FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

CPU-Entwicklung mit VHDL

On-Chip Bussysteme und Perpherie (Schwerpunkt CPU Entwicklung)

Philipp Holzinger
Informatik 3 / Rechnerarchitektur
Friedrich-Alexander Universität Erlangen-Nürnberg

Sommersemester 2025

Motivation

- Reiner Prozessor Core nicht ausreichend
- Verbindung mit Hauptspeicher
- Schnittstellen nach Außen erforderlich
- Deshalb: Standardisierung von Bussystemen

Gliederung

- On-Chip Busprotokolle
  - Allgemeine Busstruktur
  - ARM AMBA AXI
- Off-Chip Busprotokolle
  - Allgemeine Eigenschaften
  - UART
  - IIC
  - SPI
  - VGA

Allgemeine Busstruktur

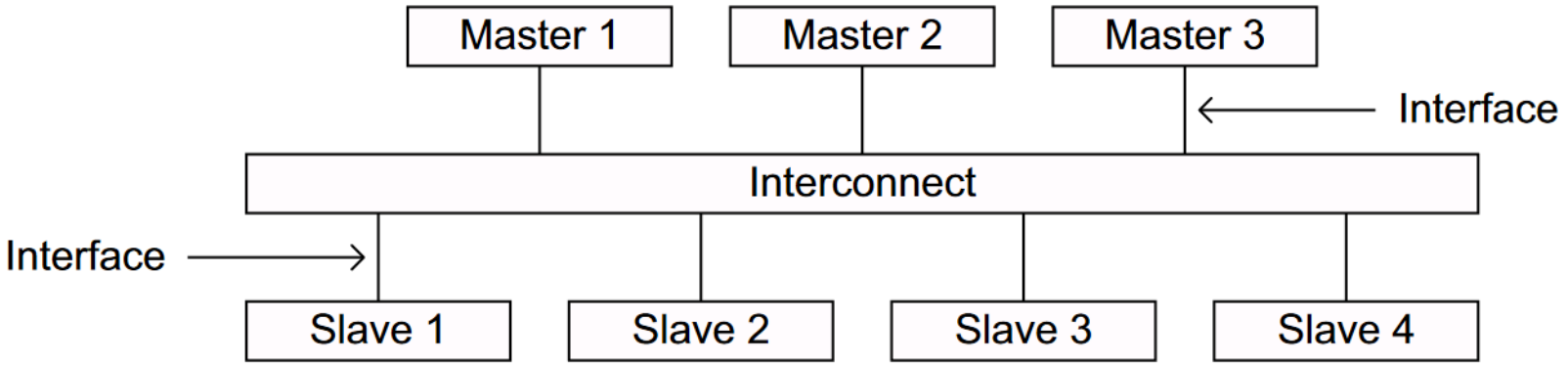
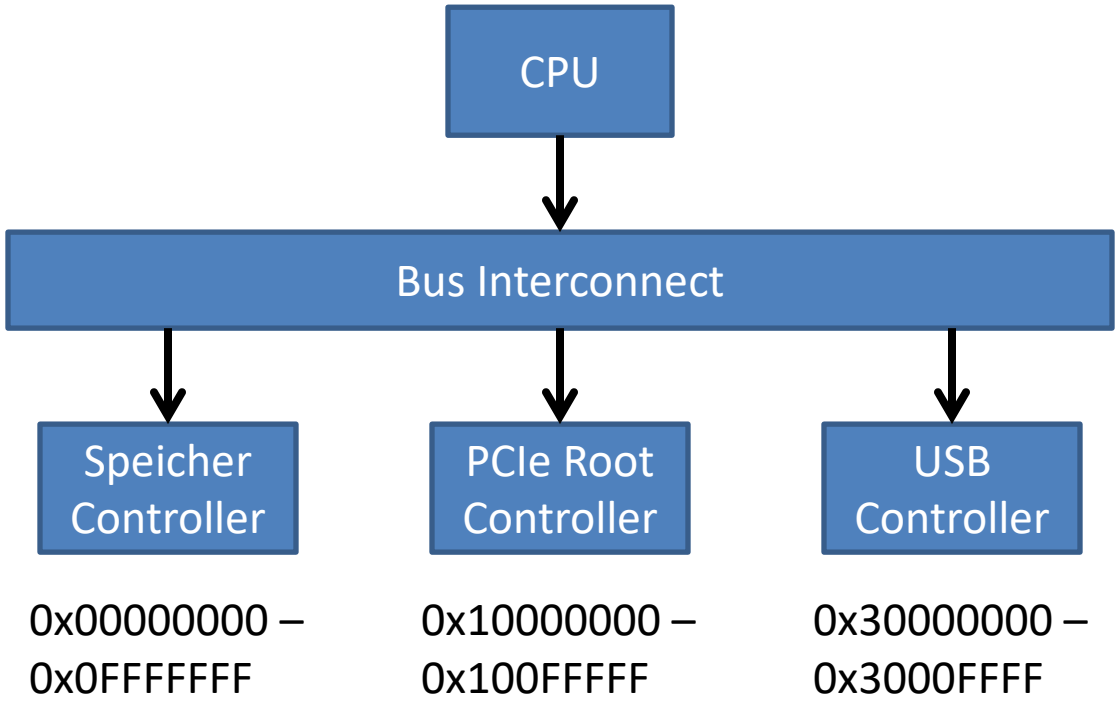


Figure A1-3 Interface and interconnect

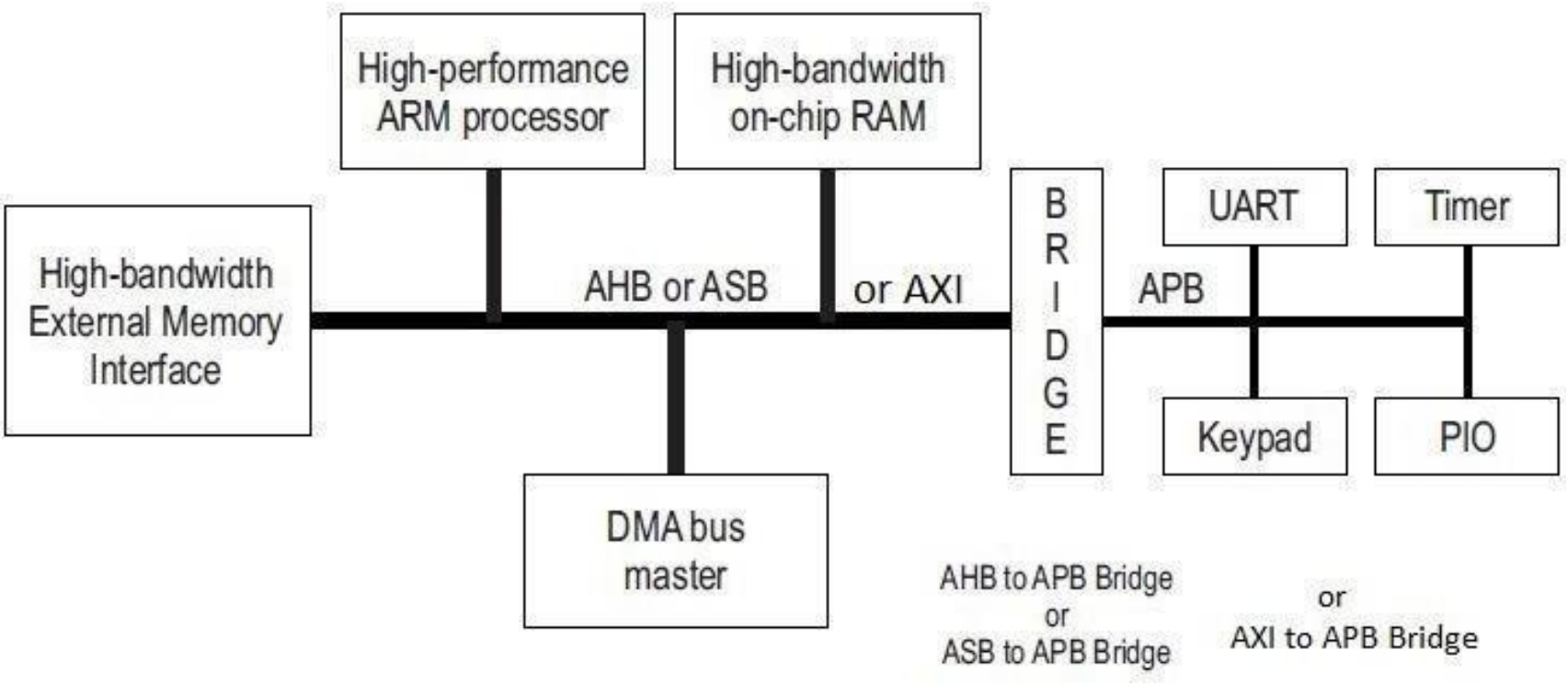
- Zwei Arten von Komponenten:
  - **Master**: Stellt selbst Lese- und Schreibanfragen über den Bus
  - **Slave**: Beantwortet Lese- und Schreibanfragen die ankommen
- Verbindung der Komponenten über Bus **Interconnect**

Allgemeine Busstruktur



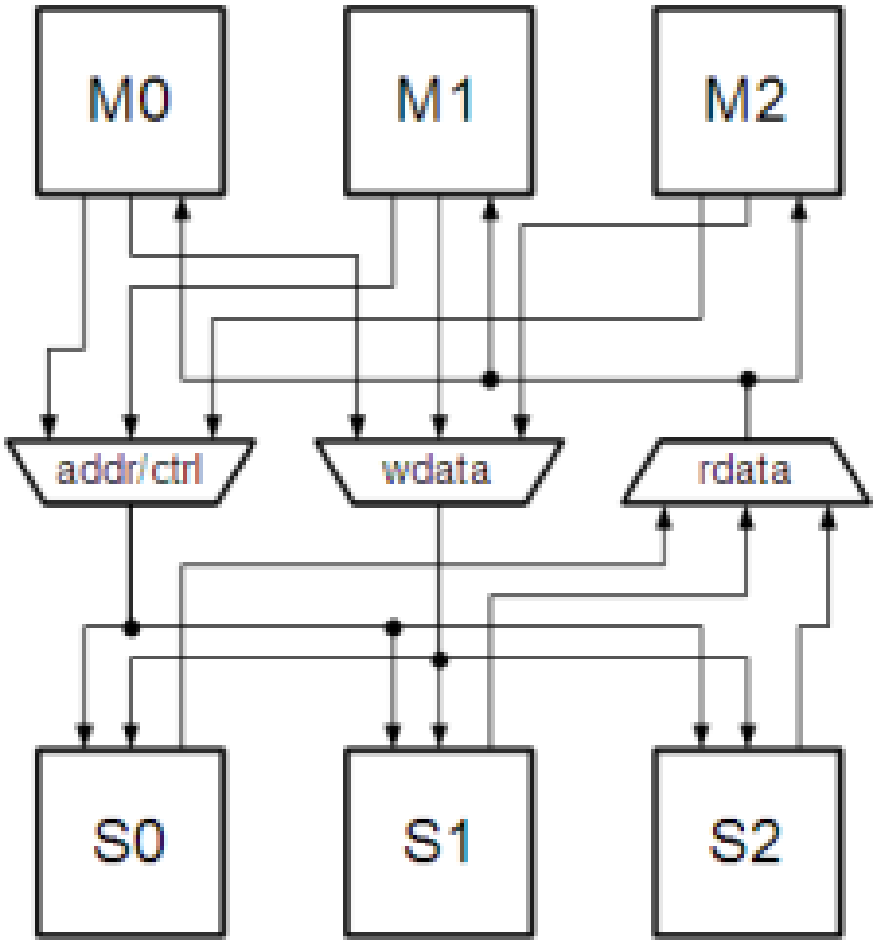
- CPU und Peripherie Controller sind an on-Chip Bussystem angebunden
- Jedem Controller wird statisch (Device Tree) oder dynamisch (ACPI) ein ausreichend großer Adressbereich zugewiesen

Allgemeine Busstruktur

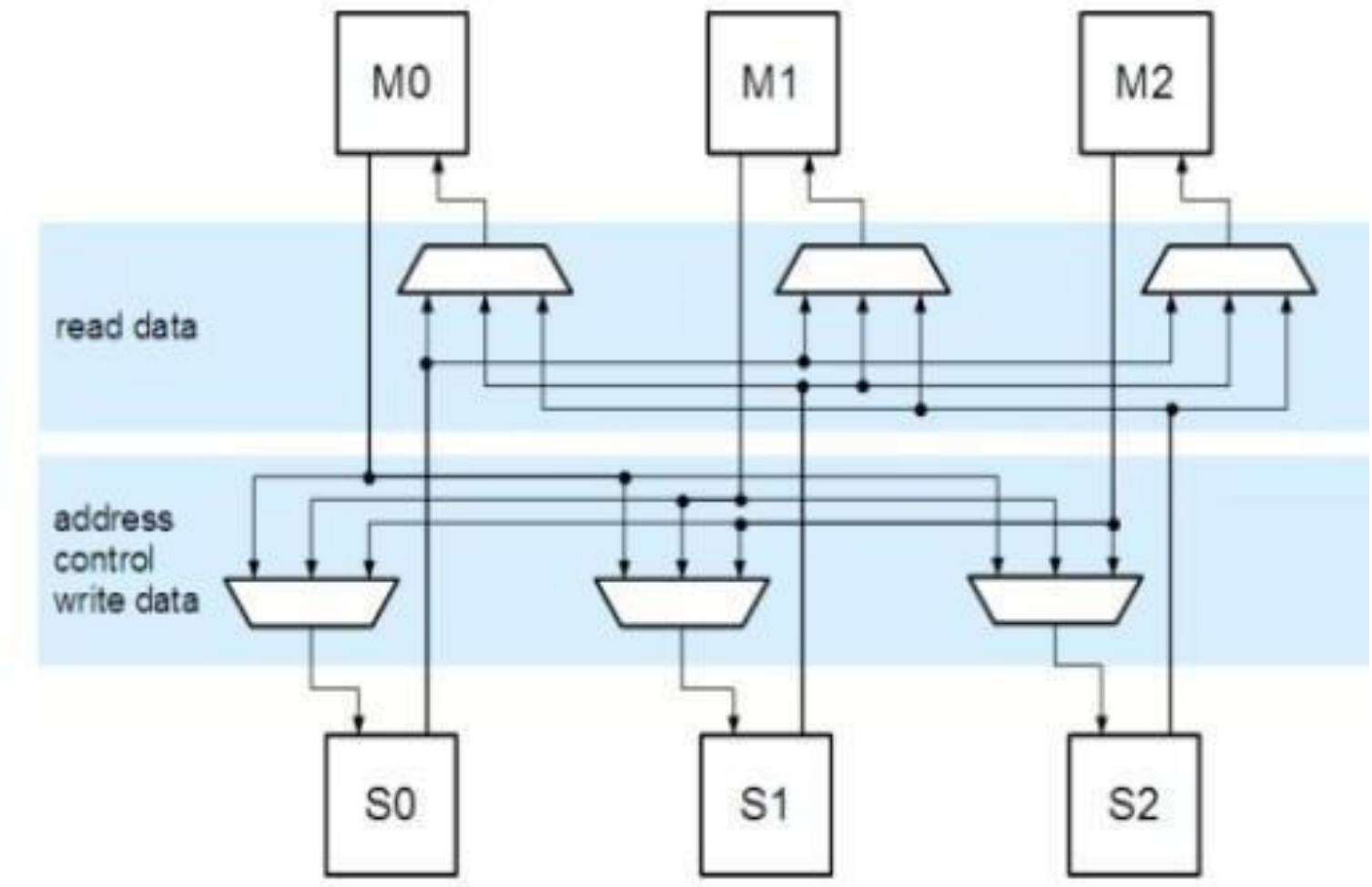


- Aufteilung des Bussystems in High- und Low-Performance Teile um Platz zu sparen
  - ➢ Verbindung der Teile mit Bus Bridge

Singlelayer Bus



Multilayer Bus



Verschiedene Protokolle

- ARM: AMBA (Advanced Microcontroller Bus Architecture)
  (AXI AHB/APB, ACE)
- Altera/Intel: Avalon
- OpenCores: Wishbone
- …

AXI Bussystem

- Advanced eXtensible Interface (AXI)
- Teil der Advanced Microcontroller Bus Architecture (AMBA) von ARM
- Spezifikation frei zugänglich und auch in der Industrie weit verbreitet
- mittlerweile in Version 5 verfügbar, jedoch 3 und 4 deutlich weiter verbreitet

Verschiedene Ausführungen von AXI4

- AXI4-Stream
  - unidirektionaler Datenstrom
- AXI4
  - memory mapped Lesen und Schreiben
- AXI4-Lite
  - AXI4 mit Einschränkungen (weniger Ressourcenverbrauch)

AXI Handshake

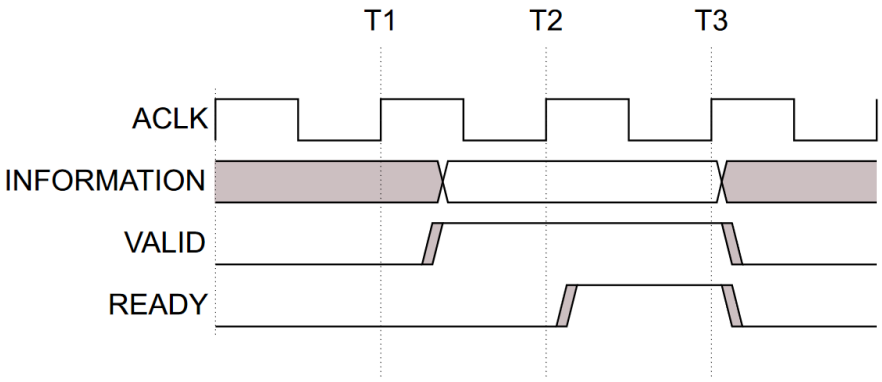


Figure A3-2 VALID before READY handshake

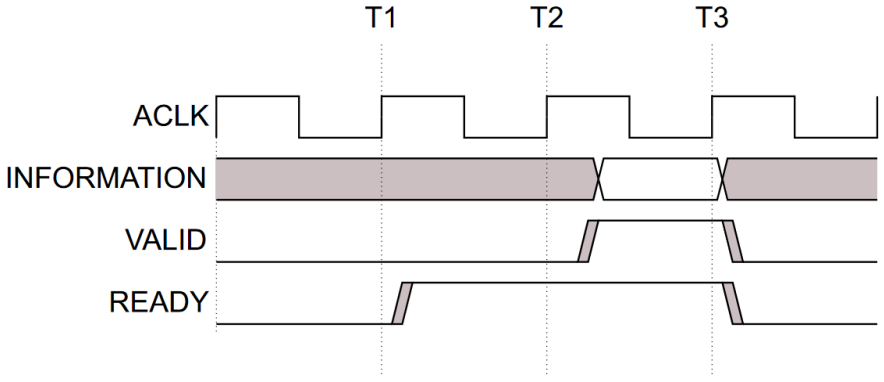


Figure A3-3 READY before VALID handshake

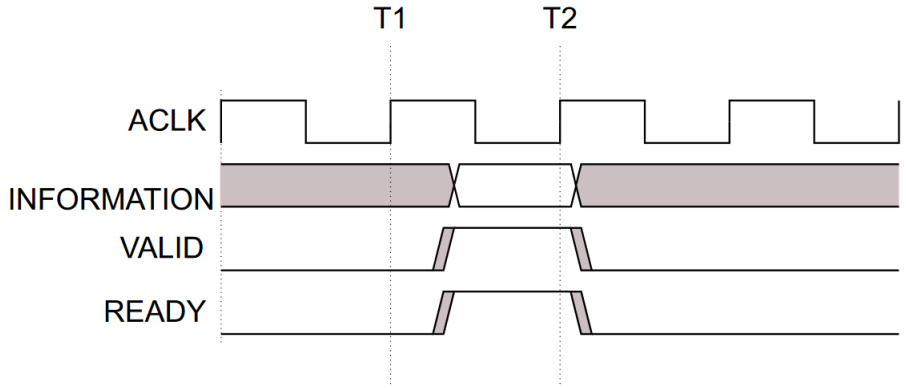


Figure A3-4 VALID with READY handshake

AXI Memory Mapped - Lesen

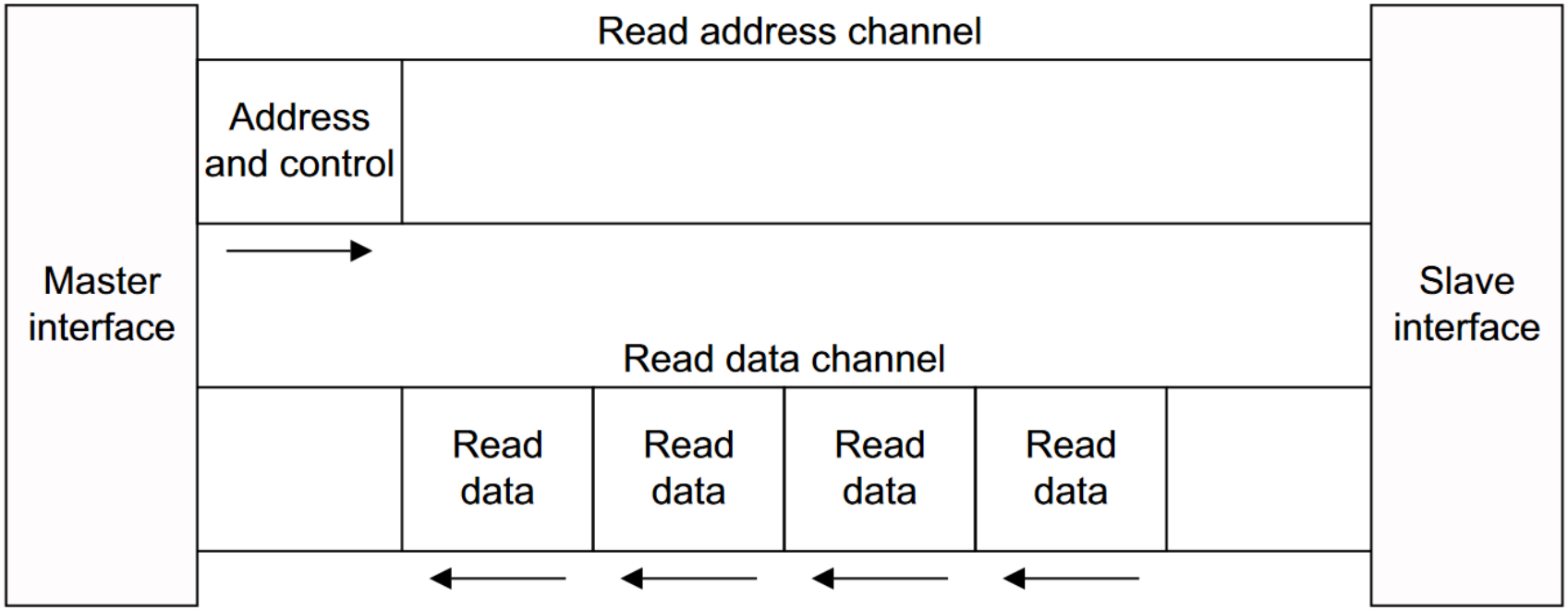


Figure A1-1 Channel architecture of reads

AXI Memory Mapped - Schreiben

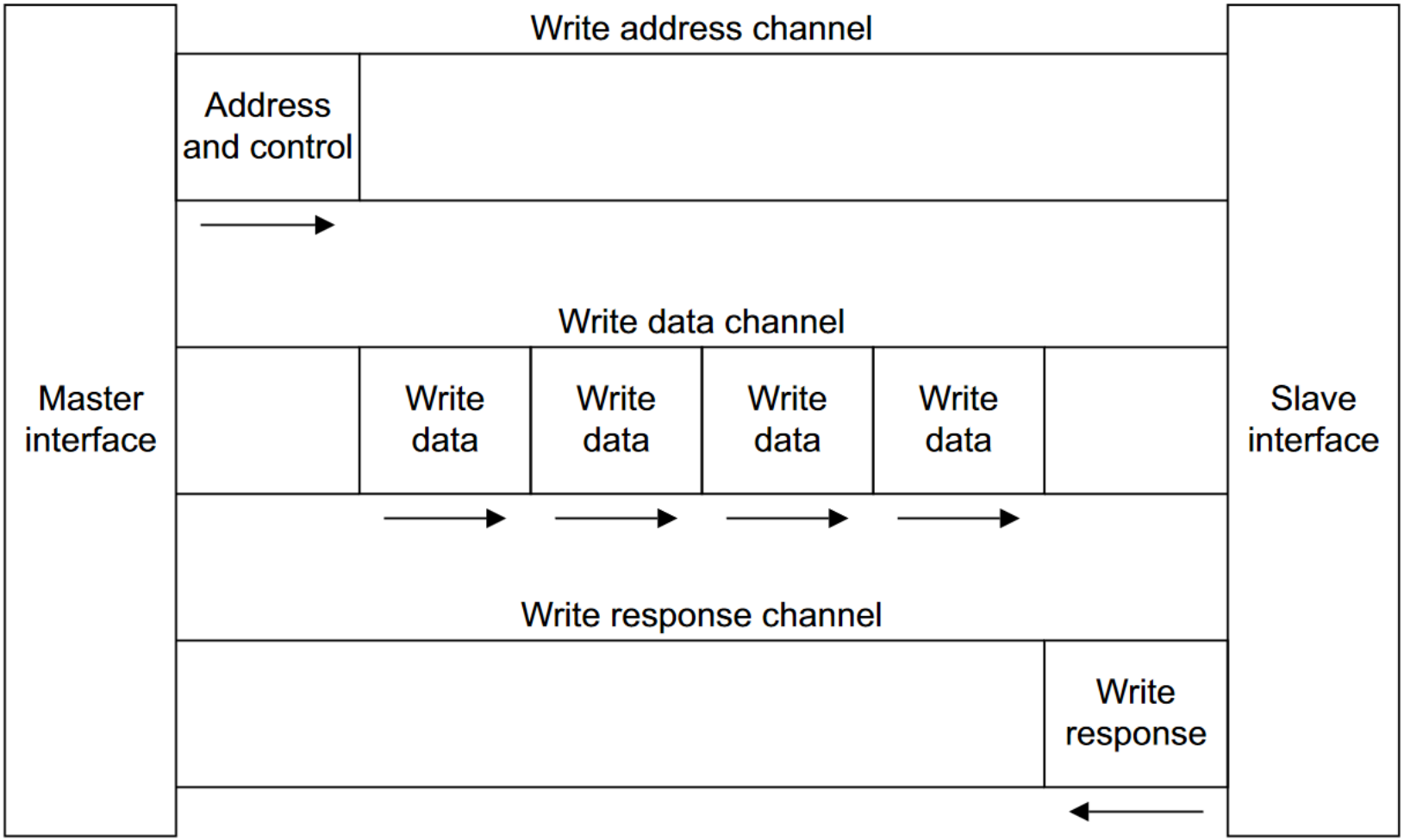


Figure A1-2 Channel architecture of writes

AXI Channel Abhängigkeiten

- Einfacher Pfeil:
  - vor oder nach dem Signal am Anfang
- Doppelter Pfeil:
  - nur nach dem Signal am Anfang

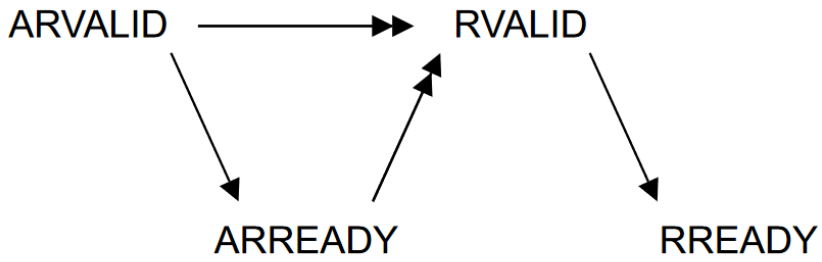
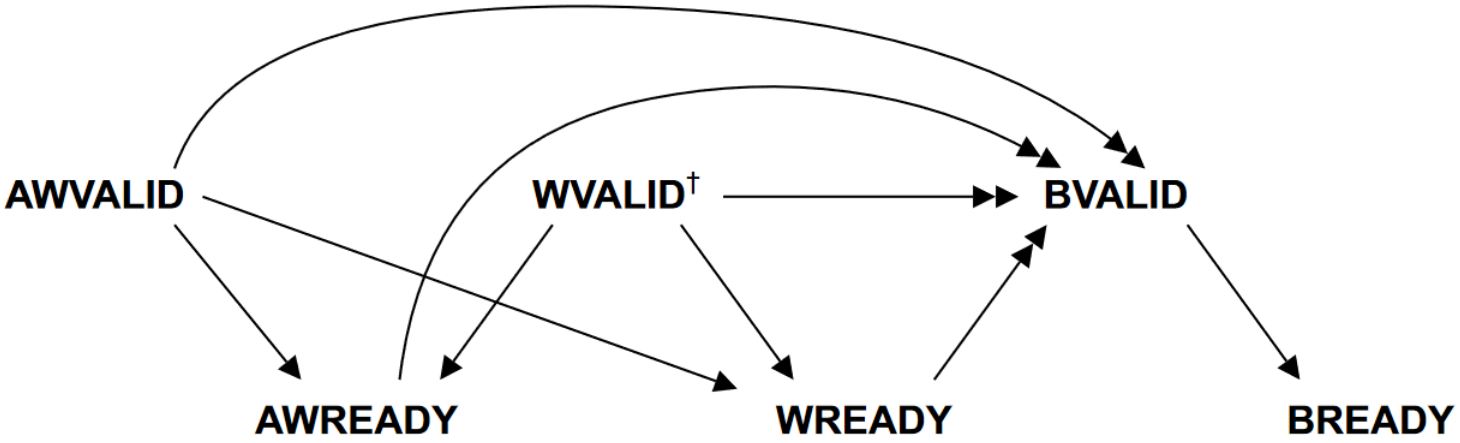


Figure A3-5 Read transaction handshake dependencies



† Dependencies on the assertion of **WVALID** also require the assertion of **WLAST**

Figure A3-7 Slave write response handshake dependencies

AXI4 Adresskontrollsignale (Auszug)

- **A*ID** - *Master address ID*
  - Bei verschiedenen IDs zweier Transactionen darf die Reihenfolge der Antworten verändert werden
- **A*LEN** - *Master Burst length*
  - Wie viele Daten sollen am Stück gelesen werden
- **A*SIZE** - *Master Burst size*
  - Wie viele Daten werden pro Handshake
- **A*BURST** - *Master Burst type*
  - FIXED, INCR, WRAP
- **A*LOCK** - *Master Lock type*
  - atomic? -> LL/SC
- **A*CACHE** - *Master Memory type*
  - Darf die Transaktion gecached oder mit anderen zusammengefasst werden?
- **A*PROT** - *Master Protection type*
  - Privilegiert, Secure, Instruktion

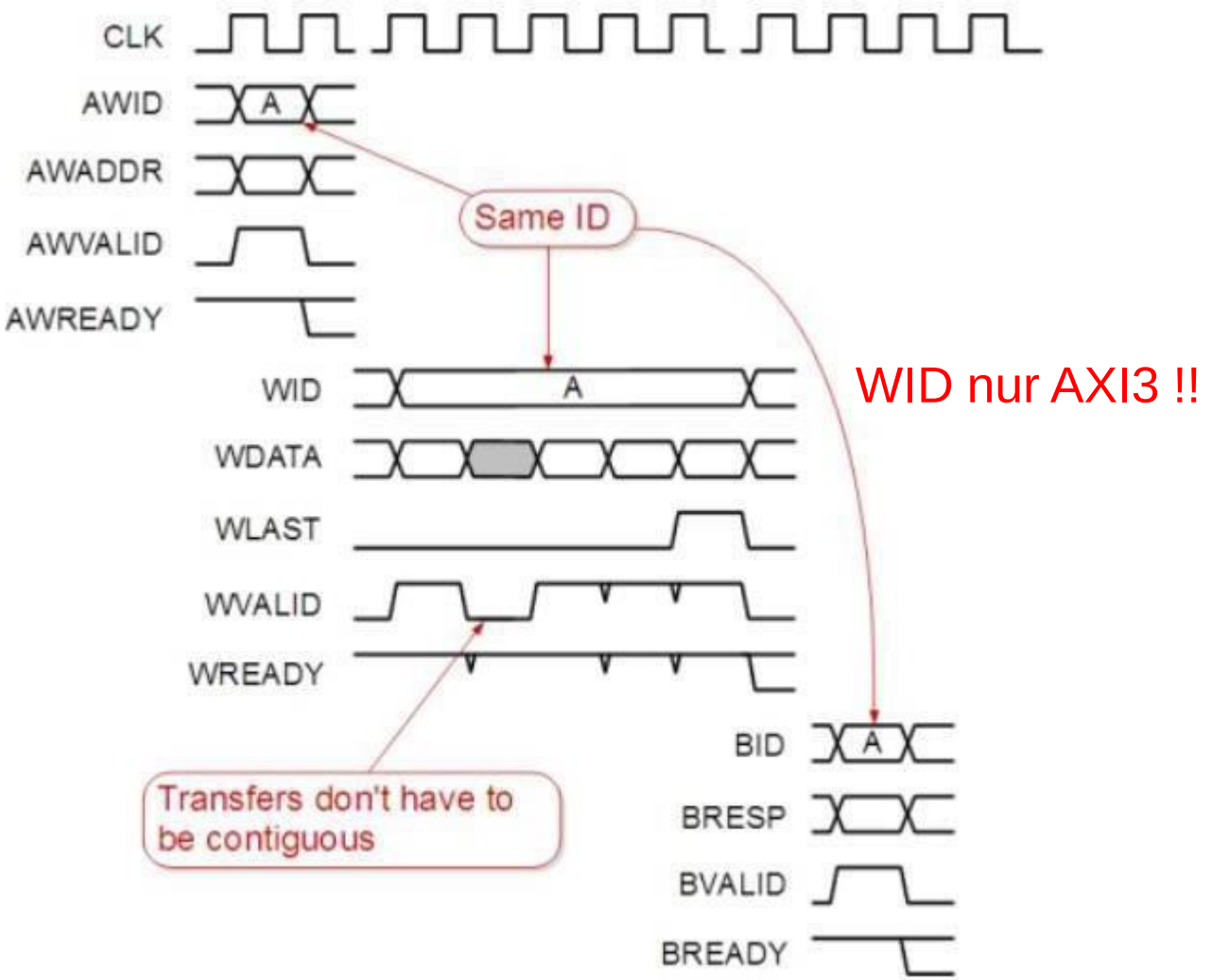
AXI4 Datenkontrollsignale

- ***LAST** - *Master (write) / Slave (read) last*
  - Das ist der letzte Transfer in einem Burst
- **WSTRB** - *Master Write strobes*
  - Ein Bit für jedes Byte im der Busbreite, das anzeigt ob es valide ist.

AXI4 Antwortkontrollsignale

- ***ID** - *Slave ID tag*
  - Zu welcher ID gehört diese Antwort?
- ***RESP** - *Slave response*
  - Ist ein Fehler aufgetreten und war ggf. der atomare Zugriff erfolgreich?

AXI IDs



WID nur AXI3 !!

Beispiele von off-chip Busprotokollen

- Hauptspeicher: DDR / LPDDR / GDDR / HBM
- Highspeed Peripherie: PCIe
- Simple Peripherie: UART / IIC / SPI
- Bildsignale: VGA / DVI / HDMI / DisplayPort
- Netzwerk: Ethernet / Infiniband

Eigenschaften von off-chip Busprotokollen

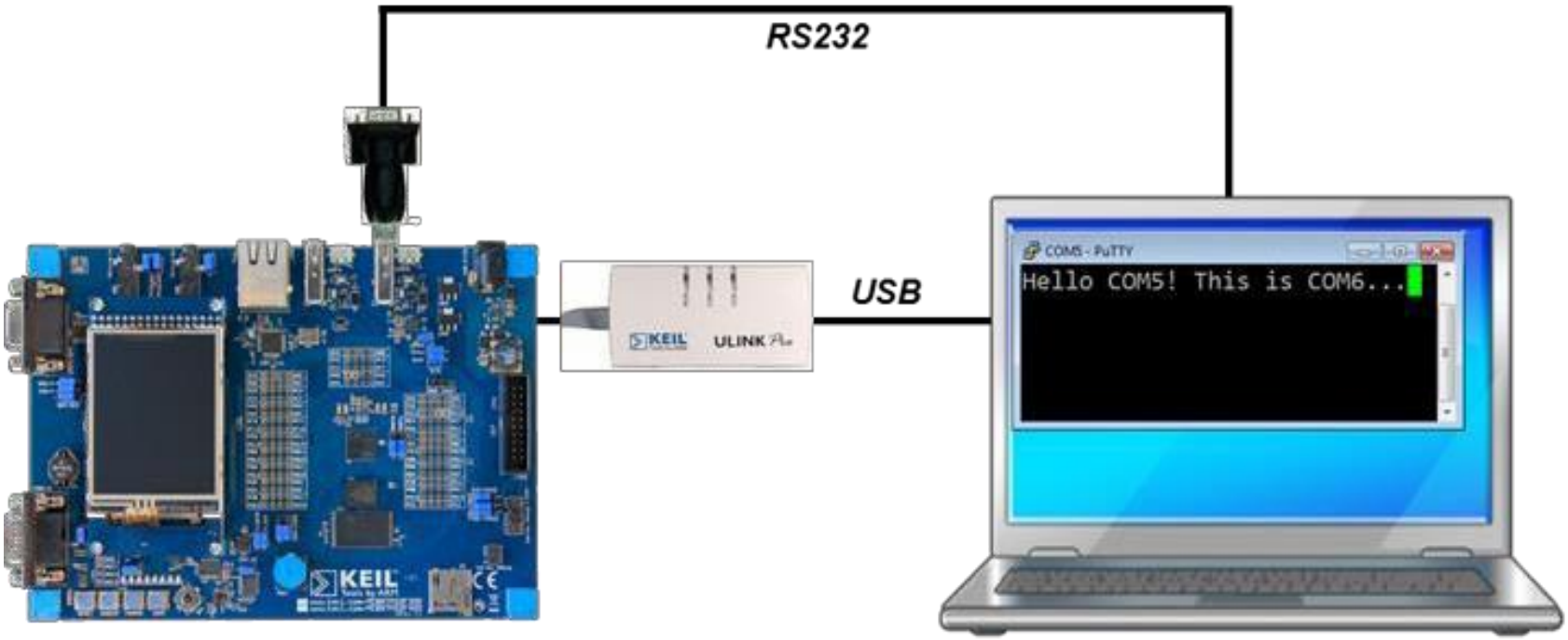
- Anzahl an Chip-I/O-Pins sehr begrenzt, da I/O-Buffer Zellen und off-chip Leitungen sehr groß sind
  - ➢ Serialisierung
    - z.B. über Multi-Gigabit-Transceiver (MGT) mit ≥ 1 Gb/s Übertragungsrate

- Verschiedene Taktsignale auf Sender- und Empfängerchip
  - ➢ Taktrückgewinnung
    - Damit Daten richtig gesampelt werden

- Off-Chip Leitungen anfälliger für Bitfehler
  - ➢ Fehlerdetektion
    - Aufteilung in Datenpakete mit Encodierung (z.B. 8b/10b) und Cyclic Redundancy Check (CRC)

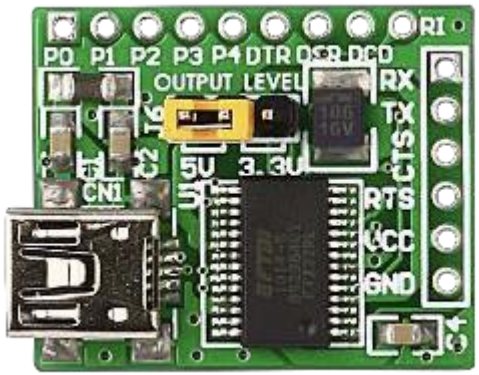
- Und vieles mehr …

UART - Anwendung

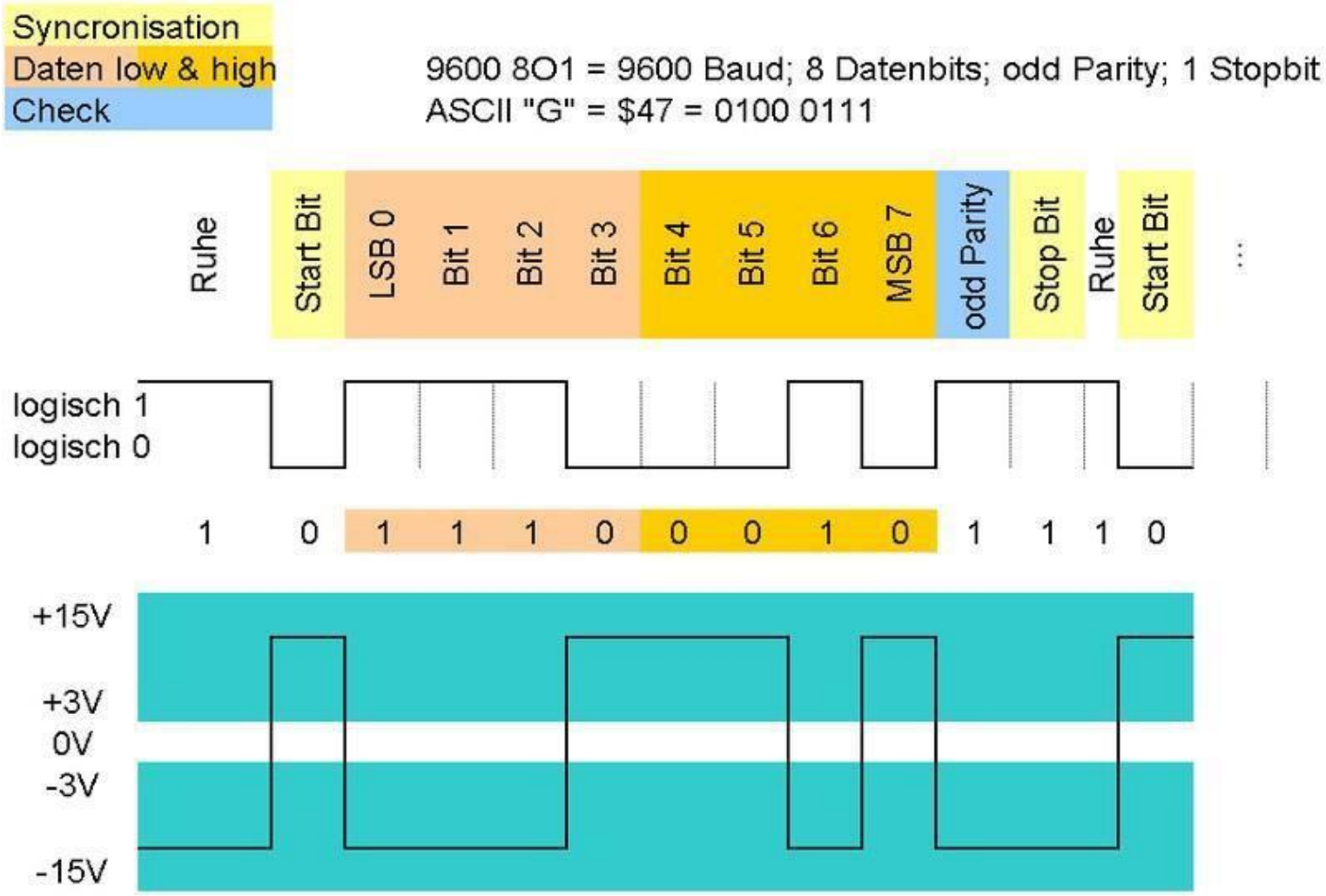


UART - Allgemein

- Realisierung einer seriellen Schnittstelle
- kein Taktsignal auf Übertragungsseite → Synchronisierung beim Empfänger (z.B. eingestellte Baudrate)
- Spezifikation frei zugänglich
- häufige Anwendung Verbindung von PC – MikroController (beispielsweise minicom, putty)

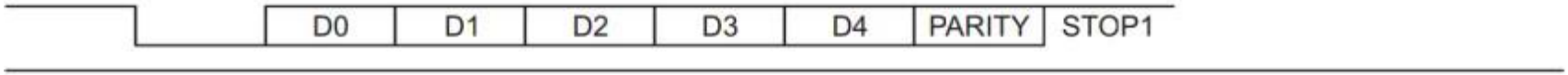


UART – Protokoll

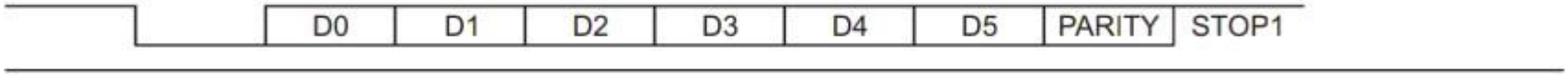


UART – Protokoll

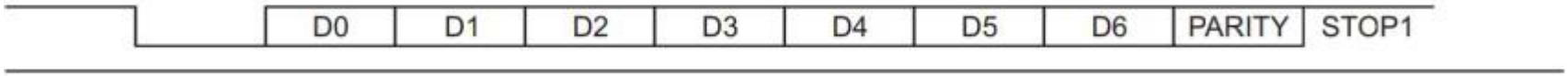
Transmit/Receive for 5-bit data, parity Enable, 1 STOP bit



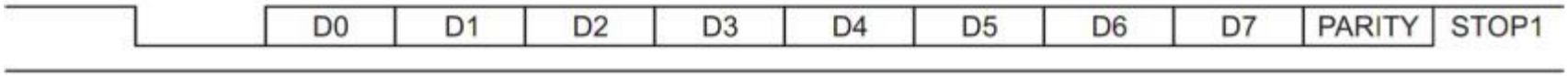
Transmit/Receive for 6-bit data, parity Enable, 1 STOP bit



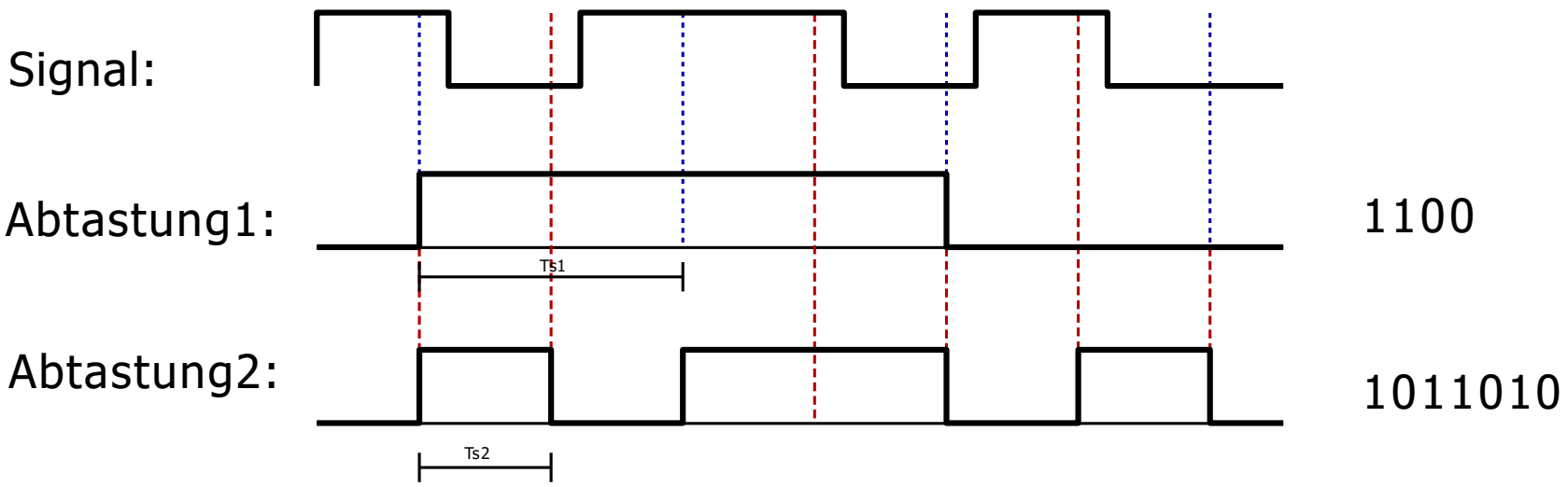
Transmit/Receive for 7-bit data, parity Enable, 1 STOP bit



Transmit/Receive for 8-bit data, parity Enable, 1 STOP bit



Schrittgeschwindigkeit / Baudrate



Schrittgeschwindigkeit = 1/Ts in Baud
Schrittgeschwindigkeit von Abtastung2 = 2 *Abtastung1

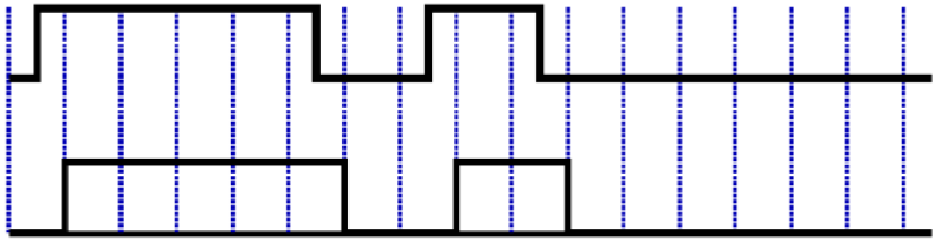
Start/Stop Bits

Orginal Signal:

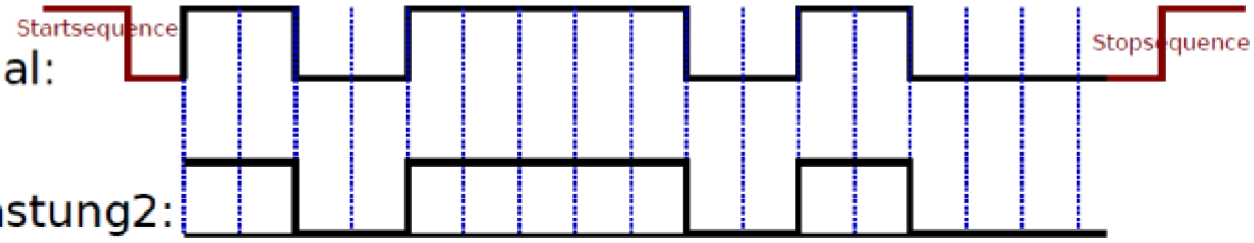
Ohne Start/Stop-Bit:

Signal:

Abtastung1:



Mit Start/Stop-Bit:



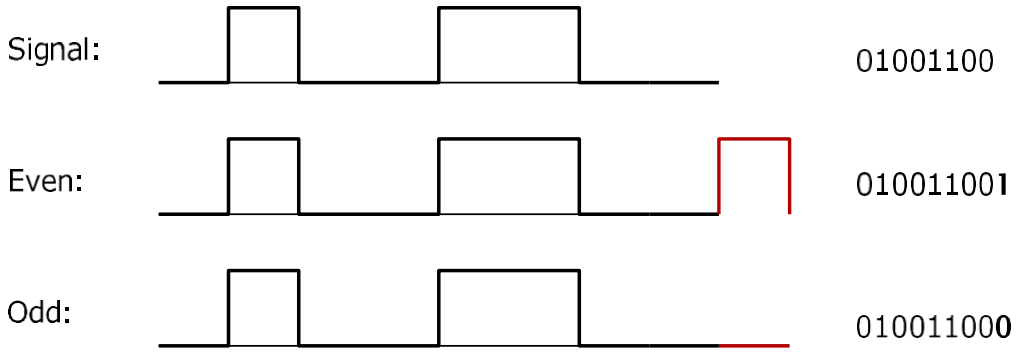
Fehlerdetektion

Fehlerdetektion bei übertragener Informationswörter

Summe der Einsen im Informationswort	even	odd
gerade	0	1
ungerade	1	0

Beispiel:

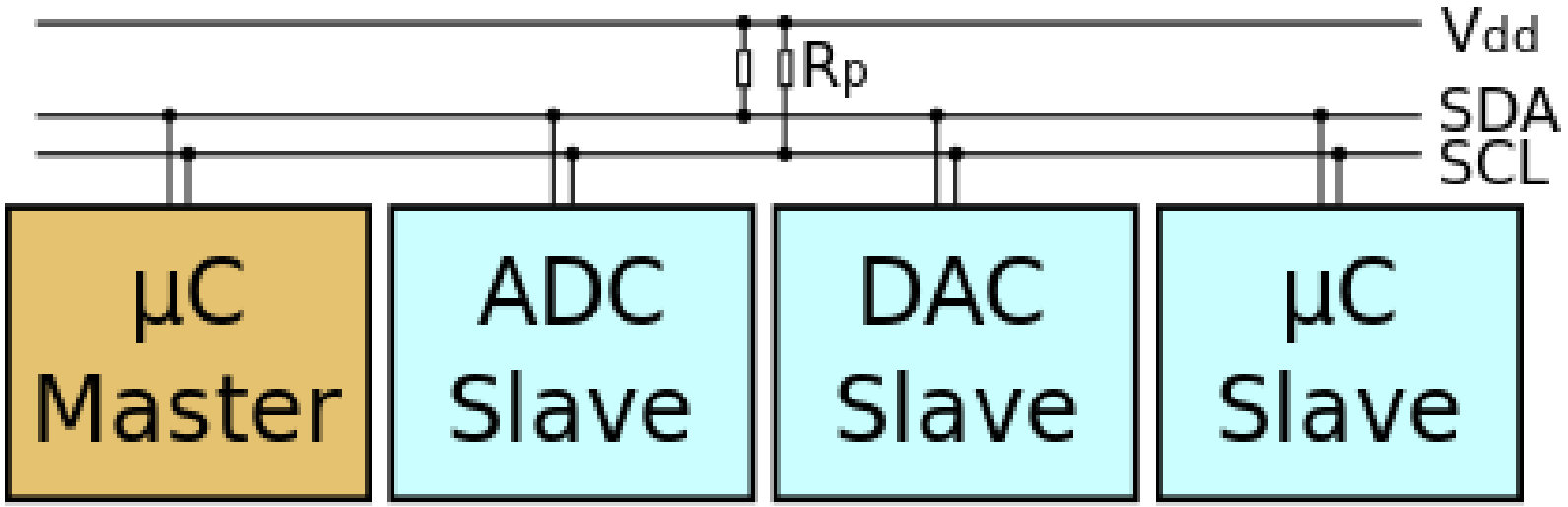
Data	=1	even	odd
0000	0	00000	00001
1011	3	10111	10110
0110	2	01100	01101
1111	4	11110	11111



IIC / I²C - Allgemein

- Serielles Protokoll zur Kommunikation zwischen i.d.R. einem Master und beliebig vielen Slaves
- Multimaster laut Spezifikation durch Arbritierung jedoch möglich
- IIC war bis 2006 lizensiert, deswegen Entwicklung TWI (Two Wire Interface)
- Nur 2 Leitungen (SDA/SCL) + Versorgungsspannung/Masse
- 0.1 – 5.0 Mbits
- Einsatz heute: Auslesen von Sensoren

Einfacher IIC Aufbau

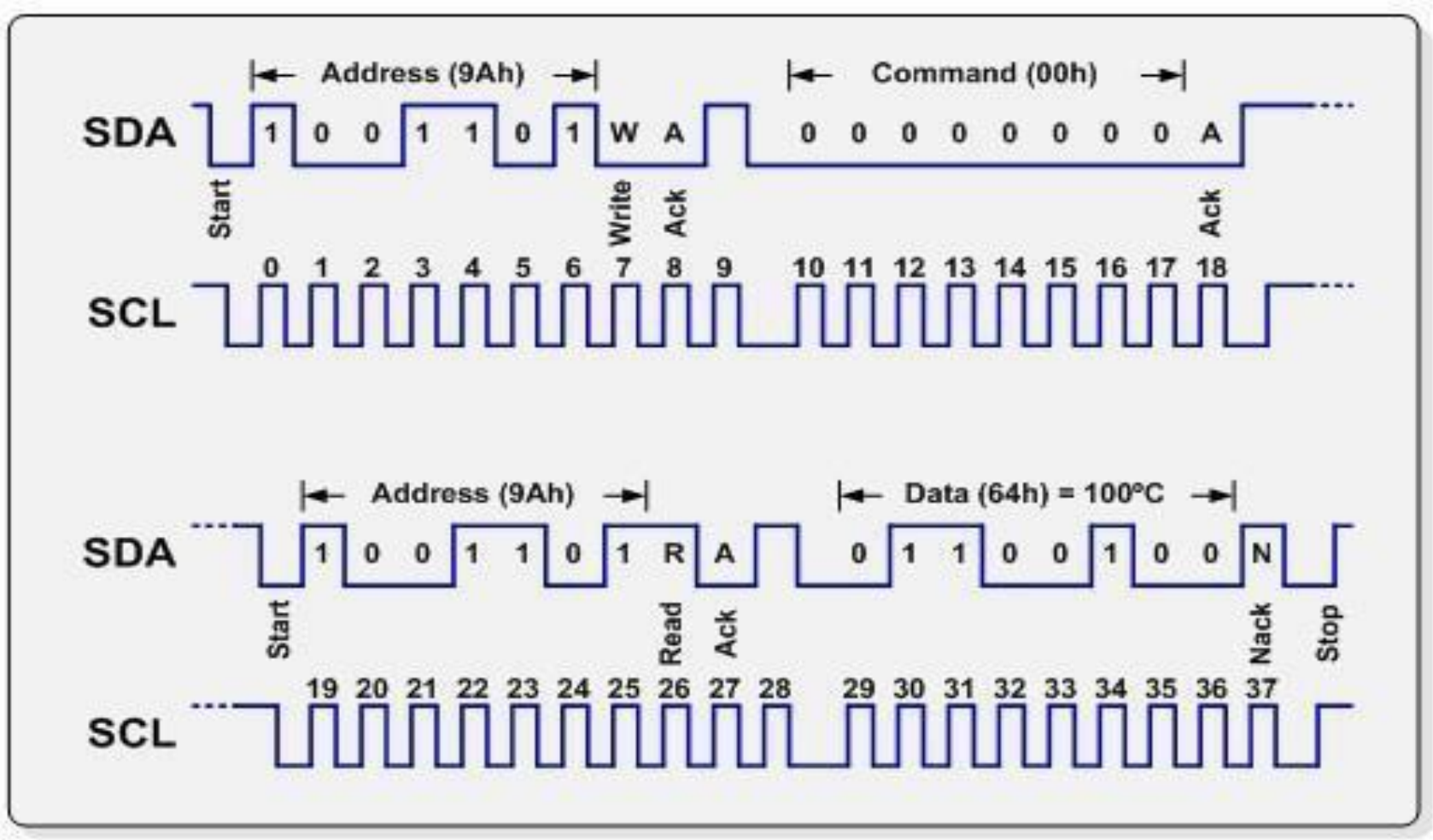


- Ein Master / 3 Slaves
- Pull-Up Widerstände

Adressierung und Datenübertragung

- Master sendet 7 Bit Device-Adresse + R/W-Bit
- Problem: Kollisionen
  - Gleiches Device auf dem Bus
  - Mehr als 128 – 16 Knoten auf dem Bus
- Übertragung durch den Master durch setzen eines Start-Signals, danach Übertragung der Adresse + R/W
- Quittieres des Slaves der Anfrage durch ACK
- Datenübertragung entweder von Master zu Slave, oder von Slave zu Master
- Ende der Daten durch NACK
- Ende der Übertragung durch Stop

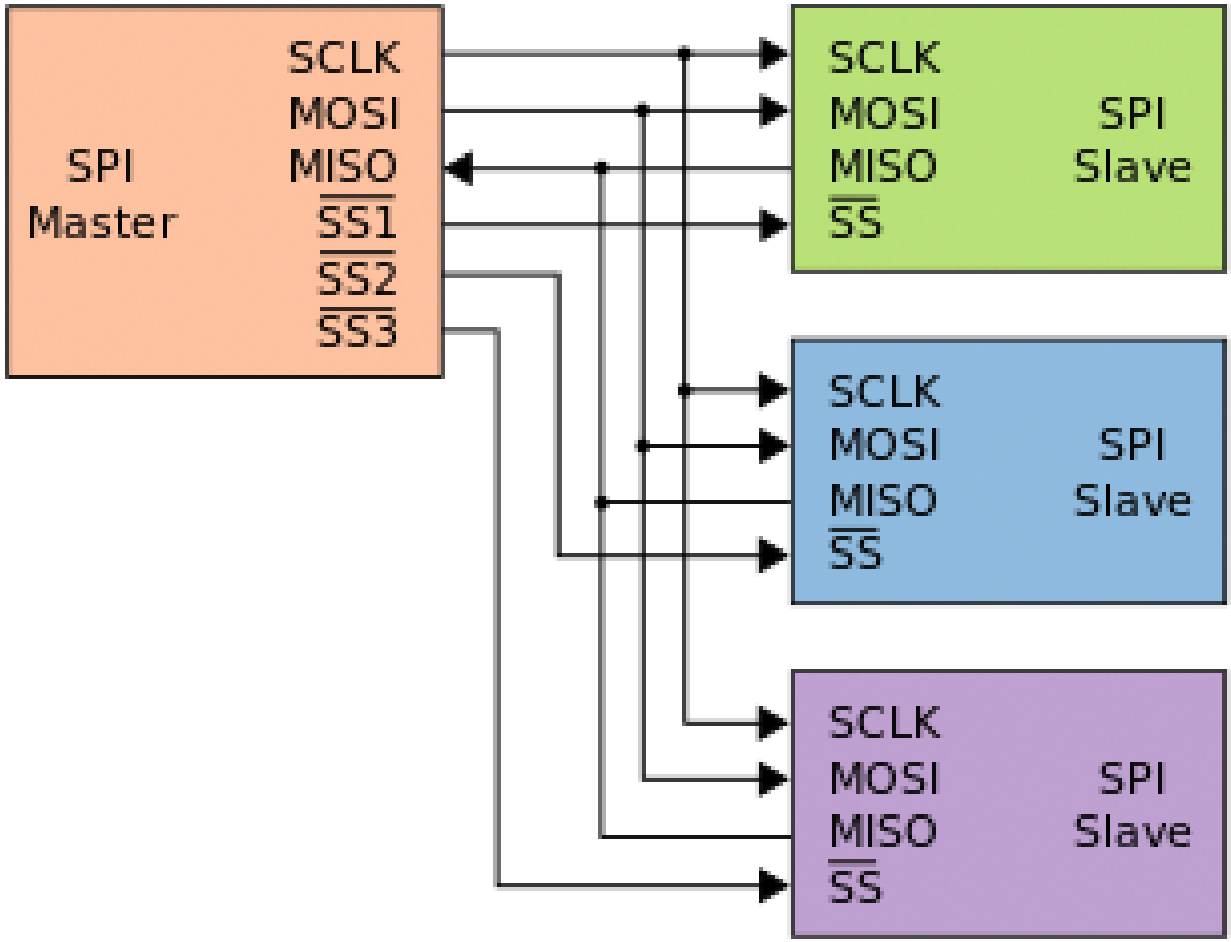
Beispielübertragung



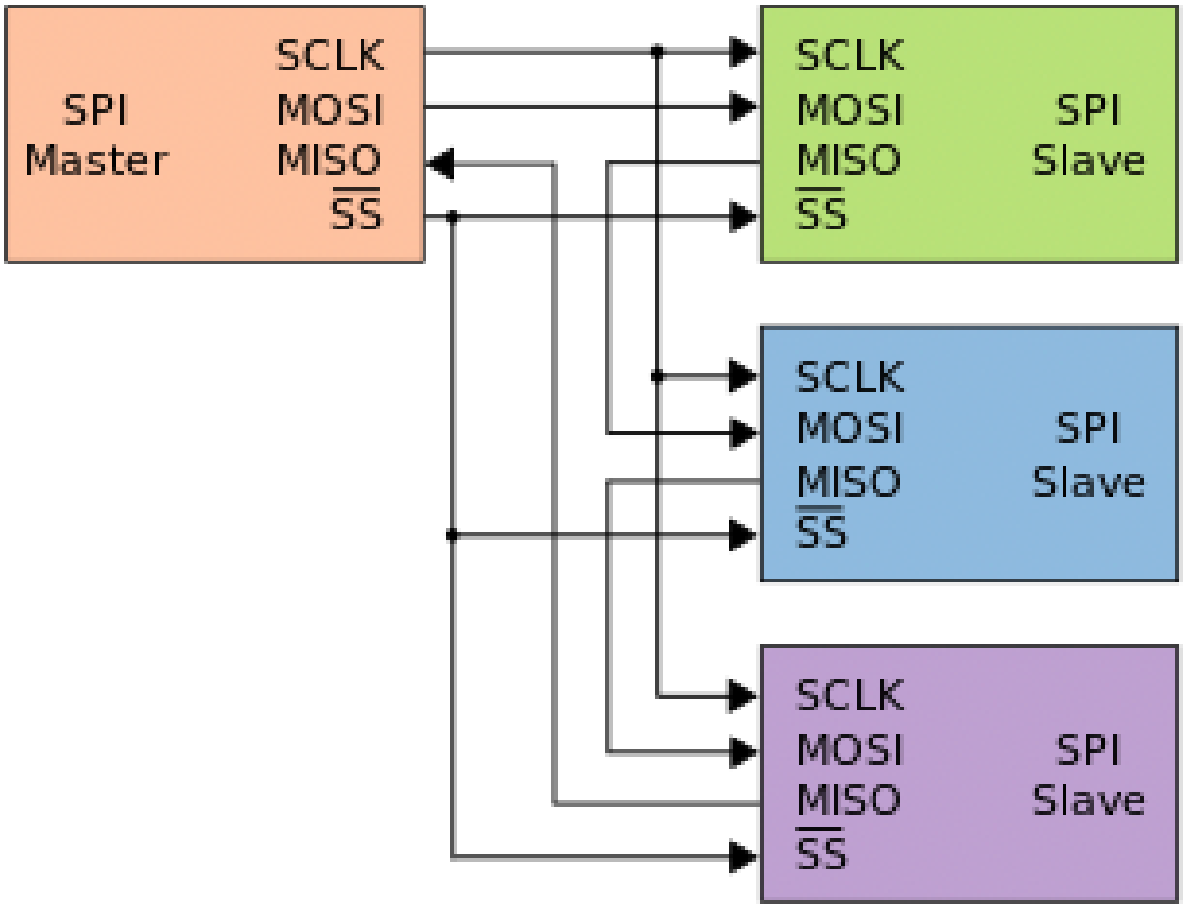
SPI - Allgemein

- Serielles Bussystem zur synchronen Datenübertragung zwischen Ics
- besteht aus 3 Leitungen:
  - MOSI (Master Out →) Slave In
  - MISO (Master In ←) Slave Out
  - SCK (Serial Clock) – Schiebetakt
- Zusätzlich zu diesen Leitungen wird für jeden Slave ein Slave Select (SS) oder Chip Select (CS) Leitung benötigt
- Master zieht SS von High auf Low um Slave auszuwählen

- SPI-Sternverbindung

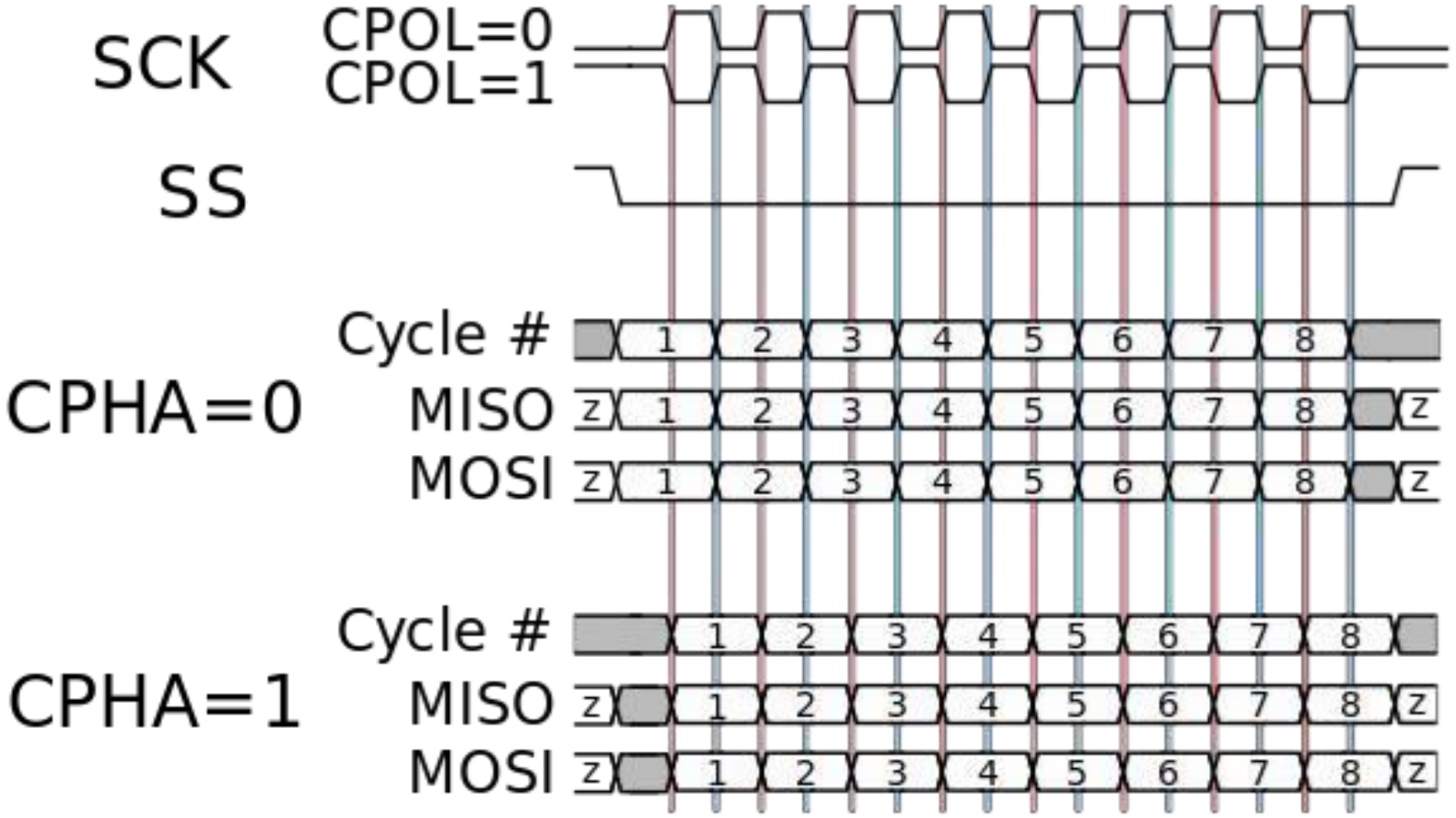


- SPI-Verbindung durch Kaskadierung der Slaves



- Kein festgelegtes Protokoll
- Clock Polarität (CPOL) und Phase (CPHA) können von Slave zu Slave unterschiedlich sein
- → verschiedene Modi möglich
- Taktfrequenz von mehreren MHz möglich
- Einsatzgebiete:
  - Schieberegister, ansteuern von LED Leisten (z.B. Rasberry Pi)
  - Zeilen LCDs, externe AD-Wandler
  - Programmierschnittstelle von Mikrocontrollern: (z.B. AVR, Erweiterung: JTAG)

- Beispiel für Übertragung



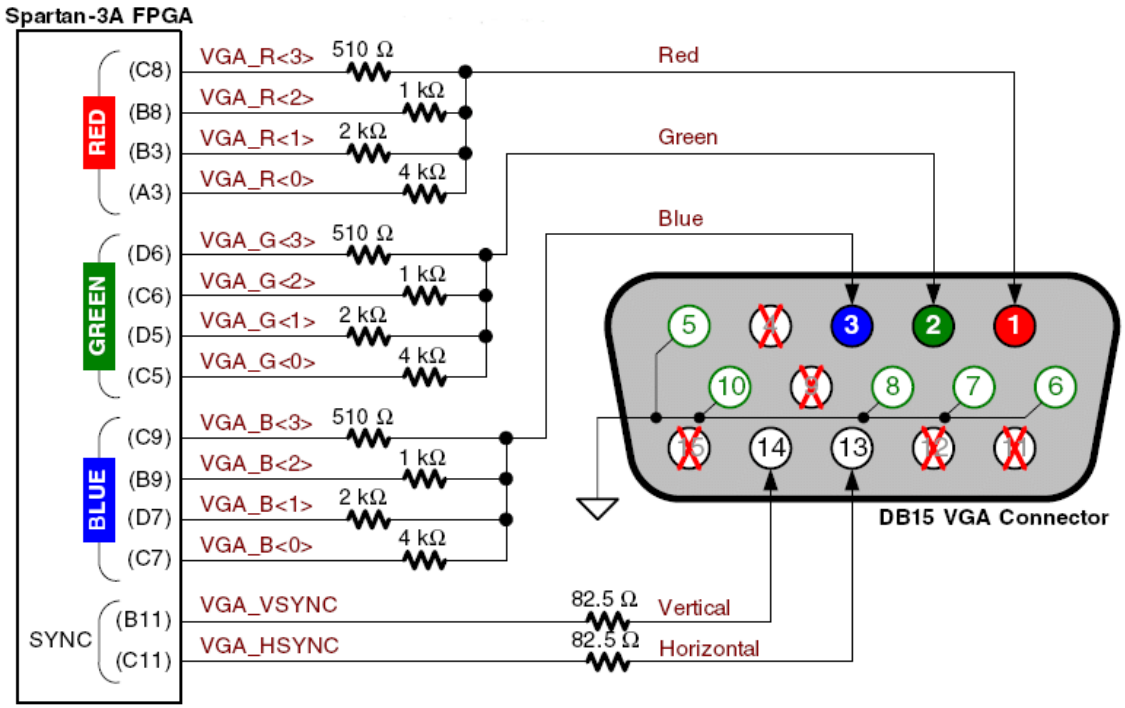
Vergleich zu anderen Bussystemen

- Vorteile:
  - Vollduplexübertragung in der Default Version
  - flexibles Protokoll (nicht nur 8 Bit)
  - simples Hardware Interface (Slave benötigt keine spez. Adresse)
  - nicht limitiert auf maximalen Takt

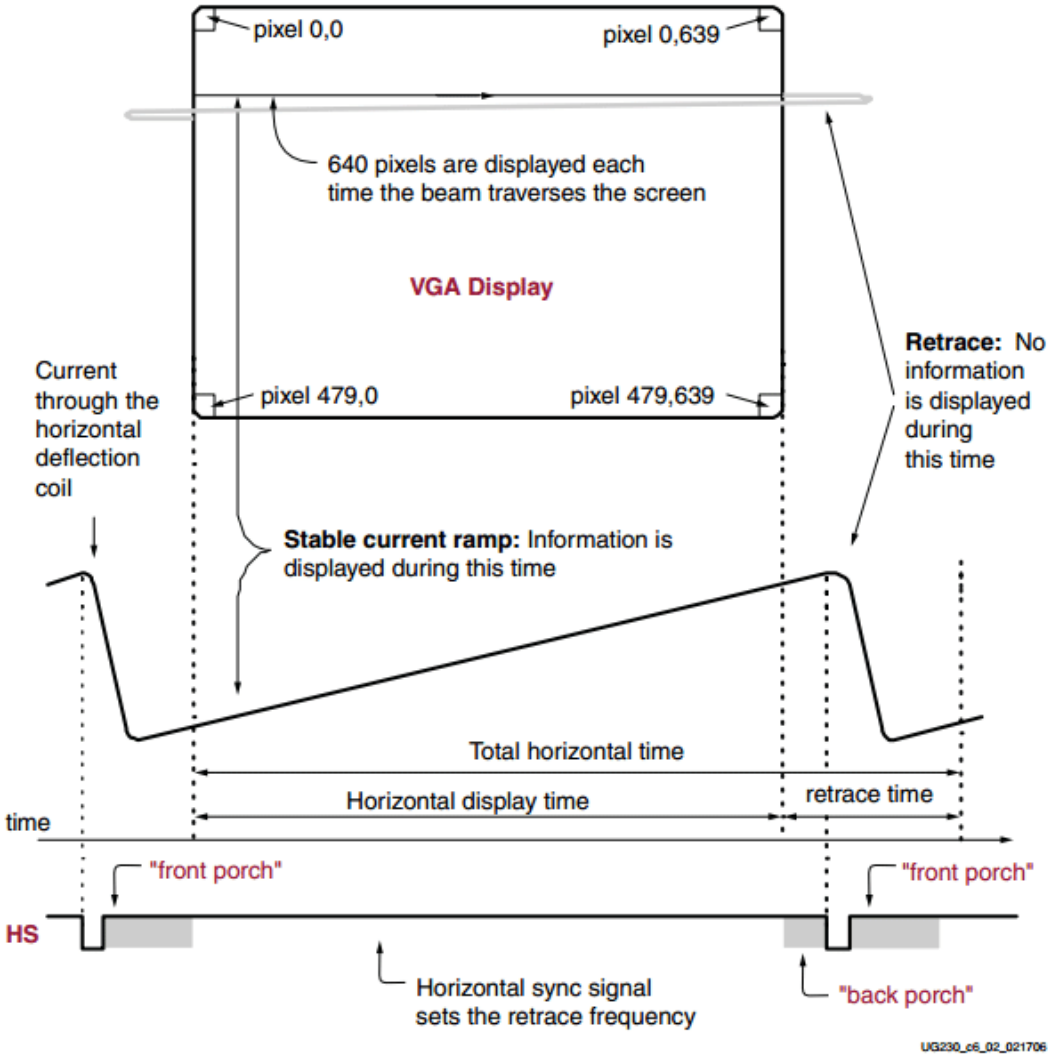
- Nachteile:
  - mehr Pins als bei I2C
  - kein Hardware/Slave Acknowledgment
  - nur kurze Distanzen möglich
  - aufwendige Integration von Interrupts

VGA – Video Graphics Array

- Standard für Computergrafik
- Veraltet, jedoch sehr oft anzutreffen, da sehr einfach
- RGB + H/V-Sync / Pixeltakt: 25.175 MHz



VGA - Timing


pixel 0,0
pixel 0,639
640 pixels are displayed each time the beam traverses the screen
VGA Display
pixel 479,0
pixel 479,639
Current through the horizontal deflection coil
Retrace: No information is displayed during this time
Stable current ramp: Information is displayed during this time
Total horizontal time
Horizontal display time
retrace time
time
"front porch"
"front porch"
HS
Horizontal sync signal sets the retrace frequency
"back porch"
UG230_c6_02_021706