TECHNISCHE UNIVERSITÄT DRESDEN

Fraunhofer IWS

Fakultät Informatik, Institut für Technische Informatik

ABLEITUNG ZUSÄTZLICHER STEUERGRÖSSEN FÜR DIE STRAHLFORMUNG IN EINEM LASER-SCANNER IM ECHTZEITBETRIEB

Studienarbeit

Patrick Schöps

Dresden, 09.02.2017

Überblick

Motivation

Analyse

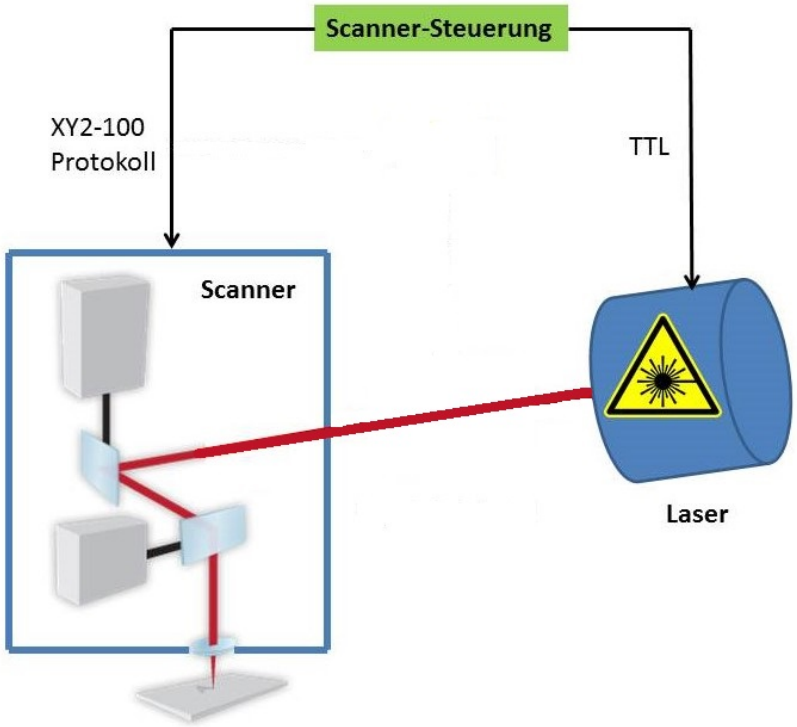
Entwurf und Implementierung

Test

Fazit und Ausblick

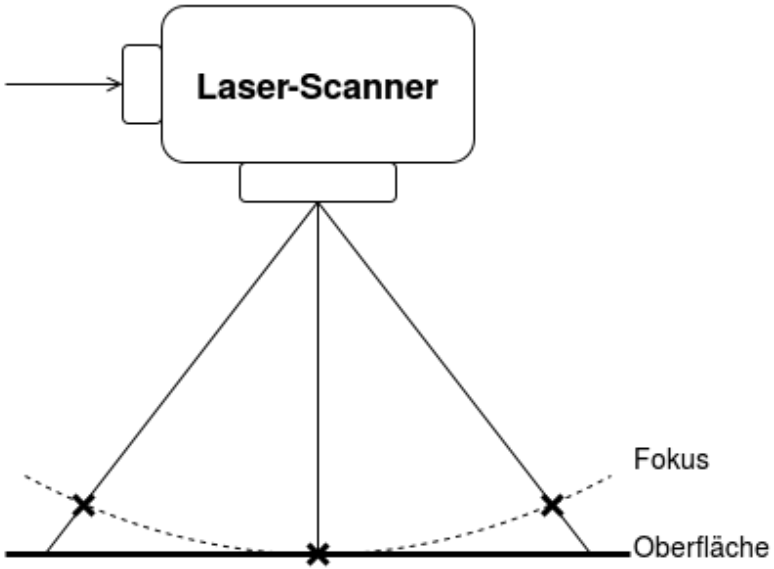
01 Motivation
Ausgangssituation

- Werkstoff- und Strahlforschung
- Scanner:
  - Galvanometer
  - Ablenkung in X/Y-Richtung

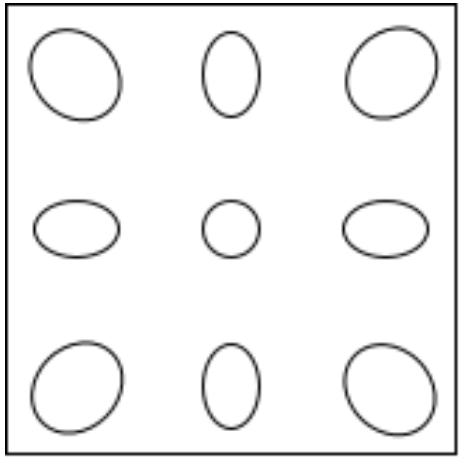


01 Motivation

Nebeneffekte von optischen Projektionsverfahren



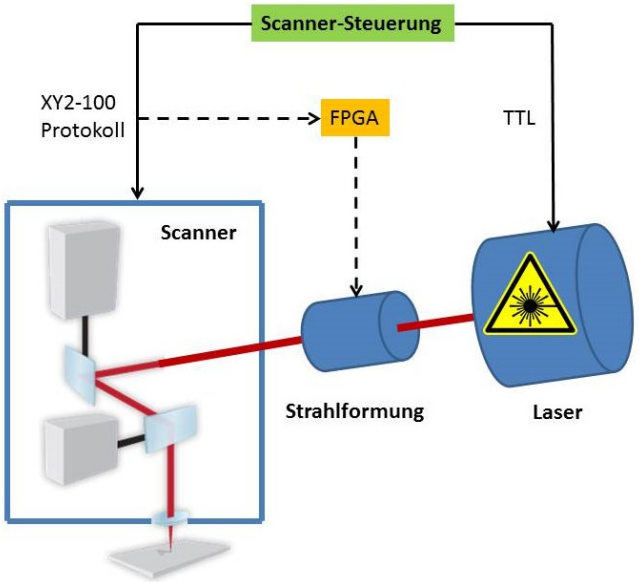
Abweichung des Fokus von der Oberfläche



Verzerrung im Außenbereich

01 Motivation
Lösungsansatz

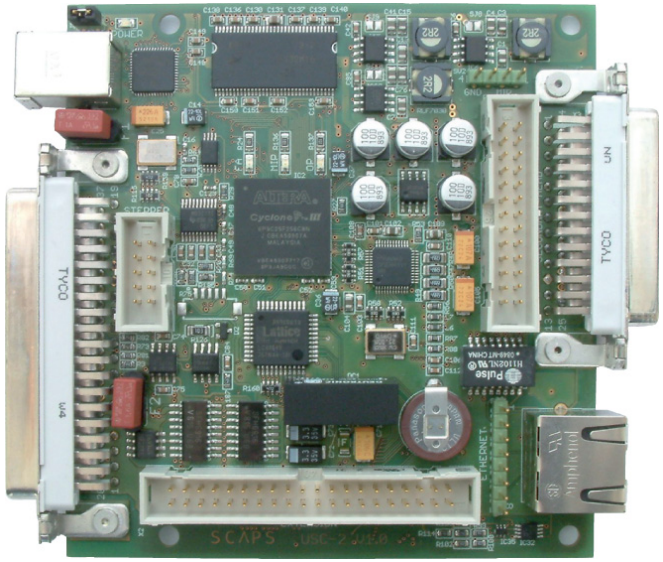
- weitere Einheit zur Strahlformung
- zusätzliche Steuergröße
- Logikteil des ZYBO:
  - Überwachung der Ansteuerung des Laser-Scanners
  - Ableiten und Ausgeben der Steuergröße als Analogpegel
  - Echtzeitzusicherung
- ARM-Prozessor:
  - Nutzerinterface
  - Änderung der Ausgabepegelfunktion



02 Analyse

Laser-Scanner

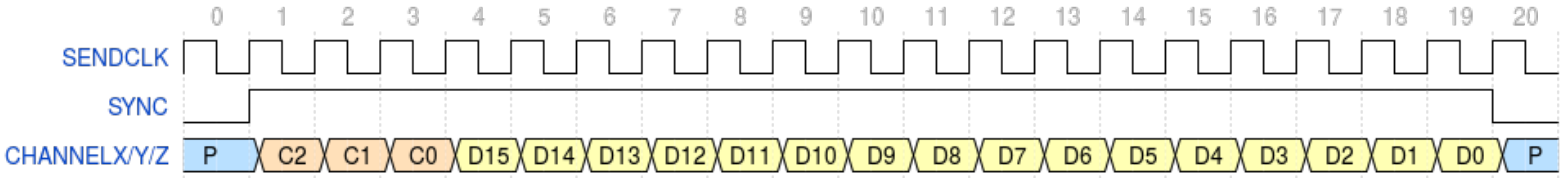
- USC-2 von SCAPS GmbH
- Steuersignale für Laser und Scanner
- programmierbar über USB und Ethernet
- Stand-Alone-Modus:
  - interner Flash-Speicher für Jobs
  - Laden und Ausführen von Jobs über Flash Control Interface mittels serieller Schnittstelle möglich



Scanner-Platine USC-2

02 Analyse

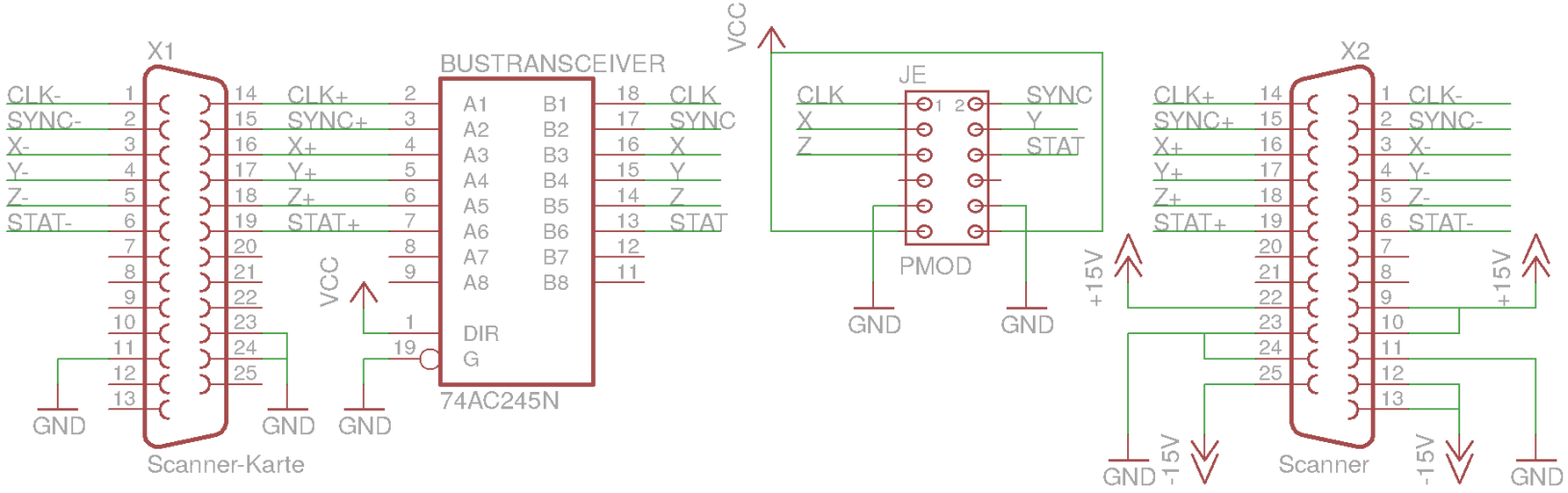
XY2-100-Protokoll



- synchrones, differenzielles und digitales Interface
- differenzielle Signalpaare für Sendetakt, Synchronisationssignal, zwei oder drei Datenkanäle und einen Rückkanal
- Positionsdaten in Exzesscode

02 Analyse

Pegelanpassung



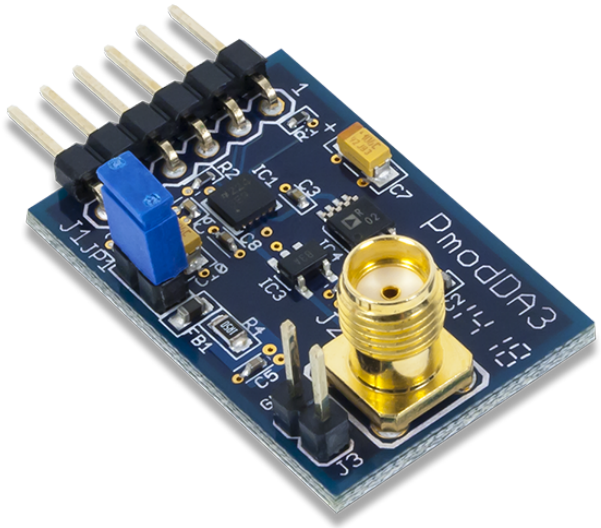
- Herunterpegeln der positiven Signale auf 3.3V mit Hilfe eines Bustransceivers

02 Analyse

Digital Analog Konverter

- PmodDA3 von Digilent
- 16-Bit-DAC
- SPI-ähnliches Protokoll:
  - 50 MHz Sendetakt
  - low-aktives Chip-Select
  - Datenkanal (16-Bit-Stream)
  - low-aktives LDAC

Binär		Analog
X"0000"	$\longrightarrow$	0 V
X"8000"	$\longrightarrow$	$(1/2) \times V_{ref}$
X"FFFF"	$\longrightarrow$	$(65535/65536) \times V_{ref}$

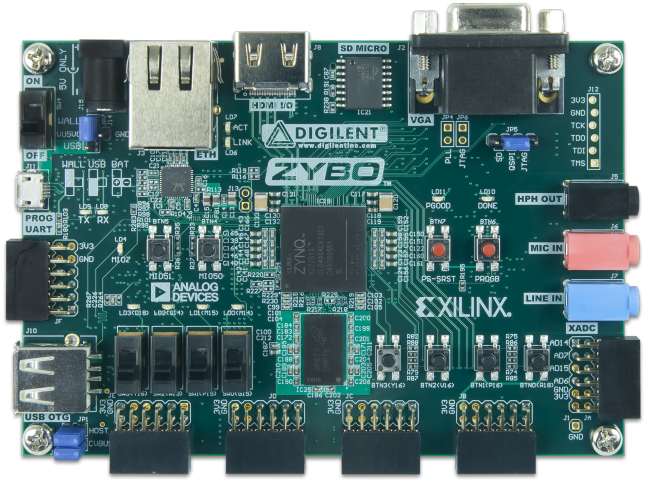


PmodDA3

02 Analyse

ZYBO - Hardware

- ZYnq BOard von Digilent
- All Programmable System-on-Chip Architektur:
  - Dual-Core ARM Cortex-A9 Prozessor
  - FPGA-Logik (7er-Serie)



ZYnq BOard

02 Analyse

ZYBO - Hardware

- ARM Cortex-A9:
  - 650 MHz Takt
  - 512 MB DDR3 Speicher
  - Ethernet, USB 2.0, SDIO
  - SPI, UART, CAN, I^2C
- Artix-7 FPGA:
  - 4400 Logikeinheiten
  - 240 kB BRAM
  - zwei Taktmanager
  - 80 DSP-Einheiten
- weitere Komponenten:
  - Onboard-JTAG-Programmierer und UART-zu-USB-Konverter
  - GPIO (Taster, Schalter, Buttons und LED)
  - sechs PMOD-Verbindungen

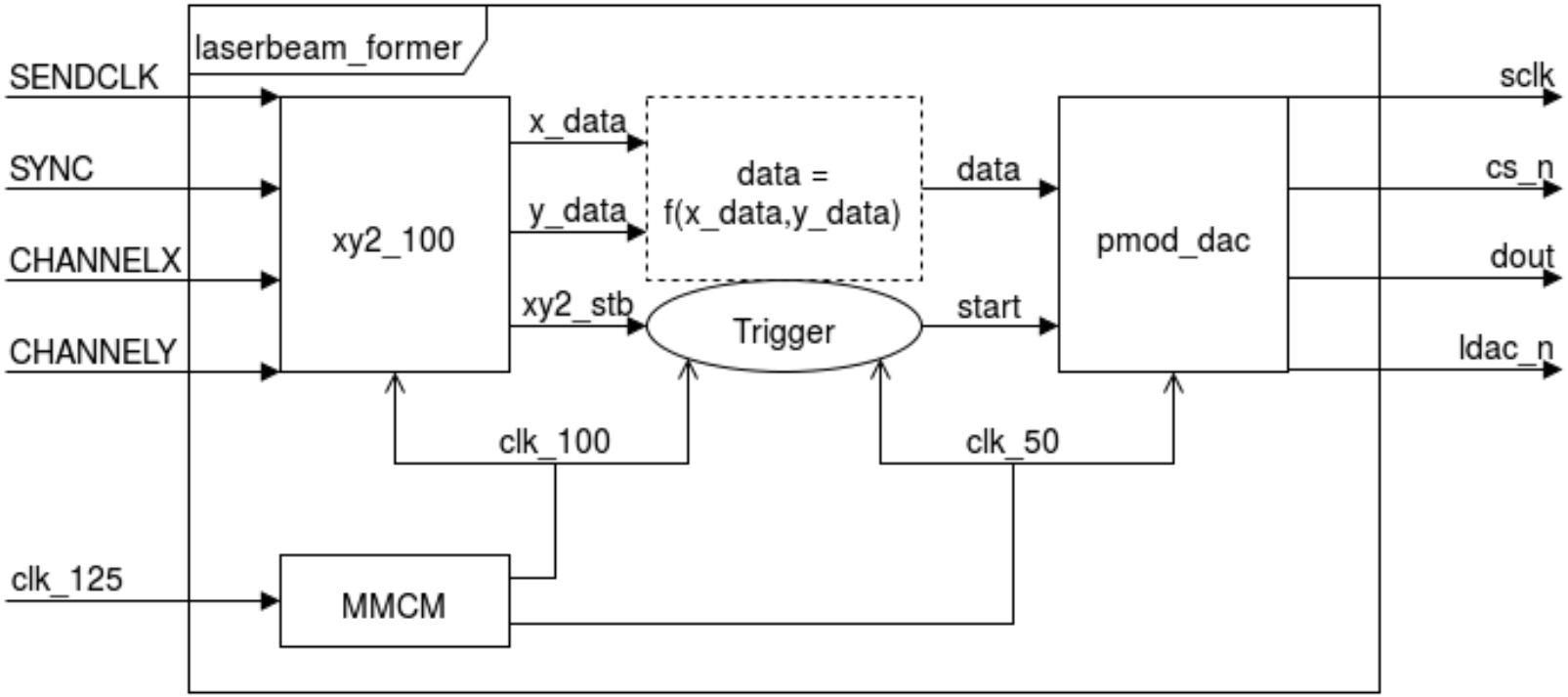
02 Analyse

ZYBO - Software

1. Entwurf eines Hardware-Design in Vivado
2. Cross-Compilierung eines BOOT-Images (FSBL + uboot)
3. Cross-Compilierung des Linux-Kernel und des Device-Baums
4. Grundinstallation von Debian in Chroot (change root)
5. Transfer des Systems auf eine SD-Karte
6. Installation des Contiguous-Memory-Allocation-Treibers in Debian

03 Entwurf und Implementierung

Übersicht



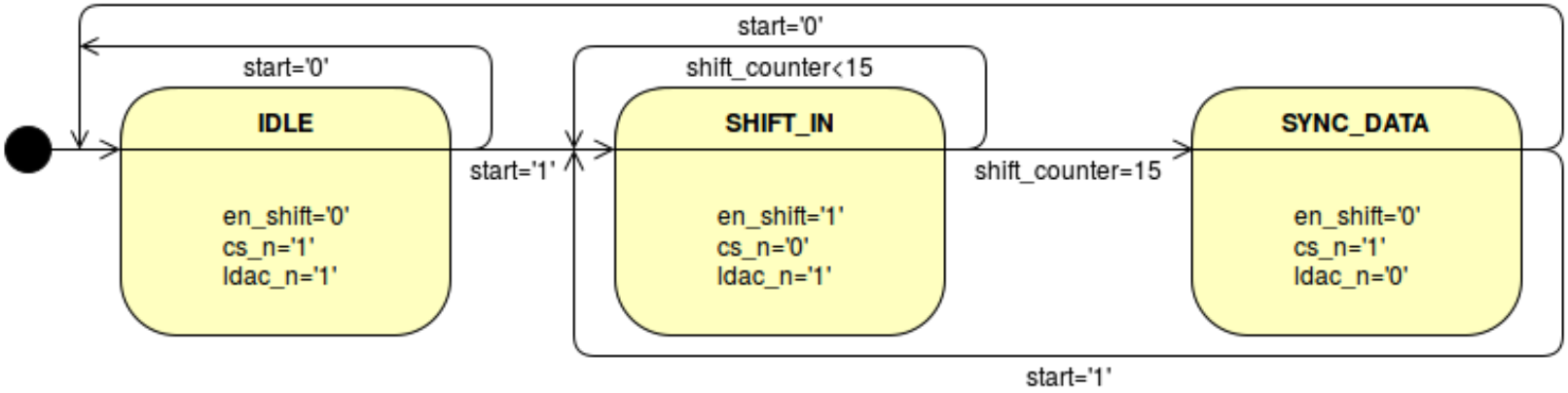
03 Entwurf und Implementierung

xy2_100

- interpretiert XY2-100-Protokoll
- Ausgabe empfangener Koordinaten
- Strobe-Signal bei Aktualisierung einer Koordinate

03 Entwurf und Implementierung

pmod_dac



- Zustandsautomat
- Übertragung des Ausgabepegel an den DAC
- durch Start-Signal ausgelöst

03 Entwurf und Implementierung

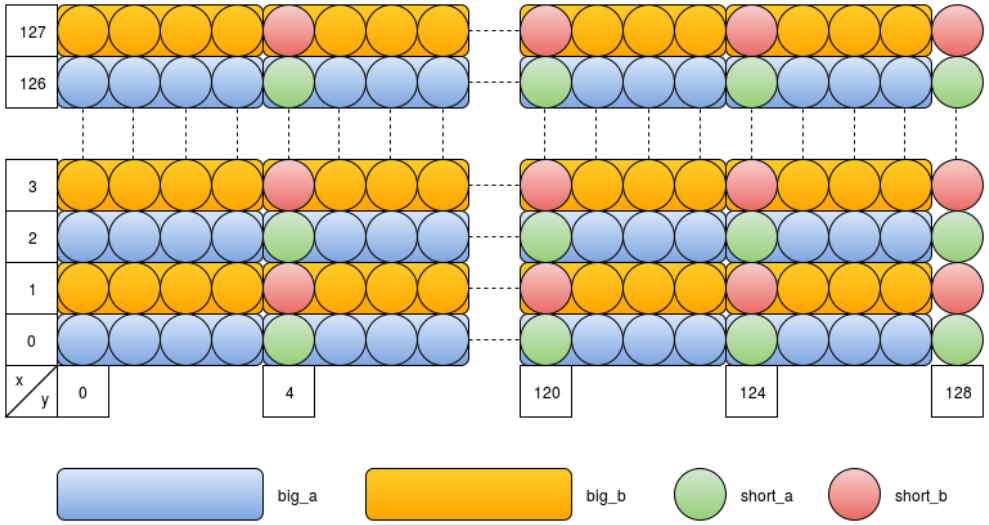
laserbeam_former

- Instanziierung von xy2_100 und pmod_dac
- Trigger-Synchronisation
- Berechnung des Ausgabewertes in Abhängigkeit der eingelesenen Koordinaten
- als IP Core im Hardware-Design eingebunden

03 Entwurf und Implementierung

Speicherorganisation

- vorberechnete Werte im BRAM
- bilineare Interpolation mit 128×129 Stützstellen
- vier BRAM-Instanzen
- 25% Redundanz



03 Entwurf und Implementierung

Speicherzugriff

- Dual-Port-BRAM
- Port A: AXI-BRAM-Controller
- Port B: laserbeam_former

<table>
<tr><th rowspan="2"></th><th colspan="2">PORTA</th><th colspan="2">PORTB</th><th rowspan="2">virtuelle Adresse</th><th rowspan="2">Adress-bereich</th></tr>
<tr><th>lesend</th><th>schreibend</th><th>lesend</th><th>schreibend</th></tr>
<tr><td rowspan="2"><code>big_a</code>
<code>big_b</code>
<code>short_a</code>
<code>short_b</code></td><td colspan="4">$2^{11} \times 32$ Bit</td><td rowspan="2">0x43c1_0000
0x43c1_2000
0x43c1_4000
0x43c1_5000</td><td>8K</td></tr>
<tr><td colspan="2">$2^{9} \times 32$ Bit</td><td>$2^{11} \times 8$ Bit</td><td>$2^{9} \times 32$ Bit</td><td>2K</td></tr>
</table>

03 Entwurf und Implementierung

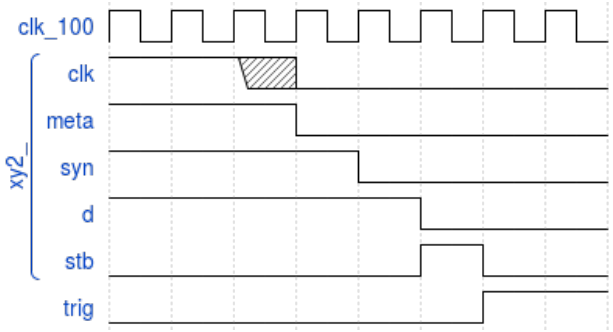
Nutzerinterface

- ipython3-notebook
- Python-HTTP-Server auf dem ZYBO
- graphische Oberfläche
- Ausführen von Python-Scripts möglich
- Paket `axi` mit Methoden `initial()`, `plot()`, `read()` und `write()`

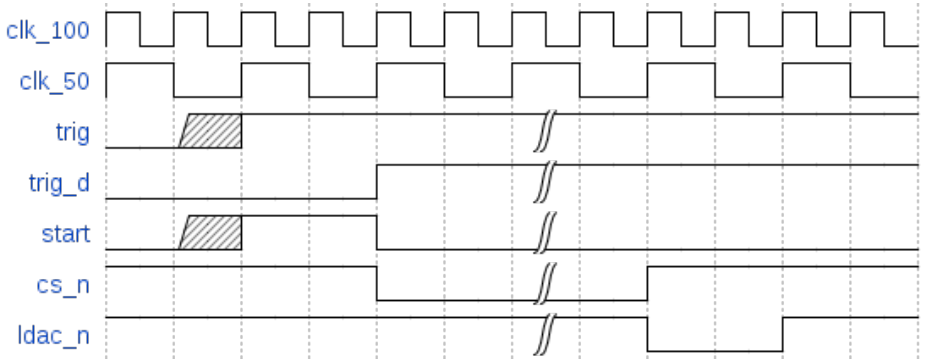
04 Test
Funktion

04 Test
Latenz

XY2-100: 30-40 ns



Trigger und DAC: 360-370 ns



05 Fazit und Ausblick

- System mit garantierter Latenz von 390-410 ns im Echtzeitbetrieb nutzbar
- Pegelfunktion aus Prozessor heraus durch Nutzer definierbar
- Verbesserungen:
  - Auflösung der Pegelwerte auf 16 Bit erhöhen
  - weitere Steuergrößen ableiten
  - vollständige Übernahme der Steuerung bei Wahrung der Echtzeitfähigkeit in Kombination mit WebAPI

Literatur

- SCAPS: USC-2. Website: http://www.scaps.com/index.php?id=26
- PYNQ-Projekt. Website: http://www.pynq.io
- Digilent: PmodDA3 und ZYBO Reference Manual
- Xilinx: User und Product Guides