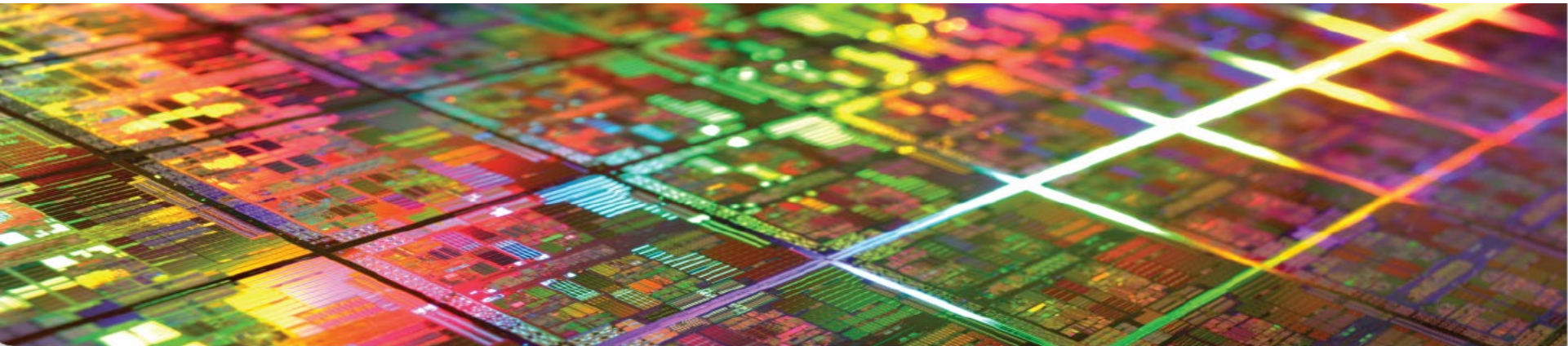


Rechnerorganisation

Prof. Dr. Wolfgang Karl

Vorlesung im Wintersemester 2025/2026 – Foliensatz: RO25-FS16-1



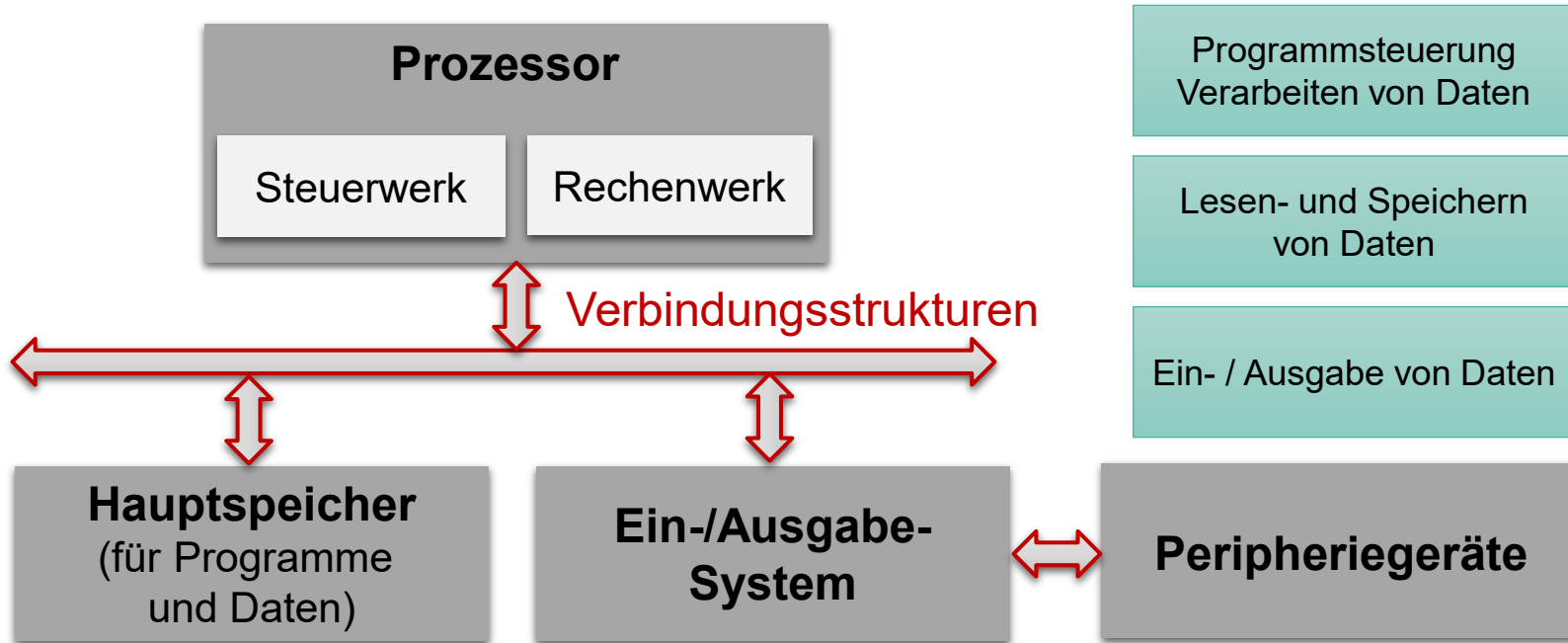
Kapitel 9

Verbindungsstrukturen

- **Einführung**
- Busse
- Punkt-zu-Punkt-Verbindungen

Programmierbarer Universalrechner

■ Einfaches Modell



Verbindungsstrukturen

■ Verbindungsstrukturen

- Das einfache Modell eines programmierbaren Universalrechners umfasst die grundlegenden Werke Prozessor, Hauptspeicher und Ein-/Ausgabe-System. Die Komponenten dieser Werke müssen miteinander kommunizieren.

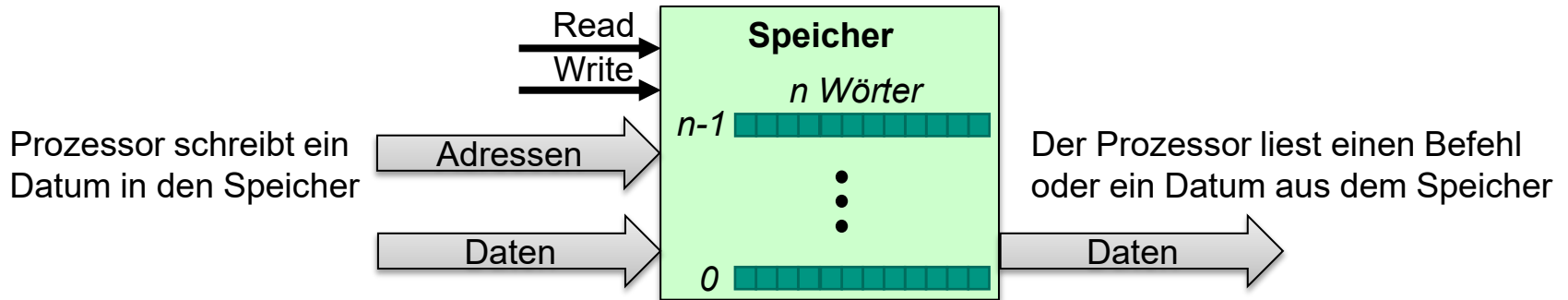
- **Verbindungsstruktur:**
 - Umfasst die Menge der Pfade, über die diese Komponenten miteinander verbunden sind und miteinander kommunizieren.
 - Der Entwurf einer solchen Struktur hängt von dem Informationsaustausch ab, der zwischen den Komponenten jeweils durchgeführt werden muss.

Verbindungsstrukturen

■ Zugriffsformen für die verschiedenen Komponententypen

■ Speicher

- Besteht aus n Wörtern gleicher Länge;
- Jedem Wort ist eine eindeutige Adresse ($0, 1 \dots, n-1$) zugeordnet;
- Durch die Angabe einer Adresse wird auf eine Speicherzelle zugegriffen und ein Datenwort kann von dort gelesen oder dorthin geschrieben werden, was durch ein Lese- oder Schreibsignal angezeigt wird.



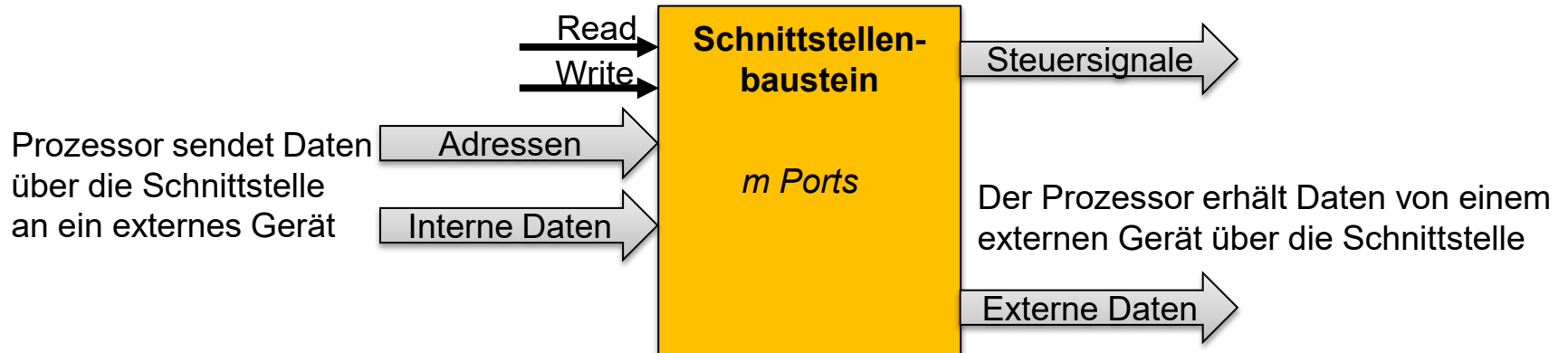
Verbindungsstrukturen

■ Zugriffsformen für die verschiedenen Komponententypen

■ Schnittstellenbaustein

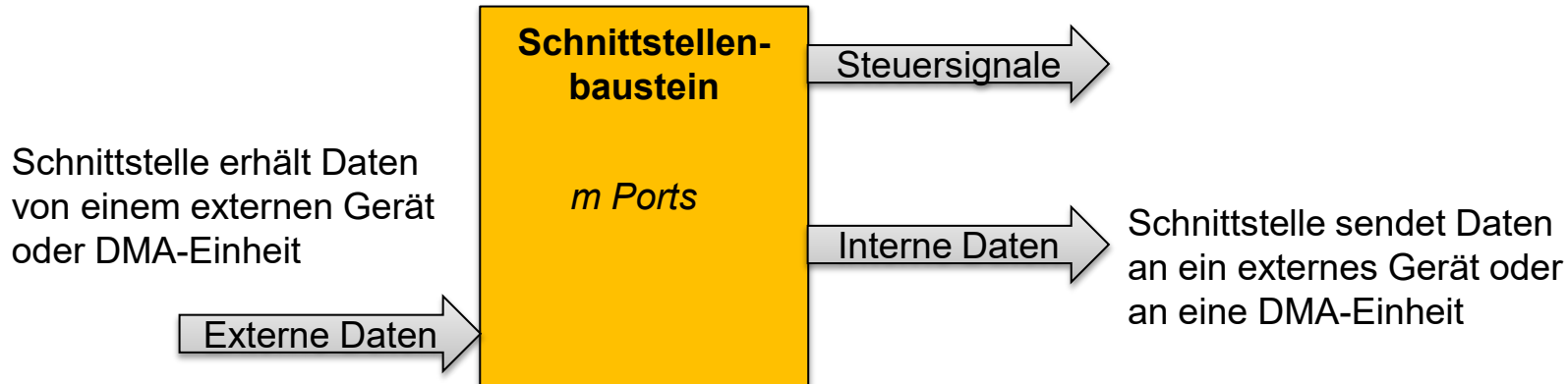
■ Aus Sicht des Systems (interne Leitungen):

- Grundlegende Operationen sind Lesen und Schreiben von Daten.
- Interne Einheiten werden über Ports adressiert.
- Jeder Port hat eine eindeutige Adresse.
- Steuersignale zur Synchronisation;



Verbindungsstrukturen

- Zugriffsformen für die verschiedenen Komponententypen
 - Schnittstellenbaustein
 - Aus Sicht des angeschlossenen peripheren Geräts (externe Leitungen)
 - Peripherer Übertragungsweg:
 - Ein- und Ausgabe von Daten;
 - Steuersignale;



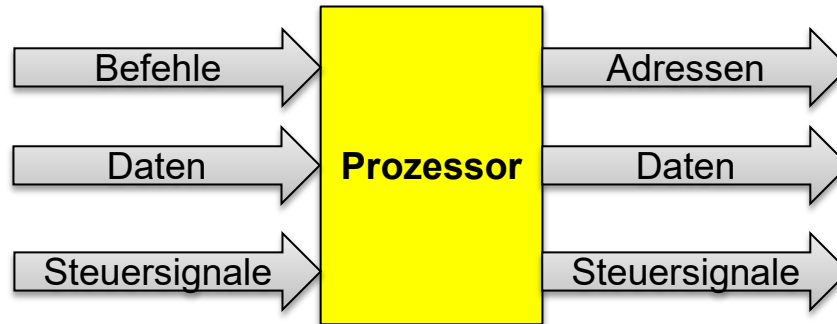
Verbindungsstrukturen

■ Zugriffsformen für die verschiedenen Komponententypen

■ Prozessor

- Lesen von Befehlen und Daten;
- Schreiben von Daten nach ihrer Verarbeitung (z. B. Ergebnisse von Berechnungen);
- Steuersignale zur Koordination der Komponenten im System;

Der Prozessor liest Befehle oder Daten vom Speicher oder erhält Daten von einem externen Gerät über die Schnittstelle



Prozessor sendet Daten an den Speicher oder über die Schnittstelle an ein externes Gerät

Verbindungsstrukturen

- **Formen**
 - **Busse-Systeme**
 - **Punkt-zu-Punkt-Verbindungsstrukturen**

Kapitel 9

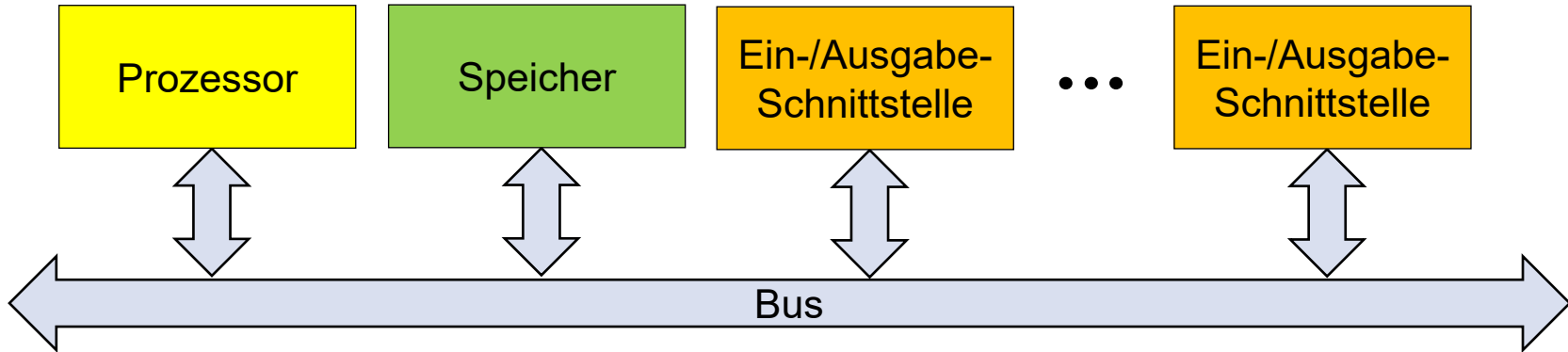
Verbindungsstrukturen

- Einführung
- **Busse**
- Punkt-zu-Punkt-Verbindungen

Verbindungsstrukturen

■ Bus-System

- Kommunikationspfad für den Anschluss von zwei oder mehr Komponenten;



- Gemeinsames Übertragungsmedium für die am Bus angeschlossenen Komponenten;
- Wenn ein Signal von einer Komponente übertragen wird, kann dies von allen am Bus angeschlossenen Komponenten empfangen werden;
- Nur eine Komponente darf zu einem Zeitpunkt ein Signal auf einer Leitung übertragen;

Bus-Systeme

■ Datenübertragung zwischen aktiver Komponente und passiver Komponente

■ Aktive Komponente:

- Bussteuernde Einheit (z. B.: Prozessor, DMA-Einheit),
- Initiiert und steuert den Übertragungsablauf;
- Adressiert die passive Komponente;
- Verwaltet die für die Übertragung erforderlichen Steuersignale;

■ Passive Komponente:

- passive Einheit (z. B.: Speicher, Interface);
- Kann nur auf Anfragen antworten;

Anmerkung: aktive Komponenten sind früher als *Master*, passive als *Slave* bezeichnet worden

Bus-Systeme

■ Datenübertragung zwischen aktiver Komponente und passiver Komponente

■ Buszyklus: Protokoll des Datentransfers

- Die aktive Komponente erhält die Kontrolle über den Bus.
- Die aktive Komponente adressiert die am Datentransport beteiligte passive Einheit.
- Angabe der Transportrichtung (Lesen oder Schreiben)
- Bereitstellung oder Übernahme der Daten
- Die Synchronisation der Aktivitäten der aktiven Komponente und der am Datentransport beteiligten Komponente erfolgt:
 - zu fest vorgegebenen Zeitpunkten, die am gemeinsamen Taktsignal ausgerichtet sind.
 - Übertragung unterliegt einem festen Zeitraster;
 - über den Austausch von Steuersignalen;

Bus-Systeme

- **Datenübertragung zwischen aktiver Komponente und passiver Komponente**
 - **Buszyklus: Interrupt-Verwaltung**
 - IRQ- und IACK-Signale;
 - Bus-Protokoll für die Übertragung der für eine Interrupt-Behandlung notwendigen Informationen;
 - Z. B. Vektor-Nummer

Bus-Systeme

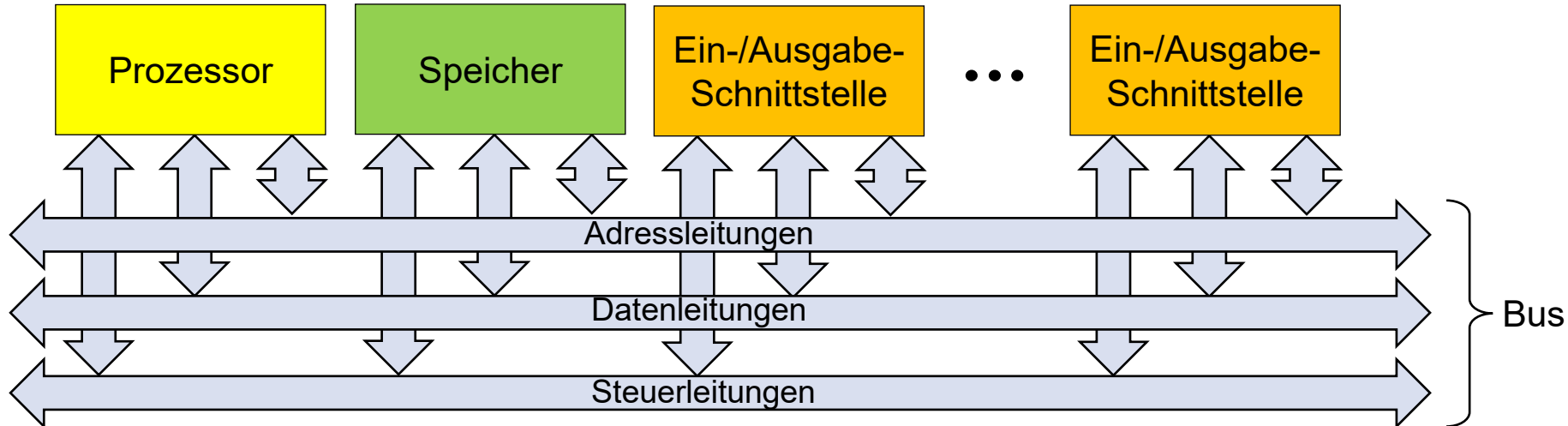
■ Busarbitrierung (Buszuteilung)

- Bei mehreren aktiven Komponenten notwendig;
- Gewährleistet, dass zu einem Zeitpunkt nur eine aktive Komponente die Kontrolle über den Bus besitzt;
- Priorisierung der zu einem Zeitpunkt von mehreren aktiven Komponenten anliegenden Anforderungen für eine Buszuteilung;
- Verwaltung der Steuersignale für die Buszuteilung an eine ausgewählte aktive Komponente;
- Zentraler oder dezentraler Busverwalter (Arbiter);

Bus-Systeme

■ Zusammenfassung von Leitungen zu logischen Übertragungseinheiten

- Adressbus
- Datenbus
- Steuerbus



Bus-Systeme

■ Zusammenfassung von Leitungen zu logischen Übertragungseinheiten

■ Adressbus

■ Übertragung von Adressinformationen:

- Diese bestimmen die Quelle oder das Ziel der auf dem Datenbus zu übertragene Daten;

- Die über den Adressbus adressierbaren Module können

 - Speicherkomponenten oder

 - Ein-/Ausgabe-Einheiten (I/O Ports) sein.

- Unterscheidung zwischen Isolierter Adressierung und speicherbezogene Adressierung:

 - Die Breite des Adressbusses (Anzahl der Leitungen) bestimmt die maximal mögliche Speicherkapazität des Systems (isolated I/O).

 - Der höherwertige Anteil der Adresse kann verwendet werden, ein am Bus angeschlossenes Modul auszuwählen. Die niederwertigen Bits wählen Speicherelement (memory-mapped I/O)

Bus-Systeme

■ Zusammenfassung von Leitungen zu logischen Übertragungseinheiten

■ Datenbus

- Übertragung von Daten zwischen den am Bus angeschlossenen Komponenten;
- Anzahl der Datenleitungen (z.B. 32, 64, 128, oder mehr) bestimmt die Breite des Datenbusses;
 - Anzahl der Leitungen bestimmt, wie viele Bits pro Zeiteinheit übertragen werden können;

Bus-Systeme

- Zusammenfassung von Leitungen zu logischen Übertragungseinheiten
 - Steuerbus:
 - Übertragung von Steuerinformationen
 - Beispiele:
 - **Memory write**: bewirkt, dass die Daten auf dem Datenbus in die adressierte Stelle geschrieben werden;
 - **Memory read**: bewirkt, dass die Daten der adressierten Stelle auf den Datenbus ausgegeben werden;
 - Bei isolierter Adressierung M/\overline{IO}
 - **out** bewirkt, dass die Daten über den adressierten I/O Port ausgegeben werden;
 - **in** bewirkt, dass die Daten vom adressierten I/O Port auf den Bus gelegt werden;

Bus-Systeme

- **Zusammenfassung von Leitungen zu logischen Übertragungseinheiten**
 - **Steuerbus:**
 - Übertragung von Steuerinformationen
 - **Beispiele:**
 - **Bus request:** zeigt an, dass eine aktive Komponente auf den Bus zugreifen möchte;
 - **Bus grant:** zeigt an, dass die einen Buszugriff anfordernde aktive Komponente die Kontrolle über den Bus erhält;
 - **Interrupt request:** zeigt an, dass eine Interrupt-Anforderung anliegt;
 - **Interrupt ACK:** zeigt an, dass einer Interrupt-Anforderung stattgegeben wird;

Bus-Systeme

- **Zusammenfassung von Leitungen zu logischen Übertragungseinheiten**
 - **Steuerbus:**
 - Übertragung von Steuerinformationen
 - **Beispiele:**
 - **Versorgungsleitungen**
 - Stromversorgung
 - **Clock:** Takt-Signal zur Synchronisation der Operationen auf dem Bus;
 - **Reset:** Zurücksetzen und Initialisieren aller Komponenten;

Bus-Systeme

- **Zusammenfassung von Leitungen zu logischen Übertragungseinheiten**
 - **Zustände von Steuerleitung**
 - **Aktiver Zustand:** Signal ist gesetzt, ihre Funktion ist wirksam;
 - **1-aktive Signale** haben, wenn sie im aktiven Zustand sind, den logischen Wert 1;
 - **0-aktive Signale (Signalname überstrichen, vorgestelltes ‚n‘, nachgestellter *)**; haben, wenn sie im aktiven Zustand sind, den logischen Wert 0
 - **Inaktiver Zustand:** Signal ist nicht gesetzt, ihre Funktion ist ohne Wirkung
 - Einige Signale (z.B. das $\overline{R/W}$ -Signal) sind sowohl mit dem Wert 1 (Read) als auch mit dem Wert 0 (Write) im aktiven Zustand.
 - Bei diesen Signalen stellt der Wert 1 den Aktivzustand des nicht überstrichenen und der Wert 0 den Aktivzustand des überstrichenen Teils dar.

Bus-Systeme

- **Zusammenfassung von Leitungen zu logischen Übertragungseinheiten**
 - **Zustände von Steuerleitung**
 - **Tri-State-Logik**
 - Bei aktiven Komponenten sind die meisten Signalausgänge mit einer Tri-State-Logik versehen.
 - Ein Signal kann neben den logischen Zuständen 0 und 1 einen hochohmigen Zustand (high-impedence state) annehmen.
 - Die Tri-State-Logik ermöglicht es dem Prozessor, seine Signale vom Bus abzukoppeln, so dass eine andere aktive Komponente den Bus übernehmen kann.
 - **Open-Collector-Verhalten (Open-Drain) von Signalausgängen**
 - Open-Collector-Ausgänge bilden, wenn sie miteinander verbunden (verdrahtet) werden:
 - bei 1-aktiven Signalen eine UND-Verknüpfung (verdrahtetes UND) und
 - bei 0-aktiven Signalen eine ODER-Verknüpfung (verdrahtetes ODER, wired-or).

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

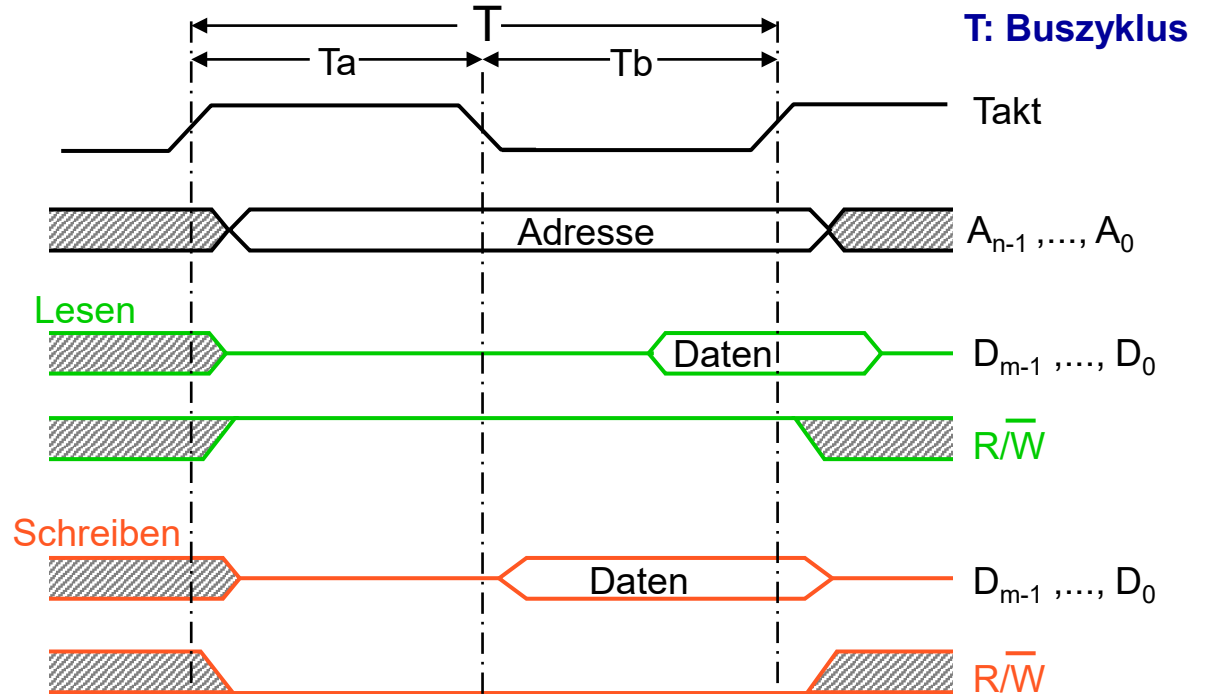
■ Synchroner Bus

- Die aktive und die passive Komponente werden durch ein gemeinsames Bustaktsignal miteinander synchronisiert, d.h. die Übertragung unterliegt einem festen Zeitraster.
- Der Ablauf wird von der aktiven Komponente durch ein Startsignal ausgelöst, das der passiven Komponente den Beginn des Buszyklus anzeigt.
- Danach gibt es feste, durch den Bus-Takt vorgegebene Zeitpunkte für
 - die Gültigkeit der von der aktiven Komponente ausgegebenen Adresse,
 - die Gültigkeit des vom Sender ausgegebenen Datums,
 - die Übernahme des Datums durch den Empfänger, d. h. den Abschluss des Buszyklus.

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Synchroner Bus



Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Synchroner Bus

- Adresse wird zu Beginn des Buszyklus (T_a) auf den Adressbus gelegt
- Anzeige der Übertragungsrichtung durch R/\overline{W}

■ Lesen ($R/\overline{W} = 1$):

- Der Speicher (oder eine andere Systemkomponente) liefert die Daten gegen Ende des Buszyklus (T_b)
- Übernahme der Daten in den Prozessor mit der steigenden Flanke des Bustakts.

■ Schreiben ($R/\overline{W} = 0$):

- Prozessor legt die Daten (spätestens) zu Beginn der zweiten Takthälfte (T_b) auf den Bus.
- Übernahme der Daten in den Speicher mit der steigenden Flanke des Bustakts am Ende der zweiten Takthälfte (T_b)

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Synchroner Bus

■ Nachteil:

- Alle am Bus angeschlossenen Komponenten müssen die durch den Bustakt vorgegebenen strengen Zeitvorgaben erfüllen.
- Die langsamste Komponente bestimmt die zulässige Geschwindigkeit des Busses.
- Oder: der Bus schließt den Einsatz von „schnellen“ Komponenten aus (z.B. schnelle Speicher)

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

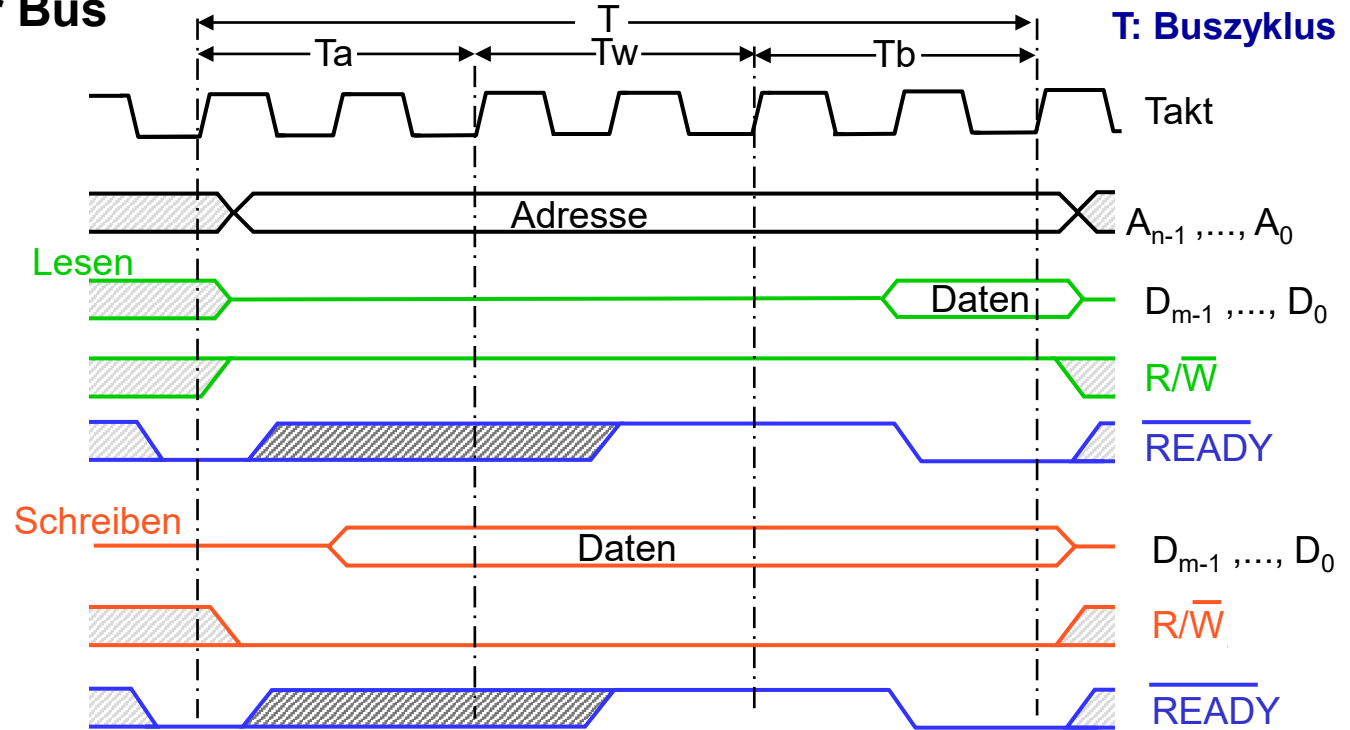
■ Semi-synchroner Bus

- Erweiterung des Protokolls gegenüber dem synchronen Bus um ein Bereitschaftssignal READY;
 - Wird von der passiven Komponente gesetzt und ist ebenfalls mit dem Bustakt synchronisiert;
 - Zeigt der aktiven Komponente die Bereitstellung oder Übernahme der Daten an;
 - Wird von der passiven Komponente über den kürzest möglichen Buszyklus hinaus inaktiv gehalten, wodurch die aktive Komponente den Buszyklus verlängern kann.
 - Einfügen von Wartezyklen;
- **Vorteil:**
 - Unterschiedlich schnelle Speicher und Geräte können individuell bedient werden

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Semi-synchroner Bus



Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Semi-synchroner Bus

- Die Adresse wird zu Beginn des Buszyklus (T_a) auf den Adressbus gelegt.
- Der Buszyklus wird in T_b nur dann abgeschlossen, wenn rechtzeitig vor dem Ende $\overline{\text{READY}} = 0$ ist.
- Ansonsten werden so lange Wartezyklen (T_w) eingefügt, bis $\overline{\text{READY}} = 0$ ist.

- Wird auch als synchroner Bus mit Wartezyklen bezeichnet;

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

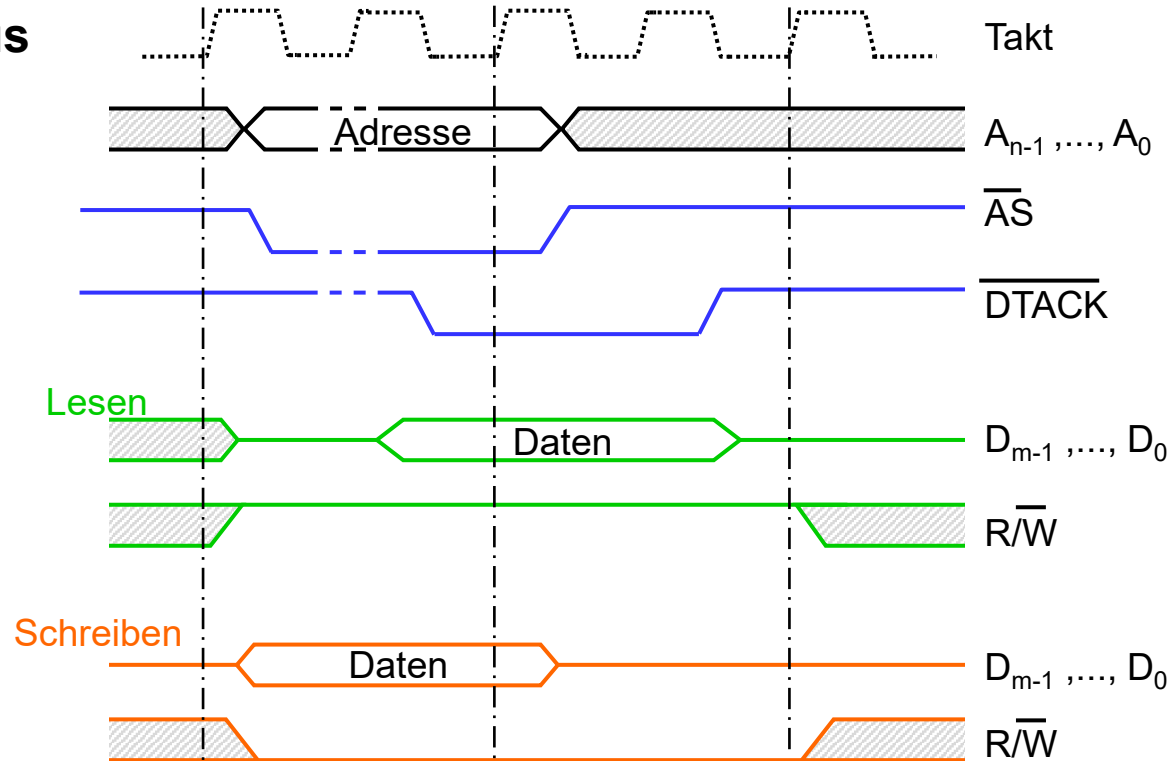
■ Asynchroner Bus

- Die aktive und die passive Komponente informieren sich wechselseitig über Steuersignale im Handshake-Verfahren.
- Steuersignale:
 - \overline{AS} (Address Strobe) von der aktiven Komponente;
 - \overline{DTACK} (Data Transfer Acknowledge) von passiver bzw. aktiver Komponente;
- Vollständig asynchroner Übertragungsablauf:
 - Anschluss von Komponenten mit unterschiedlichen Zugriffszeiten möglich;
- Der Bustakt spielt keine Rolle mehr für die Synchronisation der Übertragung (nur noch für die Synchronisation der Signale: synchrones Steuerwerk)

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Asynchroner Bus



Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Asynchroner Bus

■ Lesen:

- Mit $\overline{AS} = 0$ (Address Strobe) zeigt Prozessor an, dass er eine gültige Adresse auf den Adressbus gelegt hat.
- Mit $\overline{DTACK} = 0$ zeigt der Speicher an, dass er die Daten zur Verfügung gestellt.
 - Zwischen $\overline{AS} = 0$ und $\overline{DTACK} = 0$ kann eine beliebige Zeitspanne liegen.
- Wird $\overline{DTACK} = 0$, so nimmt der Prozessor die Adresse wieder vom Bus und setzt \overline{AS} wieder auf 1.
- Daraufhin nimmt der Speicher das Datum vom Datenbus (Lesen) und setzt \overline{DTACK} wieder zu 1.

Bus-Systeme

■ Datenübertragung: Zeitverhalten der Signale

■ Asynchroner Bus

■ Schreiben:

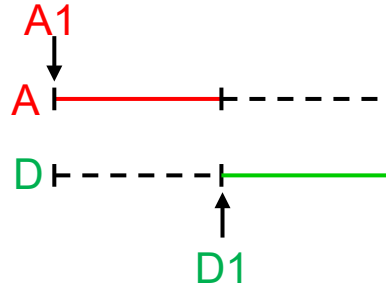
- Mit $\overline{AS} = 0$ (Address Strobe) zeigt Prozessor an, dass er eine gültige Adresse auf den Adressbus gelegt hat und die Daten auf dem Datenbus gültig sind
- Mit $\overline{DTACK} = 0$ zeigt der Speicher an, dass er die Daten übernommen hat
 - Zwischen $\overline{AS} = 0$ und $\overline{DTACK} = 0$ kann eine beliebige Zeitspanne liegen
- Wird $\overline{DTACK} = 0$, so nimmt der Prozessor die Adresse und die Daten wieder vom Bus und setzt \overline{AS} wieder auf 1 und \overline{DTACK} wieder zu 1

Bus-Systeme

■ Bus-Arten

■ Split-Bus

- Getrennte Adress- und Datenleitungen;
 - Die Leitungen des Adressbusses werden für die Übertragung von Adressen benutzt.
 - Die Leitungen des Datenbusses werden für die Übertragung von Daten benutzt.

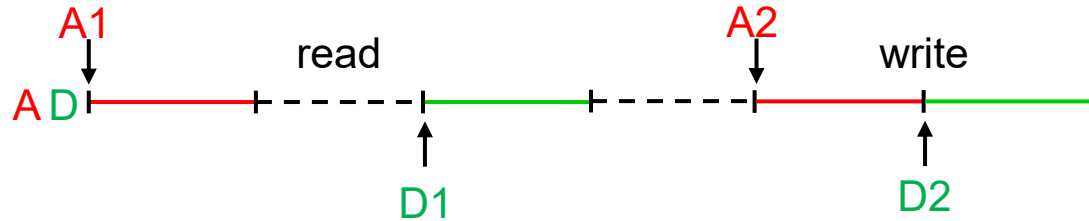


Bus-Systeme

■ Bus-Arten

■ Mux-Bus

- Adress- und Datenleitungen sind zusammengefasst.
- Die Adressen und Daten werden nacheinander im Multiplexbetrieb übertragen.



Bus-Systeme

- **Leistungsmerkmale: Busbandbreite**

- **max. Übertragungsrate (in Mbytes/s):**

- **Produkt aus**

- **Anzahl der gleichzeitig übertragbaren Bytes**

- gegeben durch die Anzahl der für die Übertragung von Daten gegebenen Leitungen;

- **Bustaktfrequenz**

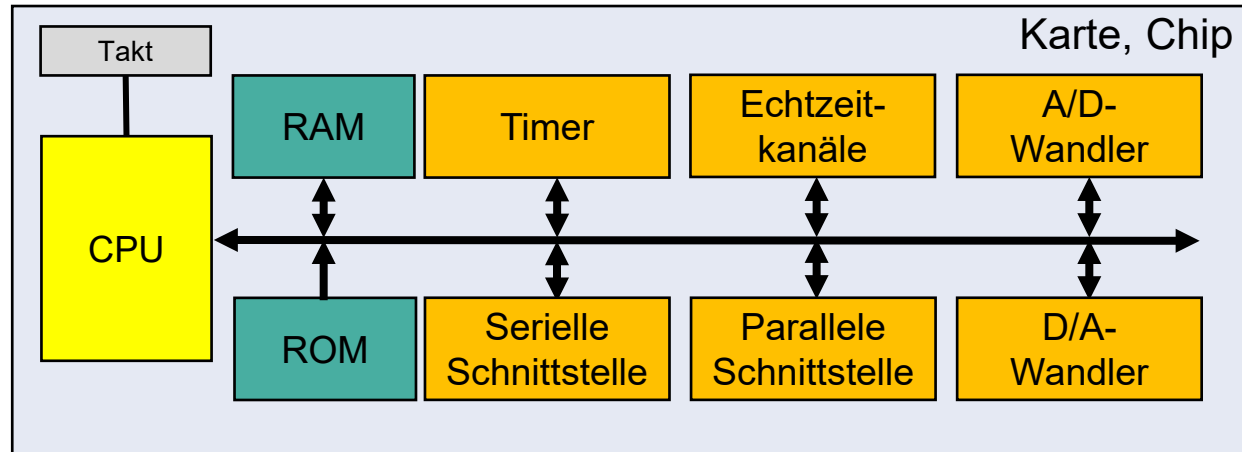
- maximale Frequenz, mit der die Signalleitungen eines Busses betrieben werden können;

- **Geteilt durch die Anzahl der für die Übertragung eines Datums notwendigen Bustakte;**

Systemstrukturen

■ Einbussystem

- Mikroprozessorsysteme mit minimaler Hardware für fest umrissene Aufgaben, z.B. für Steuerungsaufgaben;
- Die gesamte Hardware ist auf einer gedruckten Karte (Platine, Board) oder Chip untergebracht.
- **Beispiel:**

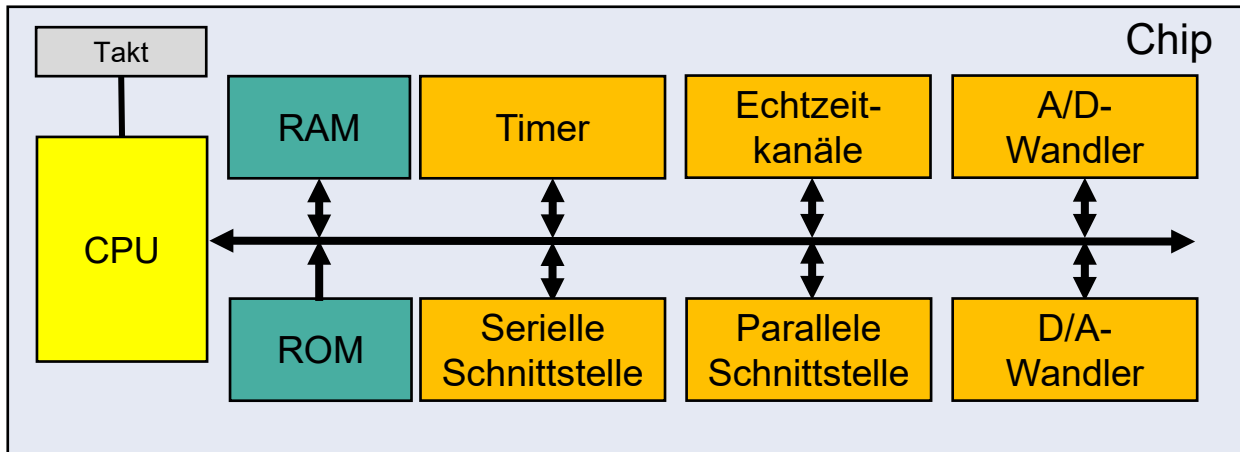


Systemstrukturen

■ Einbussystem

■ Mikrocontroller:

- Einfacher Mikrorechner auf einem Chip mit speziell für Steuerungs- und Kommunikationsaufgaben zugeschnittenen Schnittstellenbausteinen



Einfacher Mikroprozessor

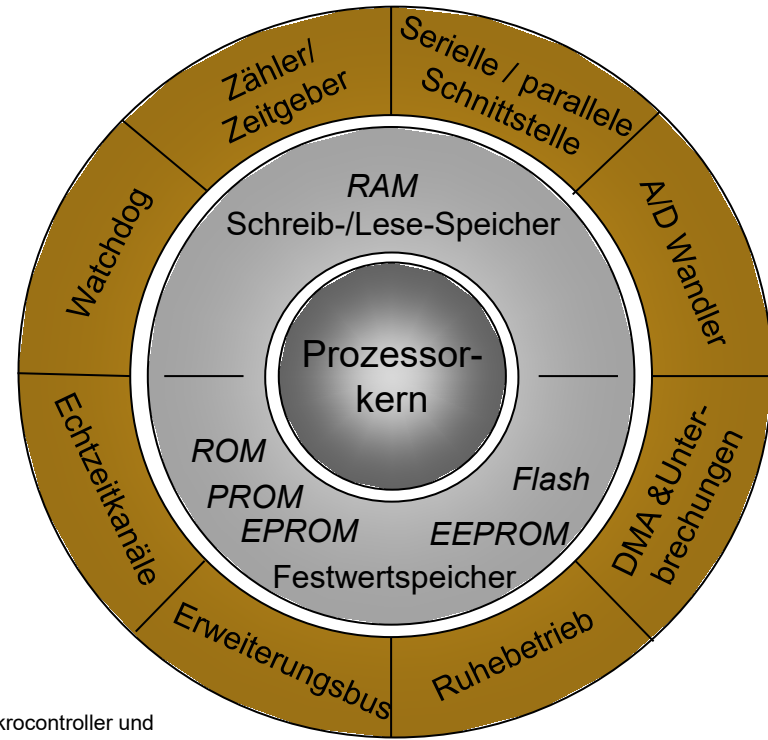
- Einfacher Prozessorkern mit geringer Verarbeitungsbreite: 4- oder 8-Bit
- Einfacher Befehlssatz

Viele Varianten:

Ausgewählte Beispiele von Speicher- und Schnittstellenbausteinen

Systemstrukturen

- Einbussystem
 - Mikrocontroller: Schalenmodell



Nach: U. Brinkschulte; T. Ungerer: Mikrocontroller und Mikroprozessoren. Springer-Verlag, Heidelberg, 2. Auflage, 2007

Systemstrukturen

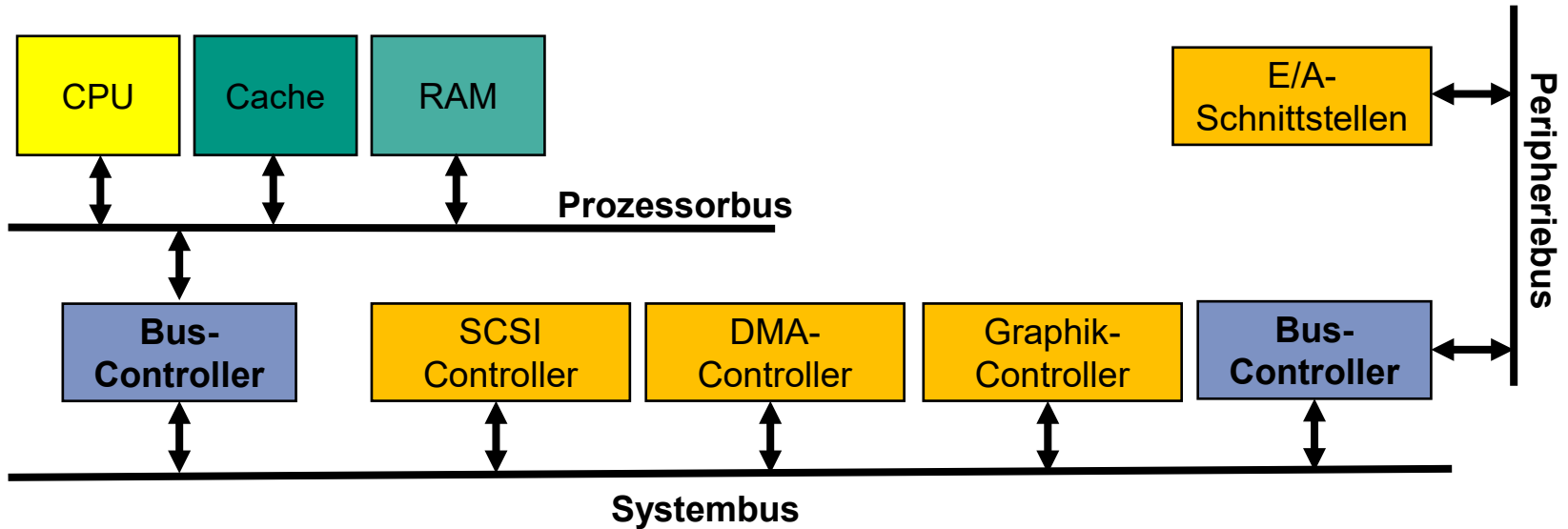
■ Mehrbussystem

- Aufbau eines Computers mit mehreren Bussen
 - Die verschiedenen Busse sind hinsichtlich ihrer Eigenschaften an die anzuschließenden Komponenten angepasst.
 - Auf den verschiedenen Bussen können zu einem Zeitpunkt unabhängig voneinander Datenübertragungen stattfinden.
- Mikroprozessorsystem mit hierarchisch angeordneten Ebenen an Bussen

Systemstrukturen

■ Mehrbussystem

■ Beispiel



Systemstrukturen

■ Mehrbussystem

■ Beispiel

■ Prozessor-/Speicherbus

- Schnelle Datentransfers zwischen den angeschlossenen Komponenten (Prozessor, Speicher)
- Wird im mit den Signalleitungen des Mikroprozessors gebildet (Busschnittstelle des Prozessors)

■ Zentraler Systembus (globaler Bus)

- Anschluss schneller E-/A-Module
- Beschleuniger

■ Peripheriebus

- Ein-/Ausgabeschnittstellen

Systemstrukturen

■ Mehrbussystem

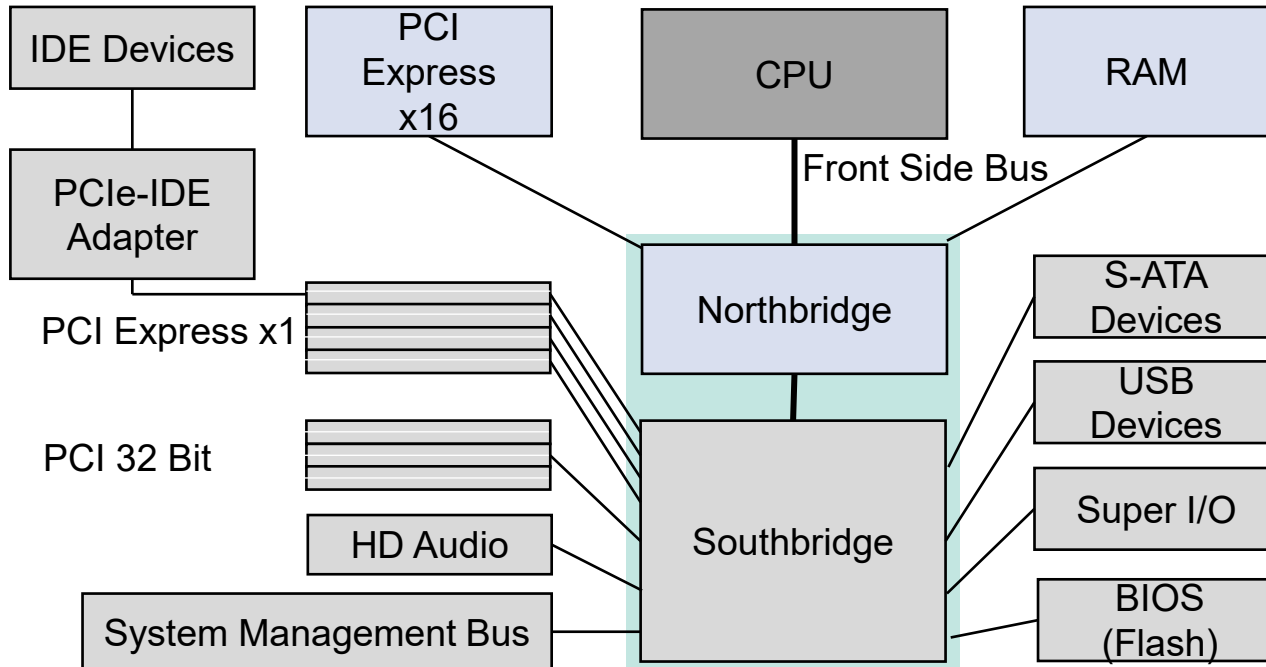
■ Beispiel

■ Bus-Controller

- bildet die Schnittstelle zwischen den Bussen zweier benachbarter Hierarchieebenen;
- übernimmt die Steuerung des Datentransfers zwischen Komponenten, die an diesen Bussen angeschlossen sind
 - Anpassen der unterschiedlichen Geschwindigkeiten und der verschiedenen Busprotokolle der Busse
 - Datenpufferung

Mehrbussystem

■ Fallstudie klassischer PC



Mehrbussystem

■ Fallstudie klassischer PC

■ Northbridge

- Schnittstelle zum Prozessor: Front Side Bus
- Speicher-Controller zum Anschluss von Speichermodulen
- PCIe Schnittstelle zur Anbindung schneller Komponenten an den Prozessor
- Schnittstelle zur Southbridge

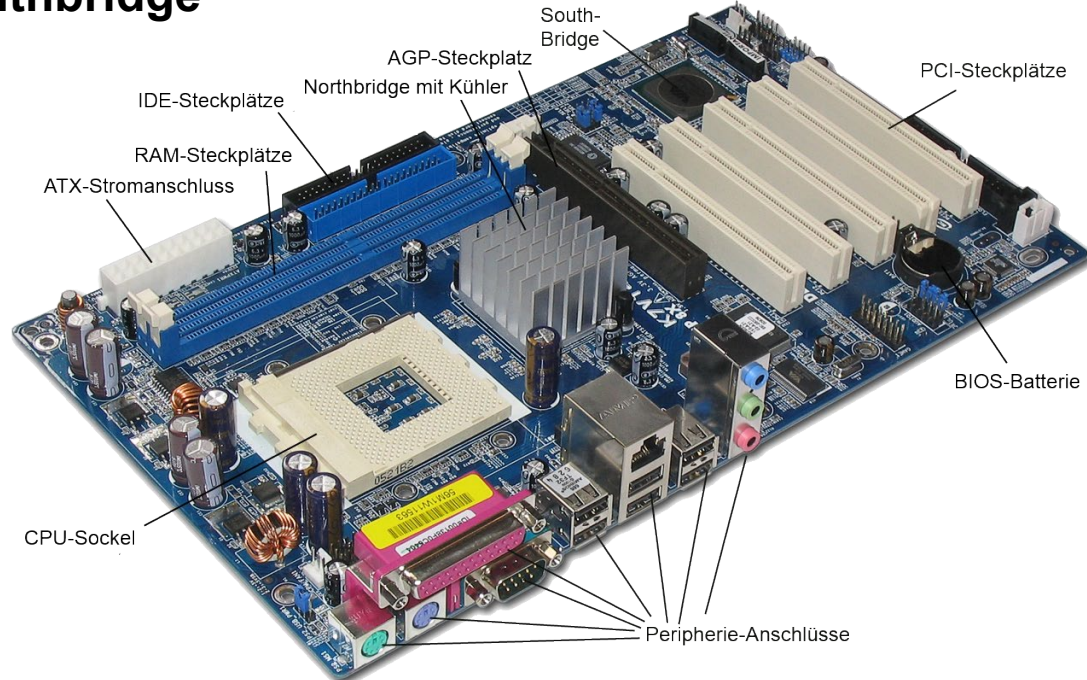
■ Southbridge

- Steuerung der Datenübertragung zu peripheren Komponenten
- Schnittstelle zu PCI Express Verbindungssystem
- Schnittstelle zu PCI Bus
- Schnittstelle zu USB
- S-ATA-Schnittstelle (Serial ATA) zur Anbindung von Festplatten

Mehrbussystem

- Fallstudie klassischer PC
- Chipset: Northbridge und Southbridge

ASRock K7VT4A Pro – Mainboard /
VIA KT400A chipset



https://de.wikipedia.org/wiki/Northbridge#/media/Datei:ASRock_K7VT4A_Pro_Mainboard_Labeled_German.PNG

Mehrbussystem

- **Fallstudie klassischer PC**

- **Anmerkung**

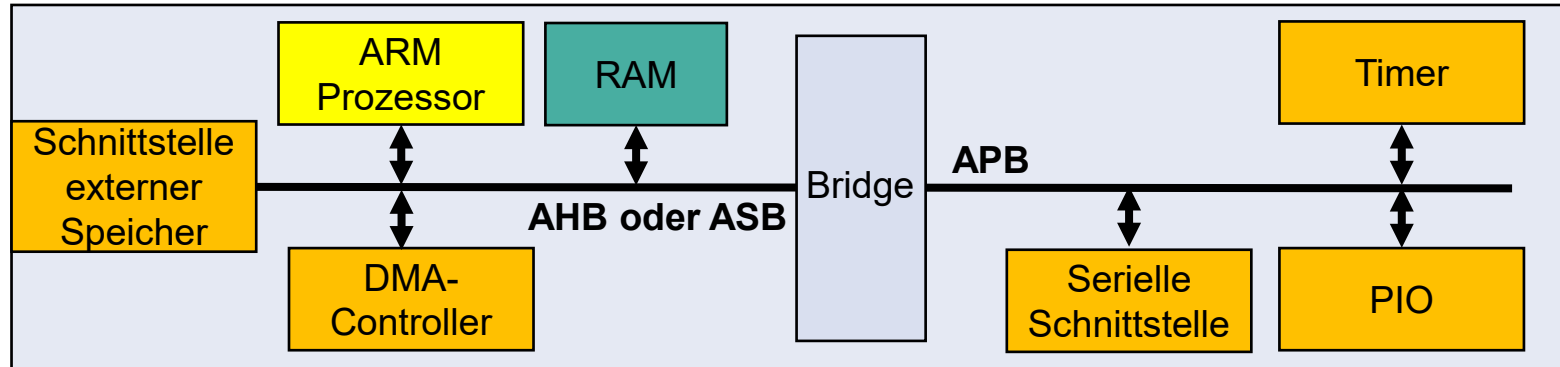
- Funktionalität der Northbridge heute vielfach auf Prozessorchip

Mehrbussystem

- **Fallstudie ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)**
 - **Bus-Spezifikation für ein On-Chip Mehrbussystem**
 - Zugeschnitten auf ARM Cores
 - Portabilität und Wiederverwendbarkeit
 - System Reset und Taktung
 - Geringer Leistungsverbrauch
 - Batteriebetriebene und kostengünstige Anwendungen im Bereich eingebetteter Systeme
 - Test -Unterstützung für CPU Makrozellen mit vielen I/O-Pins

Mehrbussystem

- ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)
 - AMBA System:



Mehrbussystem

- **ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)**
 - **AHB (Advanced High-performance Bus)**
 - Anbindung von Komponenten mit hohen Anforderungen an die Datenübertragung
 - CPU, Speicher, DMA-Controller, Digitaler Signal Prozessor, Schnittstelle zum externen Speicher
 - Brücke
 - Schnittstelle zwischen AHB / ASB und APB
 - **ASB (Advanced System Bus)**
 - Alternativer Systembus

Mehrbussystem

- **ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)**
 - **APB (Advanced Peripheral Bus)**
 - Anbindung von Ein-/Ausgabe-Schnittstellenbausteinen
 - Programmierte Ein-/Ausgabe
 - Speicherbezogene Ein-/Ausgabe

Mehrbussystem

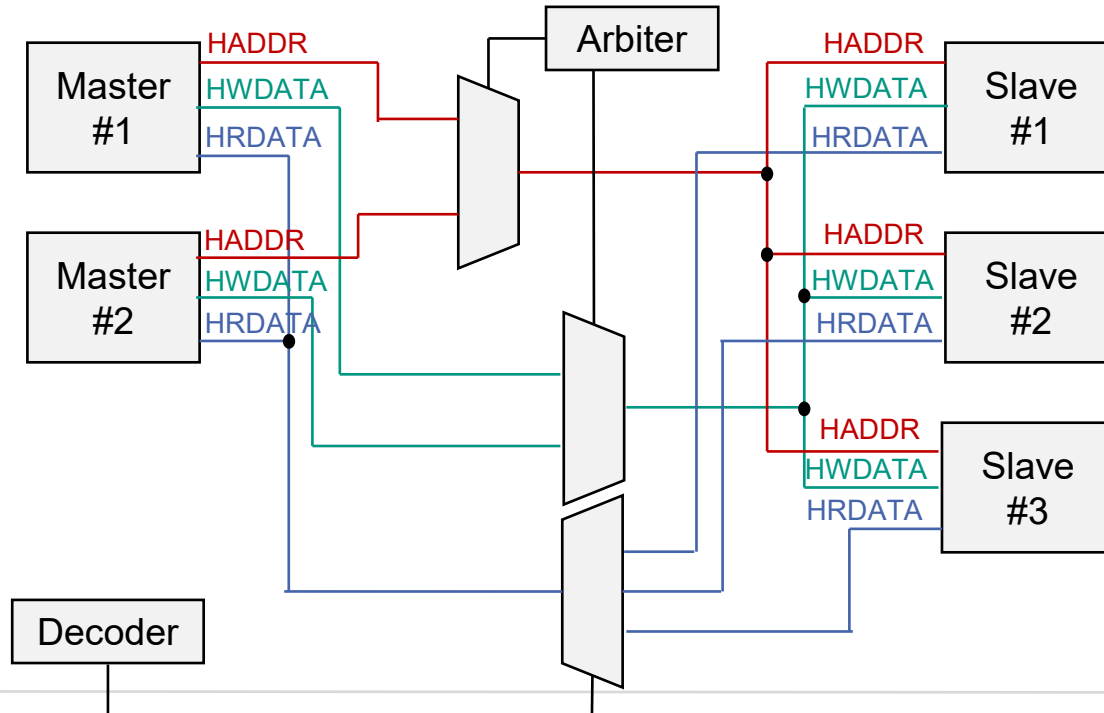
- **ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)**
 - **AHB-Komponenten**
 - **AHB Master**
 - Initiiert Lese- und Schreib-Operationen
 - Liefert Adress- und Steuerinformationen
 - Nur ein AHB Master hat zu einem Zeitpunkt Zugriff auf den Bus
 - **AHB Slave**
 - Beantwortet die Lese- oder Schreib-Operationen
 - Liefert Signale, die dem AHB Master einen erfolgreichen oder fehlerhaften Datentransfer anzeigen
 - **AHB Arbiter**
 - Erteilt einem AHB Master den Zugriff auf den AHB

Mehrbussystem

- **ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)**
 - **AHB-Komponenten**
 - **AHB Decoder**
 - Dekodierung der Adresse für einen Datentransfer
 - Liefert Select-Signal für die AHB-Slave-Komponente

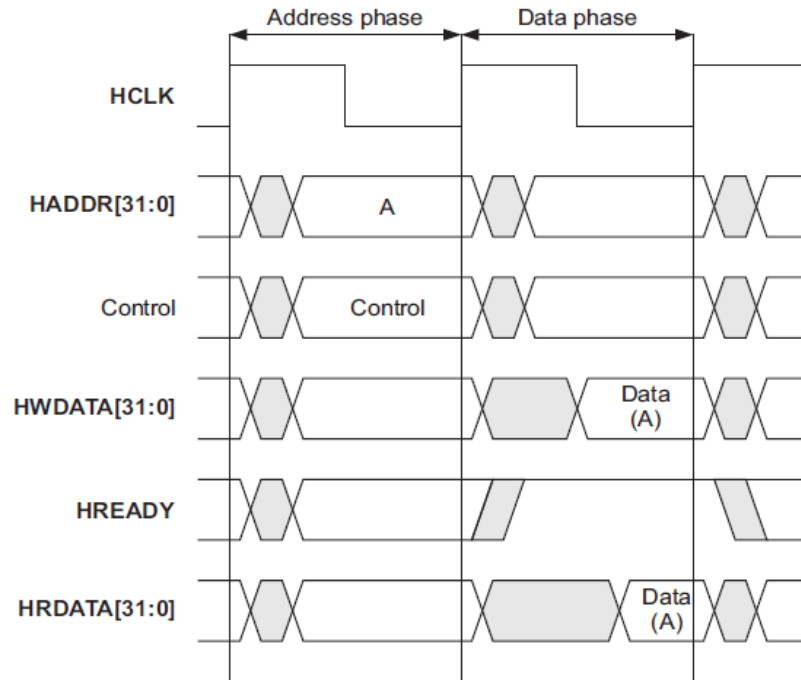
Mehrbussystem

- ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)
 - Anbindung von AHB-Master und Slave-Komponenten



Mehrbussystem

- ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)
 - AHB-Buszyklus ohne Wartezyklen

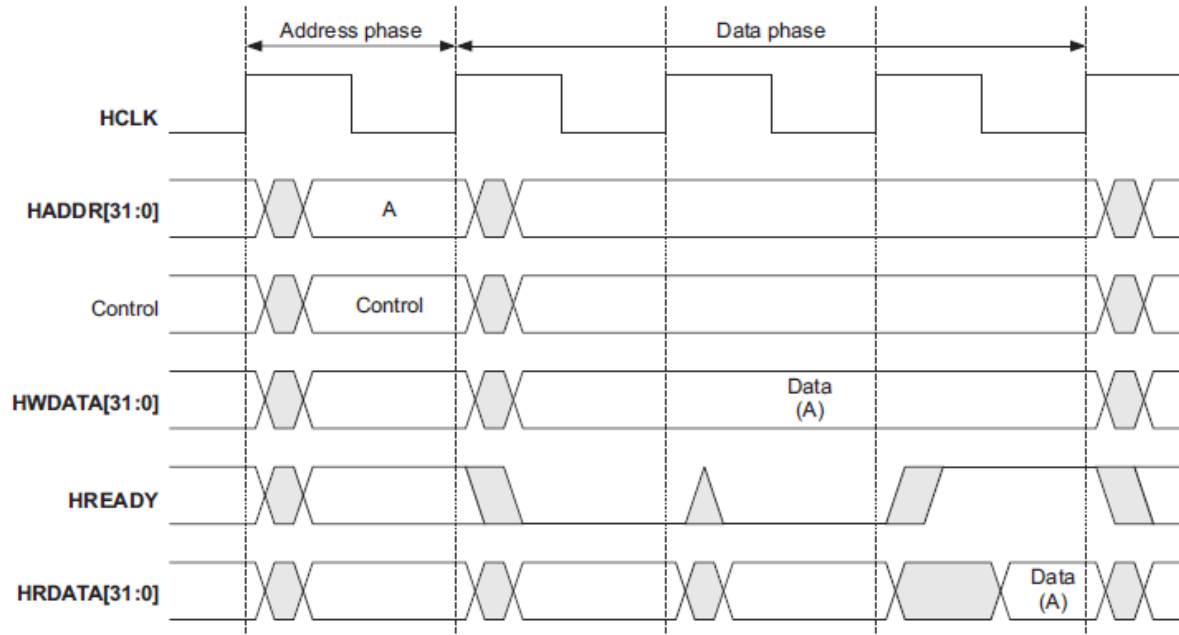


1. Master gibt Adresse auf HADDR und Steuersignale auf Control nach einer der steigenden Taktflanke
2. Mit der nächsten steigenden Taktflanke übernimmt der Slave die Adresse und die Steuerinformationen
3. Nachdem der Slave die Adresse und die Steuerinformationen übernommen hat, beginnt er die entsprechende Antwort aufzulegen, was vom Master mit der 3. steigenden Flanke übernommen wird.

Quelle: [AMBA Specification \(Rev 2.0\) - Arm](#)

Mehrbussystem

- ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)
 - AHB-Buszyklus mit Wartezyklen

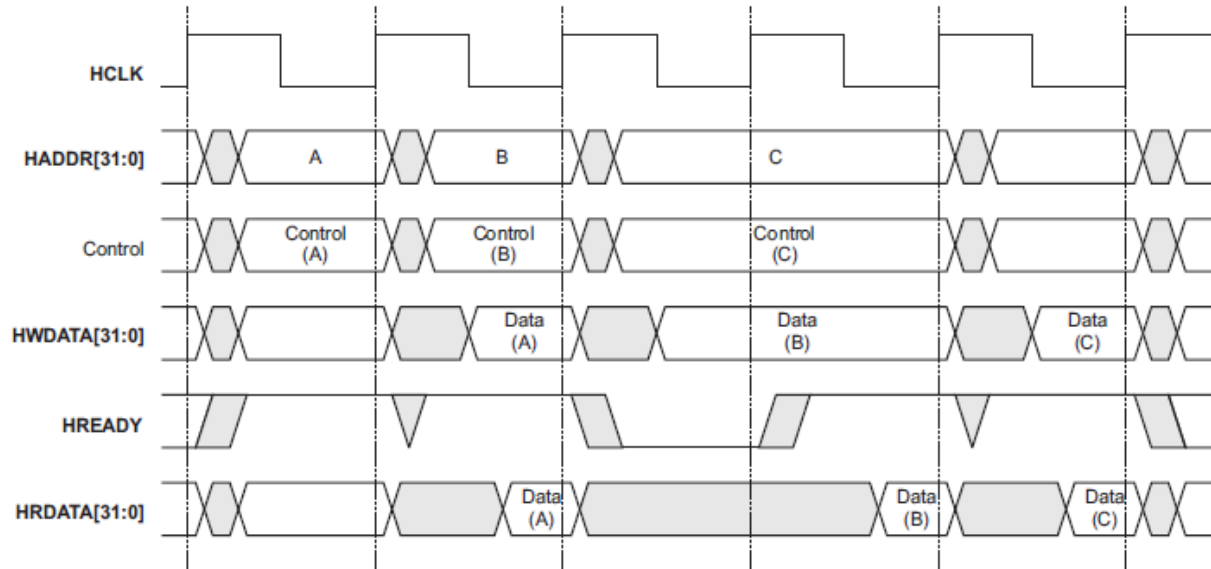


Wartezyklen werden mit HREADY = low angezeigt

Quelle: [AMBA Specification \(Rev 2.0\) - Arm](#)

Mehrbussystem

- ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)
 - AHB-Buszyklus Pipeline-Modus



Überlappte Schreib-, Lese-Zyklen

Da für den Zyklus B ein Wartetakt eingefügt wird, muss auch im Zugriff C gewartet werden.

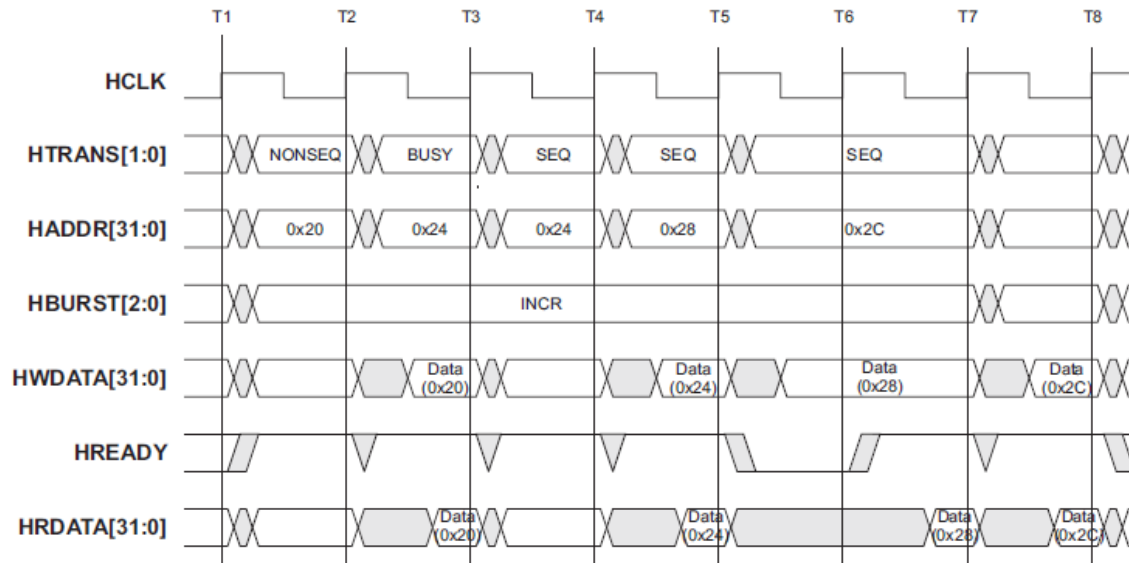
Quelle: [AMBA Specification \(Rev 2.0\) - Arm](#)

Mehrbussystem

■ ARM AMBA (Advanced Microcontroller Bus Architecture Protocol)

■ AHB-Buszyklus Burst-Zugriffe

- Blockübertragung von Daten an aufeinanderfolgenden Adressen



Quelle: [AMBA Specification \(Rev 2.0\) - Arm](#)

Die 1. Übertragung in einem Burst-Zyklus wird durch NONSEQUENTIAL angezeigt.

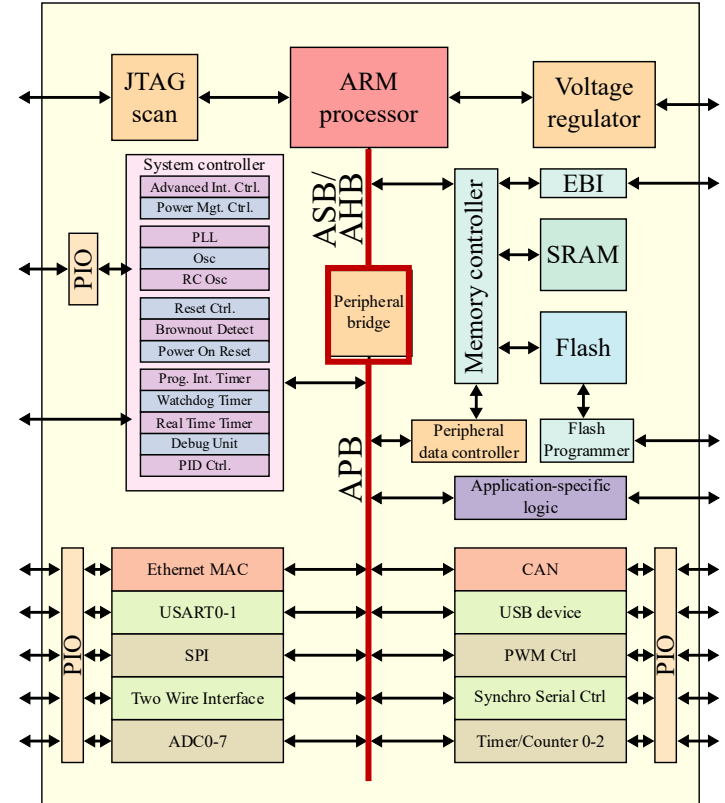
Da der Master die 2. Übertragung nicht gleich beginnen kann, wird ein BUSY-Zyklus angezeigt, um den Start des nächsten Transfers zu verzögern.

Danach folgen die Burst-Zugriffe ohne Verzögerung auf Seiten des Masters. Slave legt Wartezyklus ein.

HBURST steuert den Burst-Zugriff: z. B. INCR - Incrementing burst of undefined length);

Mehrbussystem

■ Fallstudie ARM SoC mit AMBA System



Systemstrukturen

■ System-on-a-Chip (SoC)

■ Weiterentwicklung der Mikrocontroller-Idee:

- Aufbau von anwendungsspezifischen Mikroprozessorsystemen mit möglichst wenig Hardware

■ Realisieren ein vollständiges System auf einem Chip

■ ASIC:

- IP Cores (Intellectual Property Cores) werden in ASIC-Bibliotheken zur Verfügung gestellt und mit anwendungsspezifischen Hardware-Komponenten gemeinsam auf einem Chip integriert

■ Rekonfigurierbare Hardware

- Neben festen Prozessorkern und Speicher ermöglichen konfigurierbare Komponenten die Realisierung unterschiedlicher Anwendungen