

Vorlesung Architekturen und Entwurf von Rechnersystemen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Prof. Andreas Koch, Yannick Lavan, Johannes Wirth, Mihaela Damian

Wintersemester 2022/2023
Theorieblatt 4

4.1 AXI4

In dieser Aufgabe beschäftigen Sie sich noch mal genauer mit den verschiedenen AXI4 Varianten. Sie können für die Abhängigkeiten der Handshake-Signale die AXI Spezifikation[1] zur Hilfe nehmen.

4.1.1 AXI4 vs. AXI4-Lite vs. AXI4-Stream

Erläutern Sie die wesentlichen Unterschiede der Protokolle AXI4, AXI4-Lite und AXI4-Stream. Gehen Sie bei AXI4 und AXI4-Lite auch auf die Unterschiede aus Hardware-Sicht ein.

4.1.2 AXI vs. Bluespec

Wie unterscheiden sich Bluespecs RDY/EN-Handshake von AXIs Valid/Ready-Handshake?
Hinweis: Machen Sie sich klar warum (* always_enabled *) in Bluespec (* always_ready *) impliziert.

4.1.3 Transfers

Sie können für diese Aufgabe die bereitgestellten Tabellen benutzen.

a) Nehmen Sie an, dass das Protokoll AXI4-Lite für die Transfers in Abbildung 1 verwendet wurde. Geben Sie den Zustand des Slave-Moduls nach Durchführung aller Transfers an. Die Datenworte und Adressen sind in hexadezimaler Notation gegeben.

Adresse	Wert (hexadezimal)
0x00000000	
0x0815ADD0	
0x13374224	
0x4213372C	
0x42C01330	
0xFA57F000	

Signalname Signalverlauf

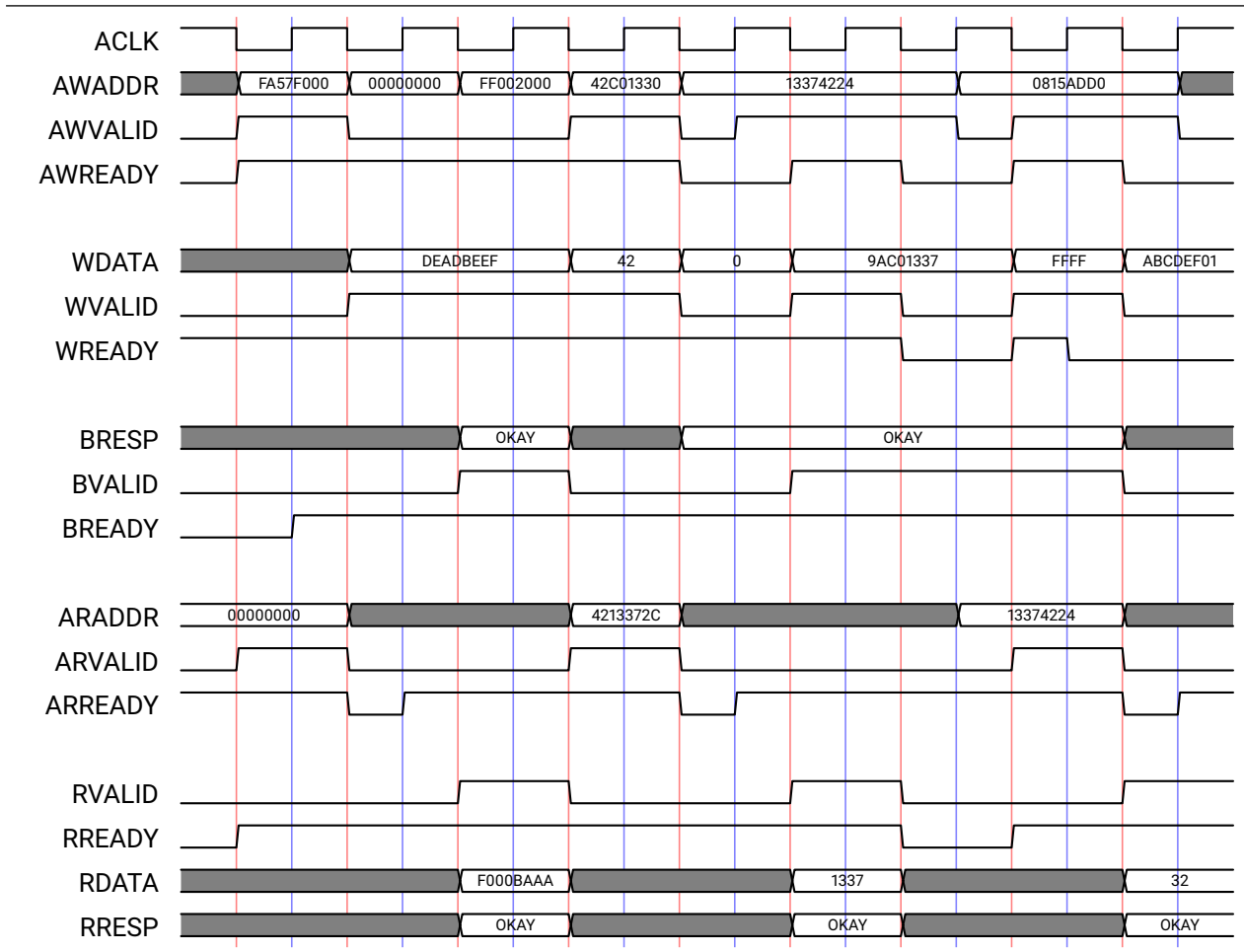


Abbildung 1: AXI4-Lite Transfers

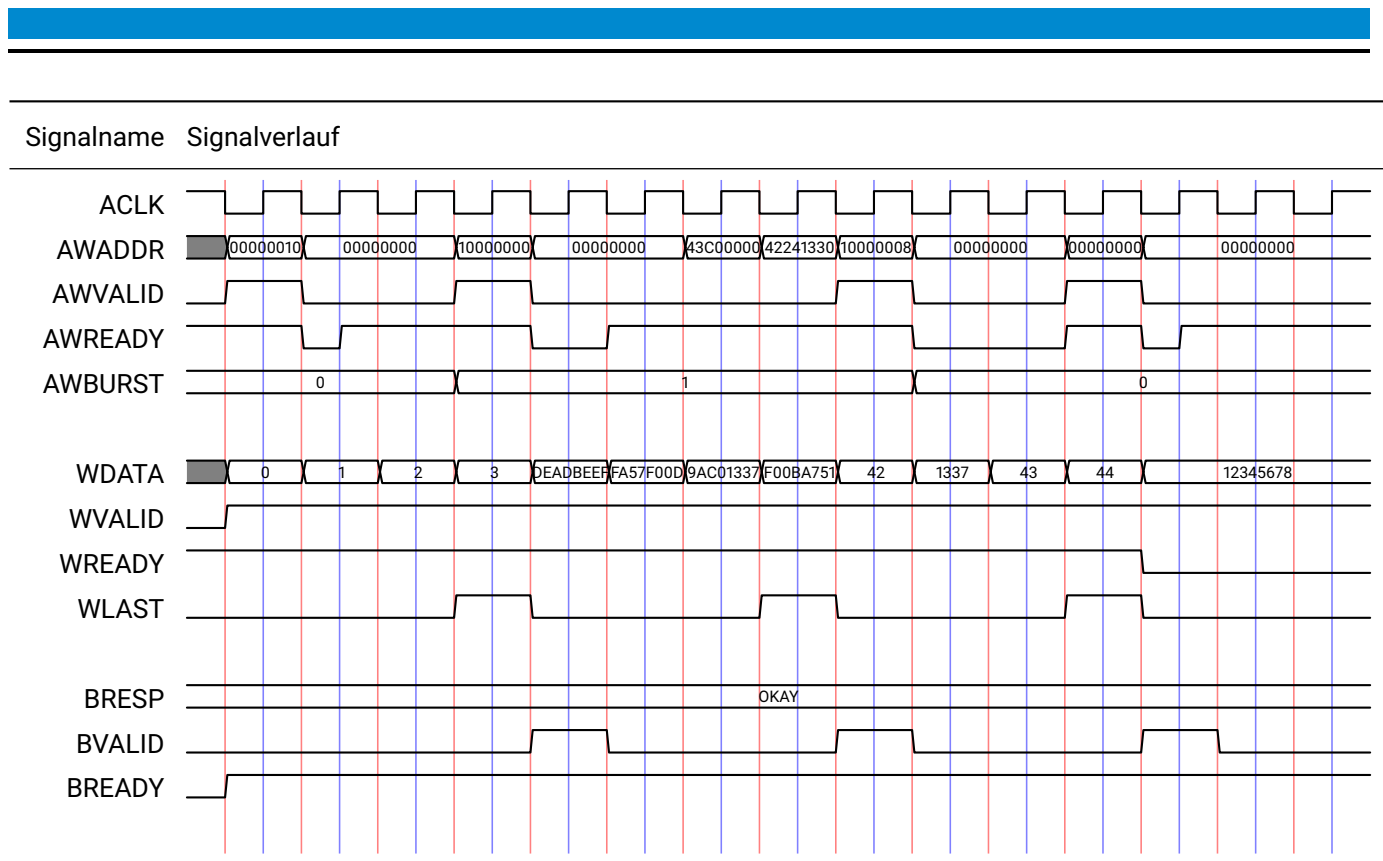


Abbildung 2: AXI4-Transfers auf dem Write-Kanal

b) In Abbildung 2 wurde AXI4 verwendet. Für uns uninteressante Signale wurden im Diagramm ausgelassen. Geben Sie auch hier den Zustand des Slave-Speichers nach Beendigung aller Transfers an. Alle Werte sind hexadezimal angegeben. Es gilt außerdem AxLEN=3, AxSIZE=2. Nehmen Sie weiterhin an, dass immer der letzte geschriebene Wert an der entsprechenden Speicher-Adresse steht.

Adresse	Wert (hexadezimal)

4.2 Rekonfigurierbares Rechnen und TaPaSCo

4.2.1 Task Parallel System Composer (TaPaSCo)

- Erläutern Sie die grundlegende Idee hinter TaPaSCo. Welche Probleme werden von TaPaSCo adressiert?
- Was ist eine DSE und wofür wird sie eingesetzt?
- Gegeben sei die aus der Vorlesung bekannte Address Map in Abbildung 3. An welche Adresse müssen Sie schreiben, wenn sie das Argument 1 Register von target_ip_00_000 setzen möchten? Sie möchten einen 32-Bit Wert aus dem Return Value Register lesen und den Interrupt bestätigen, nachdem target_ip_00_011 einen Interrupt gesetzt hat. Von

Host/ps7						
Data (32 address bits : 0x40000000 [1G], 0x80000000 [1G])						
	InterruptControl/axi_intc_00	s_axi	Reg	0x0180_0000	64K	0x0180_FFFF
	Threadpool/target_ip_00_000	s_axi_AXILiteS	Reg	0x43C0_0000	64K	0x43C0_FFFF
	Threadpool/target_ip_00_001	s_axi_AXILiteS	Reg	0x43C1_0000	64K	0x43C1_FFFF
	Threadpool/target_ip_00_002	s_axi_AXILiteS	Reg	0x43C2_0000	64K	0x43C2_FFFF
	Threadpool/target_ip_00_003	s_axi_AXILiteS	Reg	0x43C3_0000	64K	0x43C3_FFFF
	Threadpool/target_ip_00_004	s_axi_AXILiteS	Reg	0x43C4_0000	64K	0x43C4_FFFF
	Threadpool/target_ip_00_005	s_axi_AXILiteS	Reg	0x43C5_0000	64K	0x43C5_FFFF
	Threadpool/target_ip_00_006	s_axi_AXILiteS	Reg	0x43C6_0000	64K	0x43C6_FFFF
	Threadpool/target_ip_00_007	s_axi_AXILiteS	Reg	0x43C7_0000	64K	0x43C7_FFFF
	Threadpool/target_ip_00_008	s_axi_AXILiteS	Reg	0x43C8_0000	64K	0x43C8_FFFF
	Threadpool/target_ip_00_009	s_axi_AXILiteS	Reg	0x43C9_0000	64K	0x43C9_FFFF
	Threadpool/target_ip_00_010	s_axi_AXILiteS	Reg	0x43CA_0000	64K	0x43CA_FFFF
	Threadpool/target_ip_00_011	s_axi_AXILiteS	Reg	0x43CB_0000	64K	0x43CB_FFFF
	Threadpool/target_ip_00_012	s_axi_AXILiteS	Reg	0x43CC_0000	64K	0x43CC_FFFF
	Threadpool/target_ip_00_013	s_axi_AXILiteS	Reg	0x43CD_0000	64K	0x43CD_FFFF
	Threadpool/target_ip_00_014	s_axi_AXILiteS	Reg	0x43CE_0000	64K	0x43CE_FFFF
	Threadpool/target_ip_00_015	s_axi_AXILiteS	Reg	0x43CF_0000	64K	0x43CF_FFFF
	tpc_status	S00_AXI	S00_AXI_req	0x7777_0000	64K	0x7777_FFFF

Abbildung 3: Adressmap aus der TaPaSCo Vorlesung

4.2.2 Rekonfigurierbares Rechnen

- a) Was ist der Unterschied zwischen einer Lookup Table (LUT) und einem Configurable Logic Block (CLB)?
b) Setzen Sie die Funktion $Y = \bar{C}D + A\bar{B}$ in der untenstehenden LUT um, wobei + für das logische ODER steht. Setzen Sie hierfür $S_1 = 0$. Bilden Sie Y auf S_0 ab.

[illegible]

Wichtige Abkürzungen

CLB Configurable Logic Block

DMA Direct Memory Access

DSP Digital Signal Processor

FPGA Field-Programmable Gate Array

FPU Floating-Point Unit
LUT Lookup Table
PL Programmable Logic
PS Processing System
RAM Random-Access Memory
ROM Read-Only Memory
rSoC Reconfigurable System-on-Chip
TaPaSCo Task Parallel System Composer

Literatur

[1] http://www.gstitt.ece.ufl.edu/courses/fall15/eel4720_5721/labs/refs/AXI4_specification.pdf.