neuronenmodell
- Beide wollten ein vereinfachtes Modell realer Vorgänge in neuronalen Strukturen entwerfen
- Das McCulloch-Pitts-Neuronenmodell ist das einfachste Neuronenmodell der Neuroinformatik überhaupt
- Das neuronale Netz aus diesen Zellen kann ausschließlich binäre Signale verwenden
- Eine McCulloch-Pitts-Zelle besitzt  $n$  erregenden Eingangsleitungen und  $m$  hemmenden Eingangsleitungen
- Ein erregendes und ein hemmendes Eingangssignal heben sich auf
- Übriggebliebene Eingangssignale werden aufsummiert und mit der Schwelle  $\Theta$  verglichen; Ist die Summe der Erregungen größer als  $\Theta$  gibt das Neuron 1, ansonsten 0 zurück



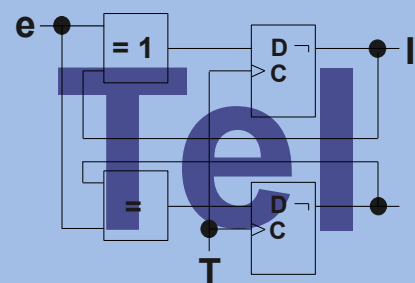


# Ablauf der FPGA-Rekonfiguration

- Erstellen mit iMPACT
- Berücksichtigung von JTAG-Clock-Option
- Bausteinspezifische Datei in allgemeine Form bringen
  - Mit iMPACT oder svf2xsvf
  - Vom ASCII- in binär-Format
- Konfigurationsdatei auf den lokalen Speicher des Mikrocontrollers ablegen
  - Mikrokontroller organisiert Download



• C-Programm

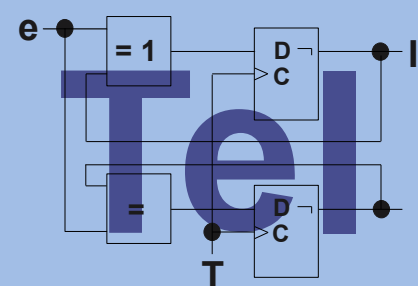
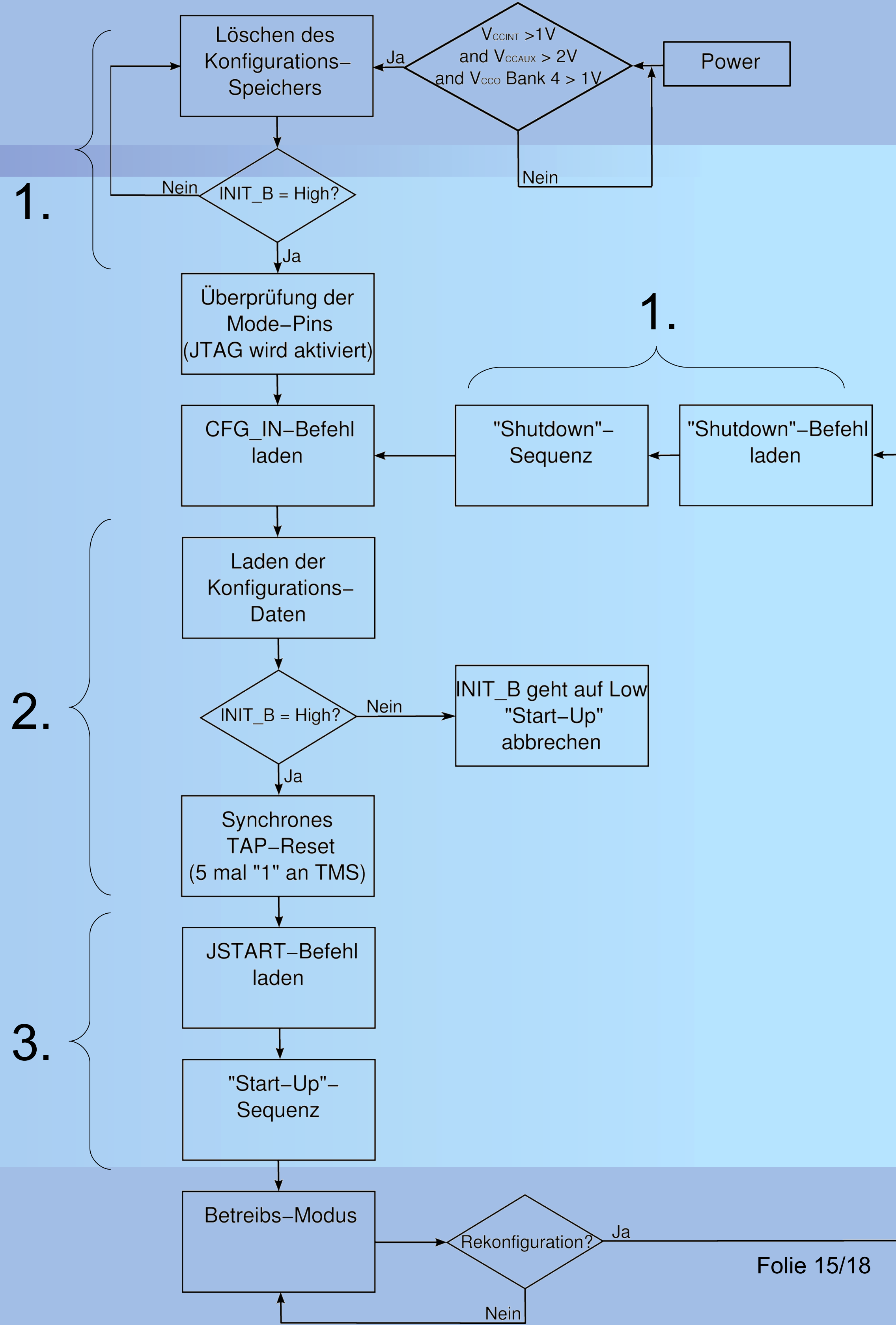


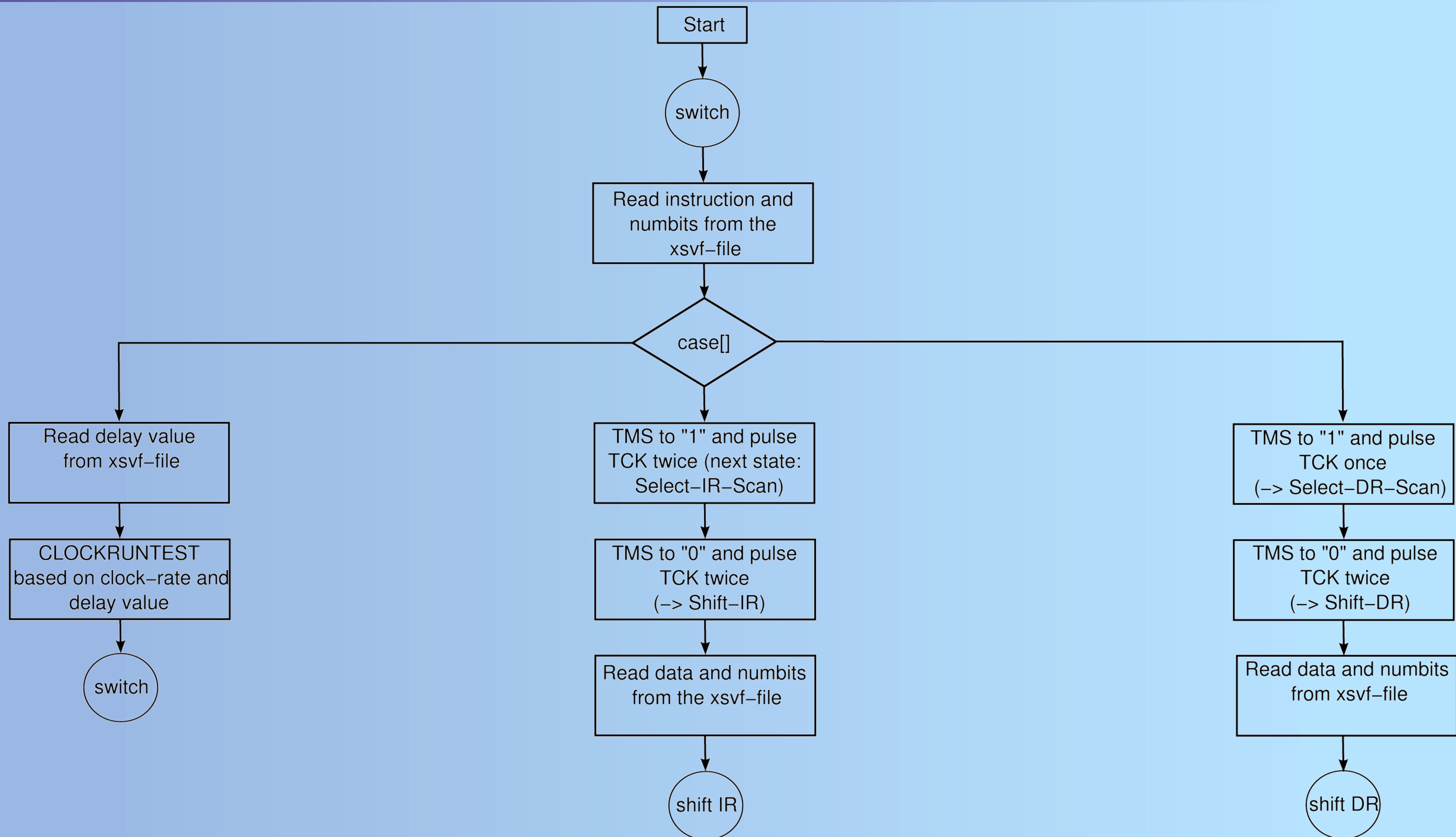
Die (Re-) Konfiguration lässt sich grob in drei Abschnitte einteilen.

2. Speicher leeren/Shutdown-Sequenz

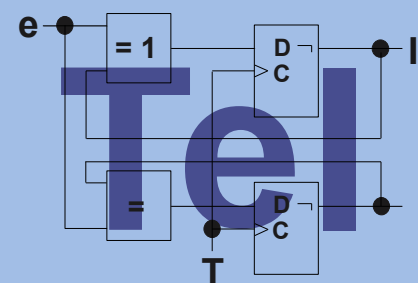
3. Konfigurations-Daten in den Speicher laden

4. Logik aktivieren mittels einer Start-Up-Sequenz





- Literaturstudium
  - Agenten
  - Organic Computing
  - Rekonfigurationsprinzipien von Xilinx-FPGAs
  - Graphentheorie für die Ablaufsteuerung
- Einarbeitung in die Prototypingplattform von ATMEL und die Entwicklungsumgebungen iMPACT, WinAVR und AVRStudio
- Einarbeitung in Quelltexte
- Aktuelle Implementationsarbeiten betreffen das Programm zur Rekonfiguration des FPGAs über JTAG
- Erstellung der schriftlichen Arbeit



- Ziel: Entwicklung von Methoden für die Selbstorganisation
- Mikrocontrollerbasiertes Management der Agenten-HW
- Implementierung der Software zur Rekonfiguration und des Scheduling
- Beispiele und Szenarien zur Demonstration der mikrocontrollergestützten Anpassungsfähigkeit rekonfigurierbarer Hardware finden
- Auswertung der Ergebnisse