



Entwicklung eines Lehrpraktikums auf Basis der Zynq-Plattform

Projektarbeit im Rahmen des Hauptseminars

Ludger Irsig

Dresden, 11.02.2015



Gliederung

- 01 Aufgabenstellung**
- 02 Die Zynq-Plattform**
- 03 Das ZedBoard**
- 04 Inbetriebnahme des ZedBoards**
- 05 Implementierung eines einfachen Beispiels**
- 06 Anpassung des Beispiels**
- 07 Dynamische Rekonfiguration**
- 08 Fazit**

01 Aufgabenstellung

Entwicklung eines Lehrpraktikums auf der Basis der Zynq-Plattform

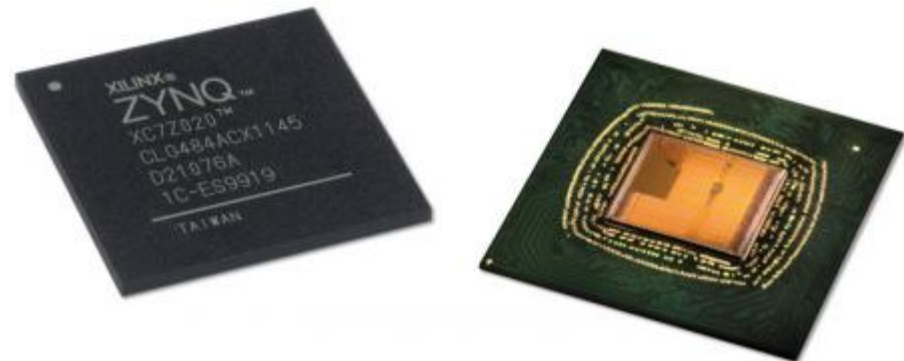
Inbetriebnahme einer Zynq-Plattform (ZedBoard) mit Ausarbeitung und Dokumentation einer demonstrativen Beispielanwendung

- der Praktikumsversuch könnte im Rahmen des Basismoduls INF-BAS-5 „Einführung in die Technische Informatik“ angeboten werden

02 Die Zynq-Plattform

Xilinx Zynq-7000 All Programmable SoC

- programmierbares
System-on-a-Chip
- kombiniert einen
ARM Dualcore
auf Cortex-A9-Basis
mit FPGA-Bestandteilen
aus der Xilinx 7-Serie

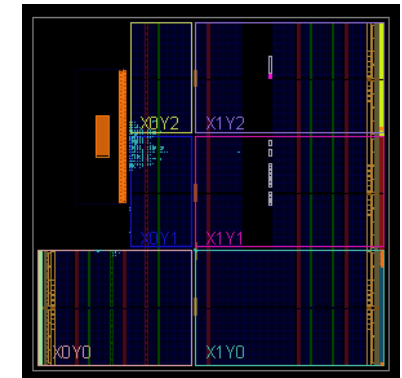
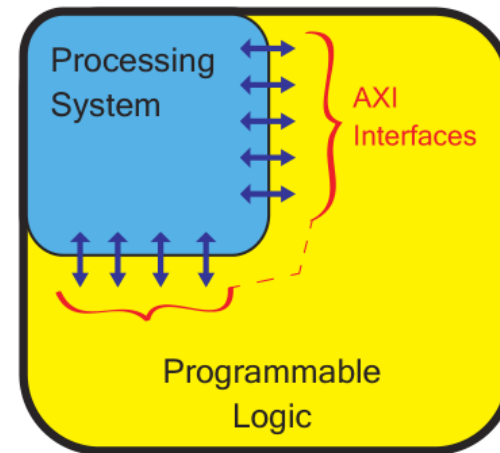


http://www.rtcmagazine.com/files/images/3339/rtc1204pr_xil_fig01_medium.jpg

02 Die Zynq-Plattform

Zusammenspiel

- Processing System (PS):
ARM Dualcore
- Programmable Logic (PL):
FPGA-Logik
- AXI Interface:
Advanced eXtensible Interface
(AMBA v3)
 - entwickelt von ARM
 - wird von Xilinx seit der
Spartan-6/Virtex-6-Generation
zur Anbindung von IP verwendet [1]



The Zynq Book, Fig. 1.2 / eigene

02 Die Zynq-Plattform

Vorteile

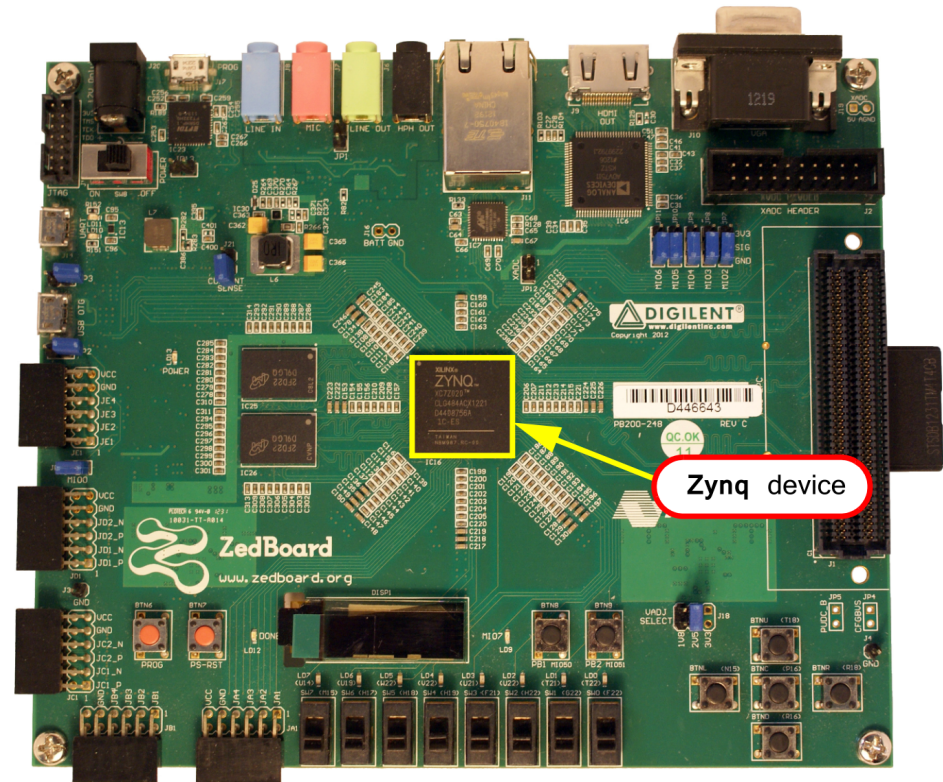
- Prozessor ist einfach zu programmieren,
FPGA ist für spezielle Berechnungen schneller
→ nutze FPGA da, wo es sich lohnt
- platzsparende Unterbringung auf einem Die
- fest integrierter Prozessor bietet mehr Performance als Soft IP
- dynamische Rekonfiguration

03 Das ZedBoard

- Development Kit für den Zynq-7000-Chip

Ergänzt den Chip um:

- 512MB DDR3-RAM
- HDMI, VGA, Audio
- Gbit-Ethernet
- UART
- SD-Karten-Slot
- Buttons, Switches, LEDs
- OLED-Display
- noch mehr Anschlüsse



The Zynq Book, Fig. 1.6

04 Inbetriebnahme des Zedboards (Praktikumsaufgabe 1)

- mitgelieferte SD-Karte
- beinhaltet bootbares Linux-System
- zur Demonstration einiger Funktionen

Vorbereitung:

- Jumper auf Zedboard setzen
- SD-Karte einlegen
- Kabel verbinden (USB, Strom)
- Power-Schalter umlegen

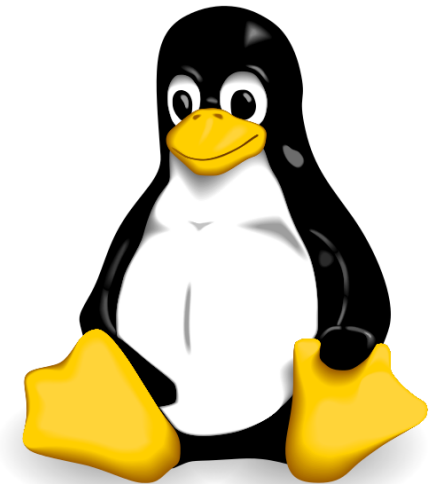
Auf PC:

- Treiber für serielle Schnittstelle installieren
- Terminal einrichten (z.B. PuTTY)

04 Inbetriebnahme des Zedboards (Praktikumsaufgabe 1)

Ergebnis:

- Terminalausgabe
- über das Terminal
 - im Linux-Dateisystem umhernavigieren
 - LEDs ein-/ausschalten
 - Switch-Stellung auslesen
 - OLED aktivieren/deaktivieren
- Testbild auf HDMI & VGA
- HTTP-Server



<https://commons.wikimedia.org/wiki/File:Tux.svg>

05 Implementierung eines einfachen Beispiels (Praktikumsaufgabe 2)

Ziel: ein erstes eigenes Zynq-Projekt implementieren

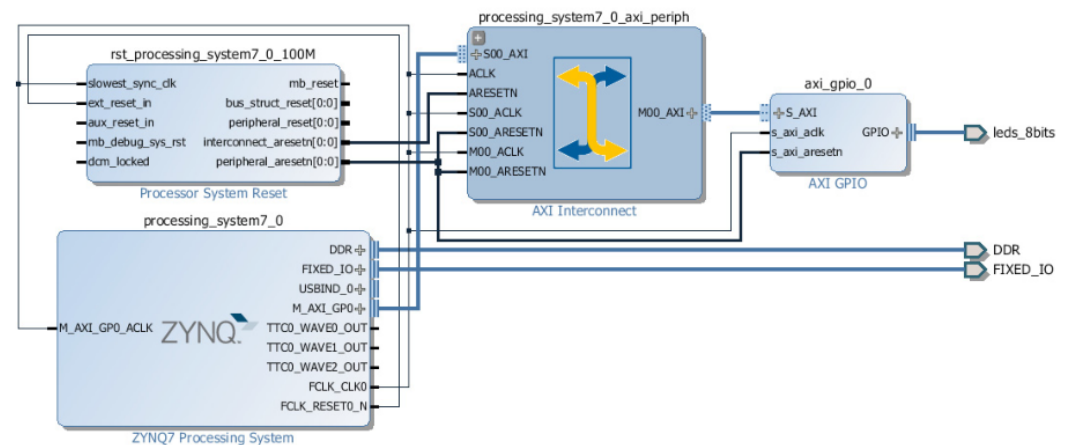
- Hardwaredesign mit Vivado 2014.4
- Softwaredesign mit Xilinx SDK
- ausführliche, bebilderte Anleitung im Zynq-Book-Tutorial (Tutorial 1)
- Ansteuerung der 8 LEDs
- HW: AXI-GPIO-Block
- SW: Ausgabe von Werten über AXI-Treiber

05 Implementierung eines einfachen Beispiels (Praktikumsaufgabe 2)

Hardwaredesign mit Vivado 2014.4

- im Lego-Prinzip IP-Blöcke zusammensetzen
- Connection Automation
- kein HDL schreiben

The Zynq Book Tutorials, Fig. 1.17



Softwaredesign mit Xilinx SDK

- vollständiger, dokumentierter Code wird bereitgestellt

06 Anpassung des Beispiels (Praktikumsaufgabe 3)

- bisher nur „nachmachen“, jetzt „selber machen“

Überlegung:

- zeigen, wie man eigene Hardware implementiert
- „einfache“ Schnittstellen: nur LEDs und Switches
- Einarbeiten in AXI nach Möglichkeit vermeiden

„Before beginning an AXI design, you need to download, read, and understand the ARM AMBA AXI Protocol v2.0 Specification, along with the AMBA4 AXI4-Stream Protocol v1.0.“ [1]

06 Anpassung des Beispiels (Praktikumsaufgabe 3)

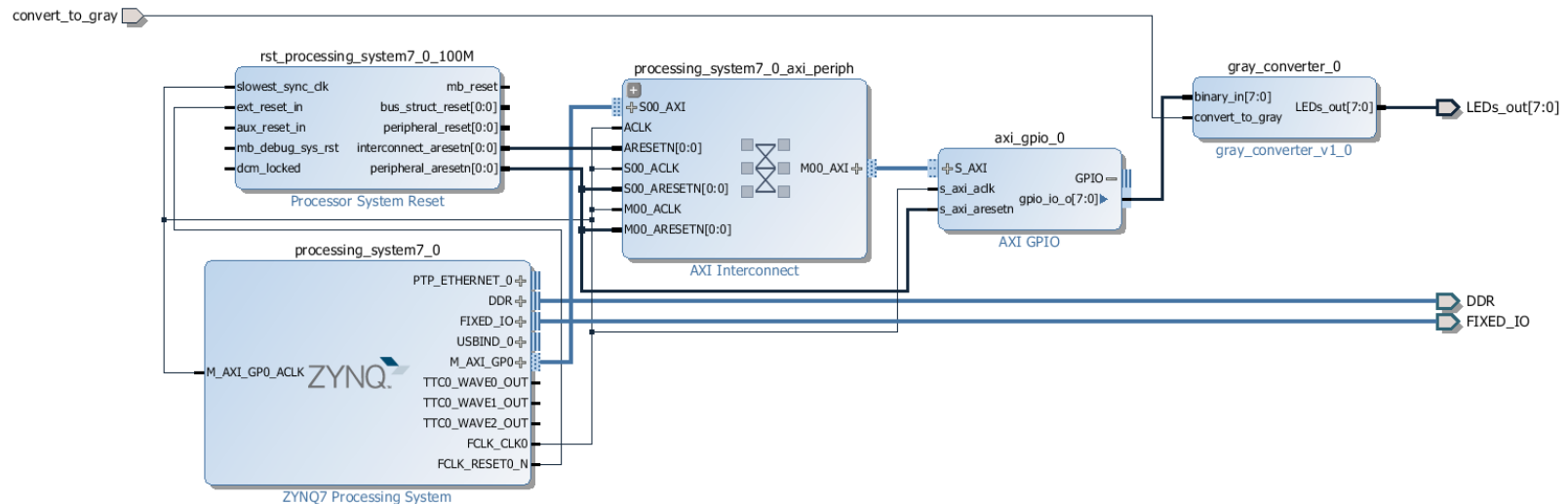
Resultat der Überlegung:

- VHDL-Modul, welches eine Binärzahl in Gray-Code umwandelt
- soll hinter dem AXI-GPIO-Block eingefügt werden

Herausforderungen:

- eigenes IP-Projekt erstellen
- (Gray-Code generieren)
- IP-Quelle importieren
- keine automatische Verdrahtung mehr verfügbar, Constraints-File anlegen

06 Anpassung des Beispiels (Praktikumsaufgabe 3)



- außerdem Anpassung der Software:
Ausgabewert inkrementieren statt negieren

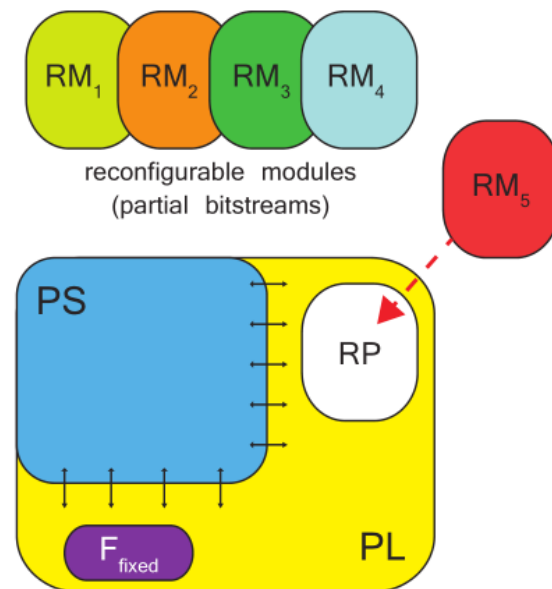
07 Dynamische Rekonfiguration

das heißt:

- im laufenden Betrieb die programmierbare Logik neu konfigurieren

Vorteil:

- Hardware kann an die aktuellen Anforderungen angepasst werden
- es können mehr Coprozessoren zur Verfügung gestellt werden, als eigentlich auf die Fläche passen „time-multiplexing“



The Zynq Book, Fig. 5.9

07 Dynamische Rekonfiguration

Umsetzbarkeit

zwei Ansätze:

Eigenes Programm, dass ICAP nutzt

- ICAP = Internal Configuration Access Port

Funktion des mitgelieferten Linux-Systems

- `cat *.bit.bin > /dev/xdevcfg`

07 Dynamische Rekonfiguration

Umsetzbarkeit (II)

zwei Sackgassen:

Eigenes Programm, dass ICAP nutzt

- Wissen über ICAP, AXI und das Bitfile-Format benötigt
- Beispielcode >450 Zeilen

Funktion des mitgelieferten Linux-Systems

- ein modifiziertes Bitfile muss erstellt werden
- Projektdateien nur für Xilinx Platform Studio
- LEDs hier nicht über AXI angeschlossen

08 Fazit

Ziel war:

„Inbetriebnahme einer Zynq-Plattform (ZedBoard) mit Ausarbeitung und Dokumentation einer demonstrativen Beispielanwendung“

→ erreicht

Möglich im Rahmen des Praktikums:

- Grundlegende Fähigkeiten des ZedBoards zeigen
- Einführung in den Entwurfsfluss und die Softwarewerkzeuge
- Lust auf mehr wecken?

Nicht möglich im Rahmen des Praktikums:

- das Potential der Zynq-Plattform auch nur annähernd auszunutzen

Bonus-Folie

AXIOM
Alpha by apertus^o

The first step to an open source digital cinema camera.

Image Sensor:
Global Shutter
4K Resolution
4:3 aspect ratio
Super35 diameter

Recording:
Uncompressed raw 4K still images
Full HD HDMI (up to 60 FPS Deep Color)

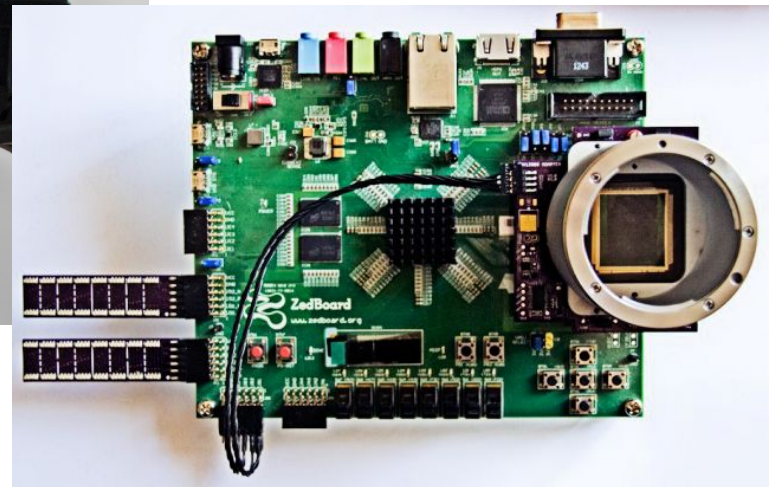
Misc:
Ethernet & USB Interfaces
powered by Linux
battery powered for mobile application

1080P Full HD 12 MPixels Deep Color powered by Linux

www.apertus.org



apertus Axiom:
eine Open-Hardware
Free-Software
4K-Kamera



https://www.apertus.org/alpha_prototype

<http://forums.xilinx.com/t5/Xcell-Daily-Blog/Zynq-based-Axiom-Alpha-open-4K-cine-camera-proto-debuts-in/ba-p/430066>

Quellen

- [1] AXI Reference Guide (Xilinx UG761)
http://www.xilinx.com/support/documentation/ip_documentation/axi_ref_guide/v13_4/ug761_axi_reference_guide.pdf
- The Zynq Book & The Zynq Book Tutorials, <http://zynqbook.com/>



»Wissen schafft Brücken.«