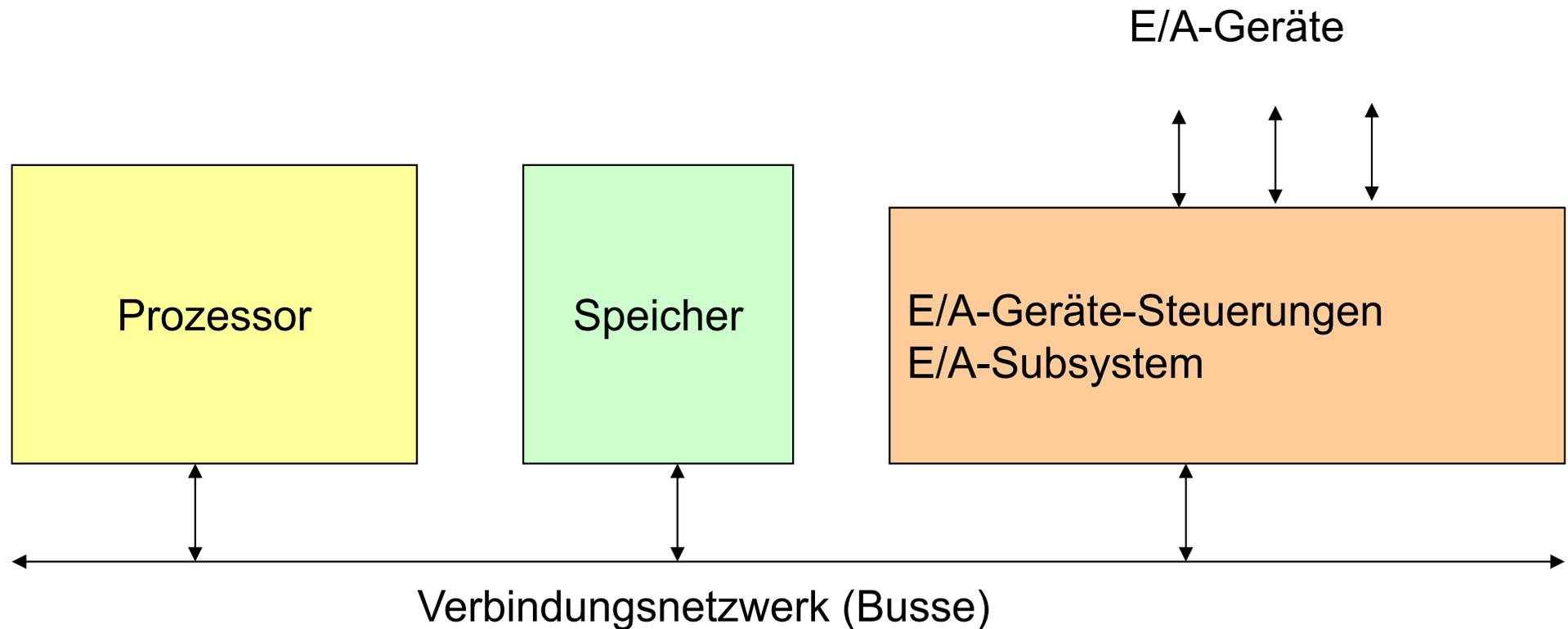


Kapitel 7

Systemstrukturen und Busse

7.1 Systemstrukturen

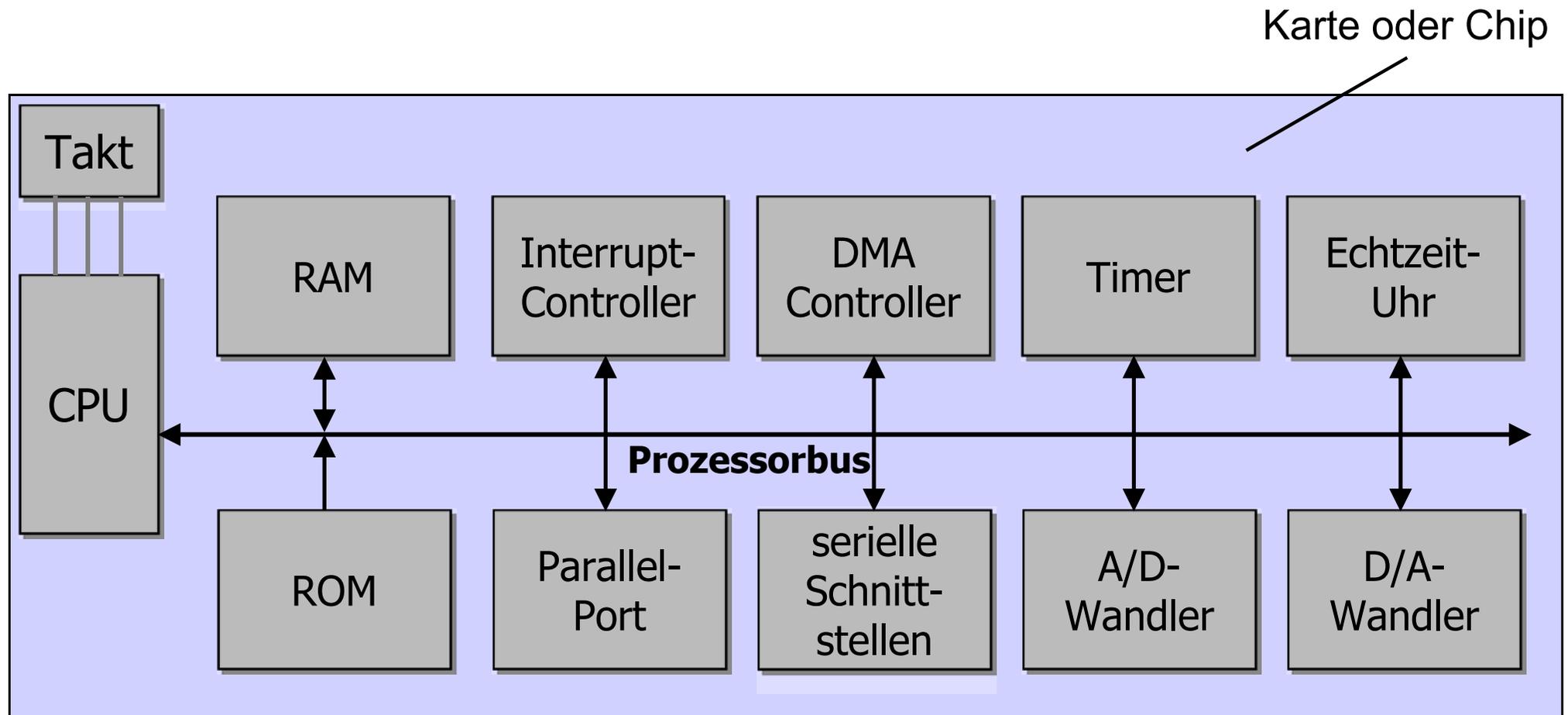
■ Einfaches Rechnermodell



7.1 Systemstrukturen

■ Einkartensystem

- Mikroprozessorsysteme mit minimaler Hardware



7.1 Systemstrukturen

■ Einkartensystem

- Mikroprozessorsysteme mit minimaler Hardware:
 - z.B. für Steuerungsaufgaben (fest umrissene Aufgaben)
 - gesamte Hardware ist auf einer gedruckten Karte (Platine, Board) untergebracht;
 - über den Prozessorbus sind die Systembausteine (Prozessor, Speicher, Ein-
Ausgabeschnittstellen miteinander verbunden
 - niedrige Herstellungskosten;
 - Systemstruktur kann im nachhinein nicht geändert werden

7.1 Systemstrukturen

- **Busorientierte Mehrkartensysteme**
 - Mikroprozessorsysteme für allgemeine Anwendungen
 - modulare Mehrkartensysteme
 - Systemstruktur kann geändert werden

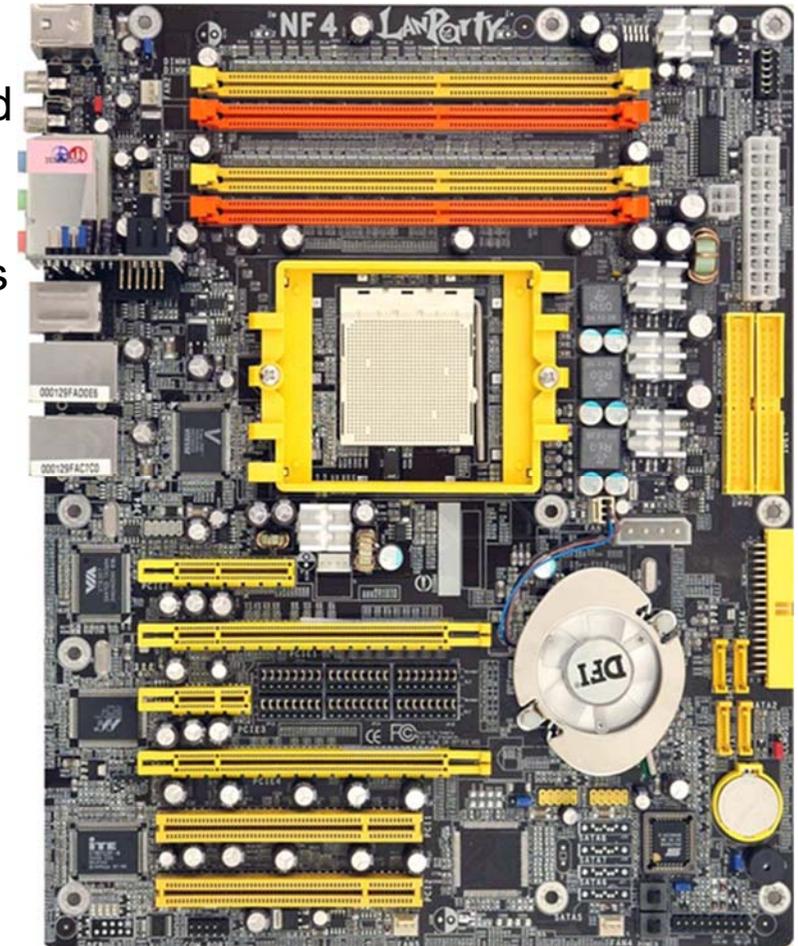
7.1 Systemstrukturen

■ Busorientierte Mehrkartensysteme

■ Erweiterung eines Grundsystems

- elementare Komponenten des Systems sind üblicherweise auf einer großformatigen Grundkarte (main board, motherboard) zusammengefasst und über den Systembus miteinander verbunden;
- der Systembus wird über die auf der Grundkarte untergebrachten Komponenten verlängert und mit Stecksockeln (slots) versehen;
- Erweiterung des Systems durch Einstecken von Karten in Slots

PC Motherboard



7.1 Systemstrukturen

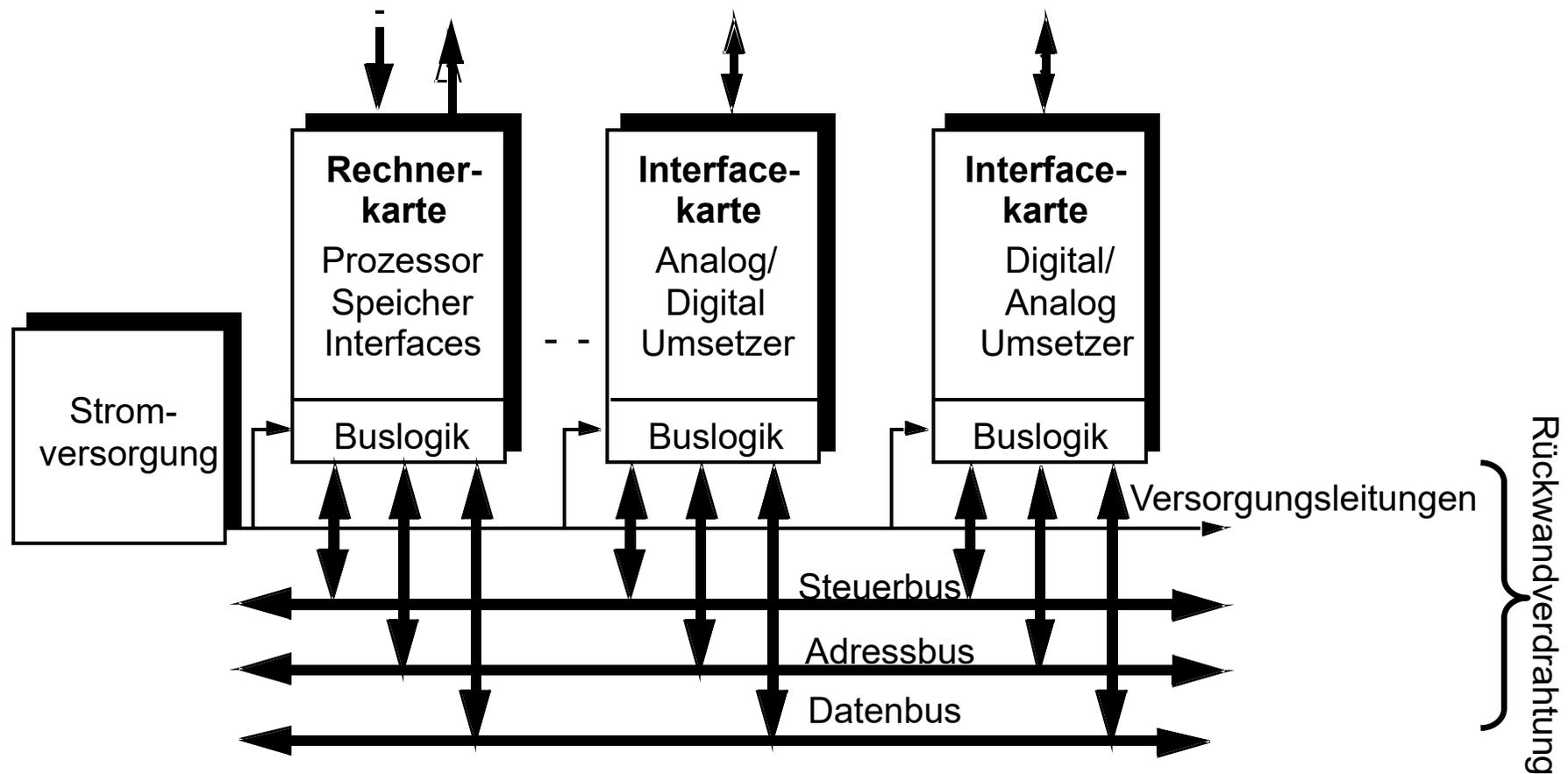
■ Busorientierte Mehrkartensysteme

■ Vollständiges Konfigurieren eines Systems

- Industrielle Mikroprozessorsysteme
- Funktionseinheiten (Baugruppen) eines Systems werden grundsätzlich als Steckkarten ausgelegt, wobei diese als Einschübe nebeneinander in einem Baugruppenträger (Rahmen) untergebracht werden
- zur Zusammenschaltung der Karten werden ihre Signalleitungen auf eine Rückwandverdrahtung (realisiert als gedruckte Karte, backplane) geführt;
- Prozessorunabhängiger Systembus (Standardisierung)

7.1 Systemstrukturen

- Busorientierte Mehrkartensysteme
 - Vollständiges Konfigurieren eines Systems



7.1 Systemstrukturen

■ Busorientierte Mehrkartensysteme

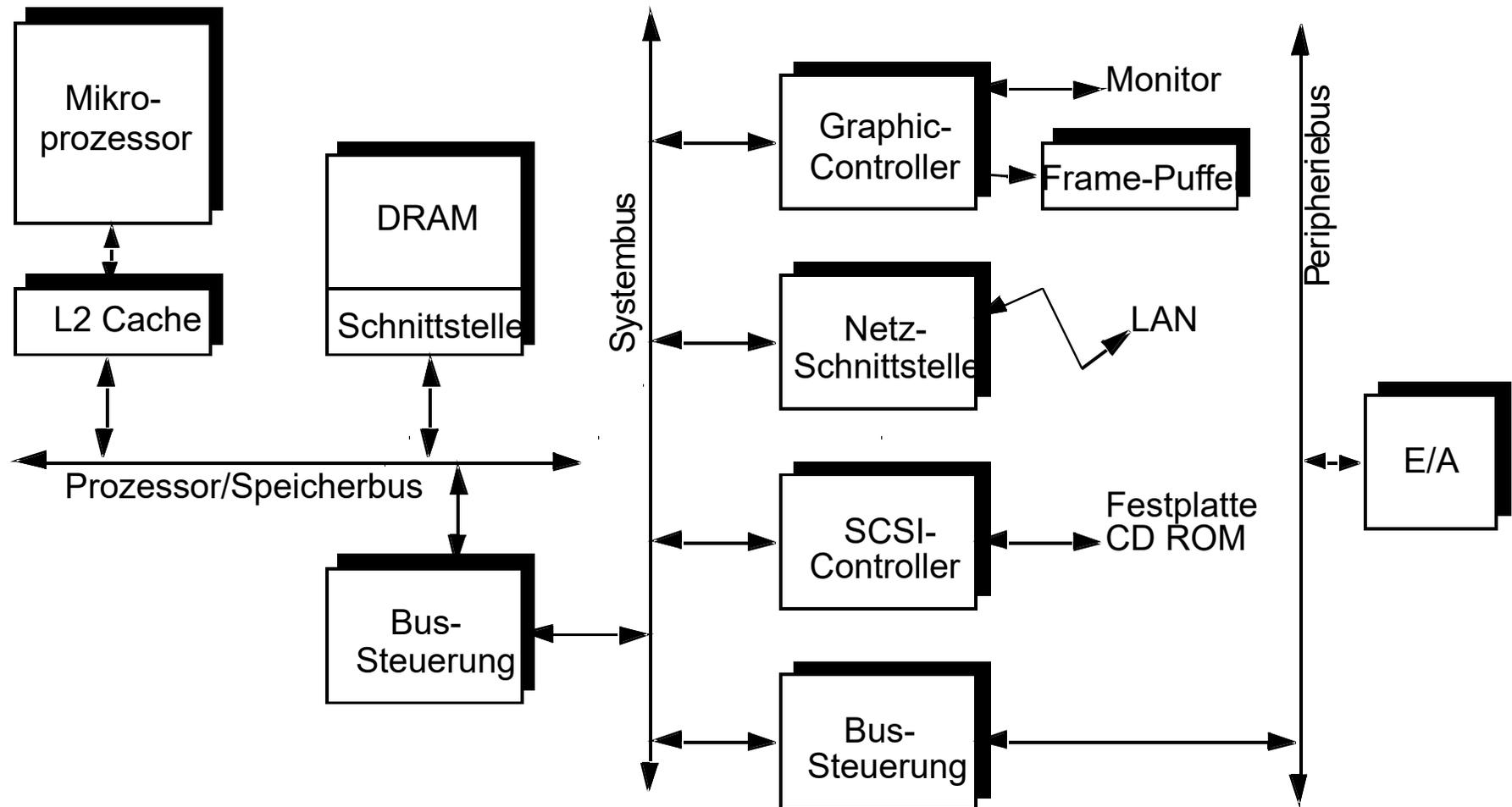
■ Vollständiges Konfigurieren eines Systems

- Industrielle Mikroprozessorsysteme
- Funktionseinheiten (Baugruppen) eines Systems werden grundsätzlich als Steckkarten ausgelegt, wobei diese als Einschübe nebeneinander in einem Baugruppenträger (Rahmen) untergebracht werden
- zur Zusammenschaltung der Karten werden ihre Signalleitungen auf eine Rückwandverdrahtung (realisiert als gedruckte Karte, backplane) geführt;
- Prozessorunabhängiger Systembus (Standardisierung)

7.1 Systemstrukturen

■ Mehrbussysteme

■ Beispielkonfiguration



7.1 Systemstrukturen

■ Mehrbussysteme

■ Prozessor-/Speicherbus

- schnelle Datenübertragung zwischen den angeschlossenen Einheiten
- Datenbusbreite des Prozessors
- auf Taktgeschwindigkeit des Prozessors zugeschnitten
- Systembus-Controller
 - Verbindung zwischen Prozessor- und Systembus;
 - Kopplung der Adress- und Datenleitungen;
 - Anpassung der Buszyklen;
 - Anschluss von anderen als Speicherkomponenten möglich (z.B. Graphik-Controller)

7.1 Systemstrukturen

■ Mehrbussysteme

■ zentraler Systembus (globaler Bus)

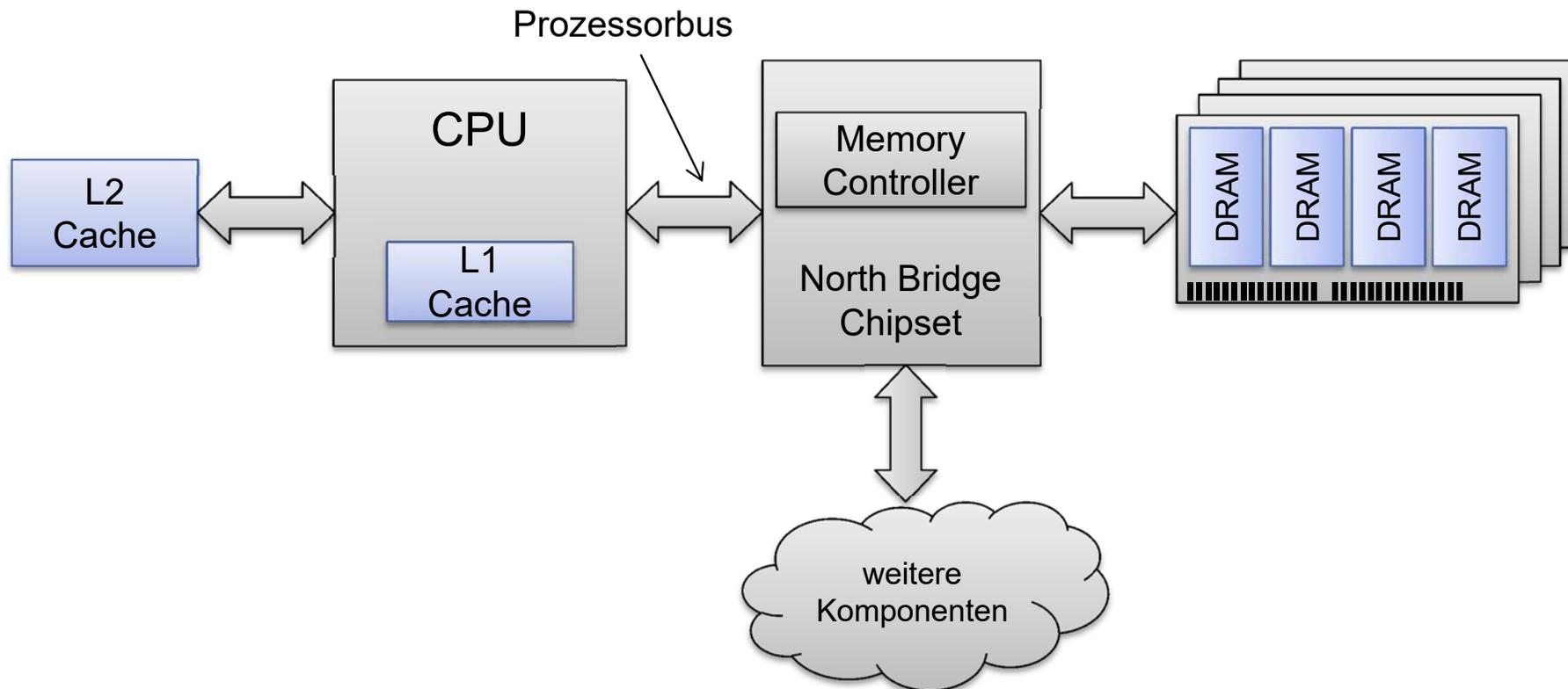
- mehrere Master (Prozessoren, DMA-Controller)
- Master kann die Steuerung des Busses übernehmen (aktive Komponente)
- global zugängliche Speicher- und Ein-Ausgabeeinheiten
- wird ergänzt durch weitere Busse (lokale Busse, Nachrichtenbus)

■ Peripheriebus

- fasst periphere Geräte (Magnetplatten, Magnetbandspeicher zusammen;
- Übertragung zwischen Geräten und Speicher und zwischen Geräten möglich
- 8-Bit Datenbusbreite (z.B. wegen Byte-Übertragung bei E/A)
- SCSI-Bus
- Übertragungsgeschwindigkeit ist an die Zugriffsgeschwindigkeit der Geräte angepasst;

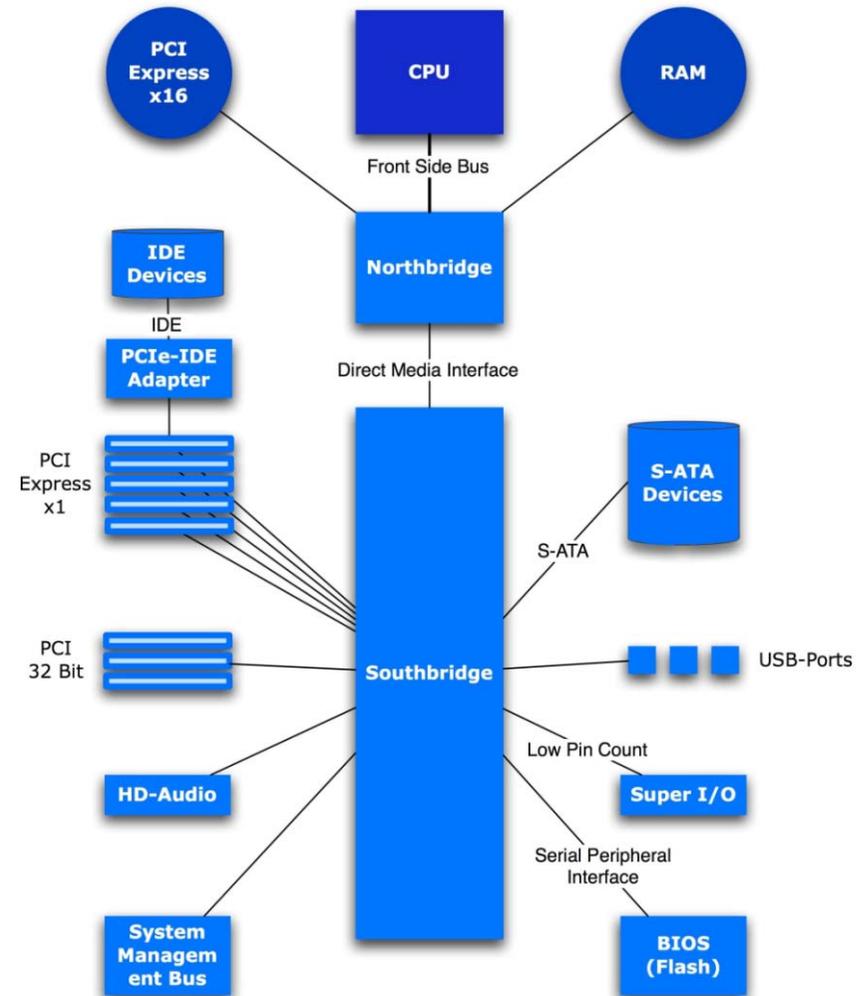
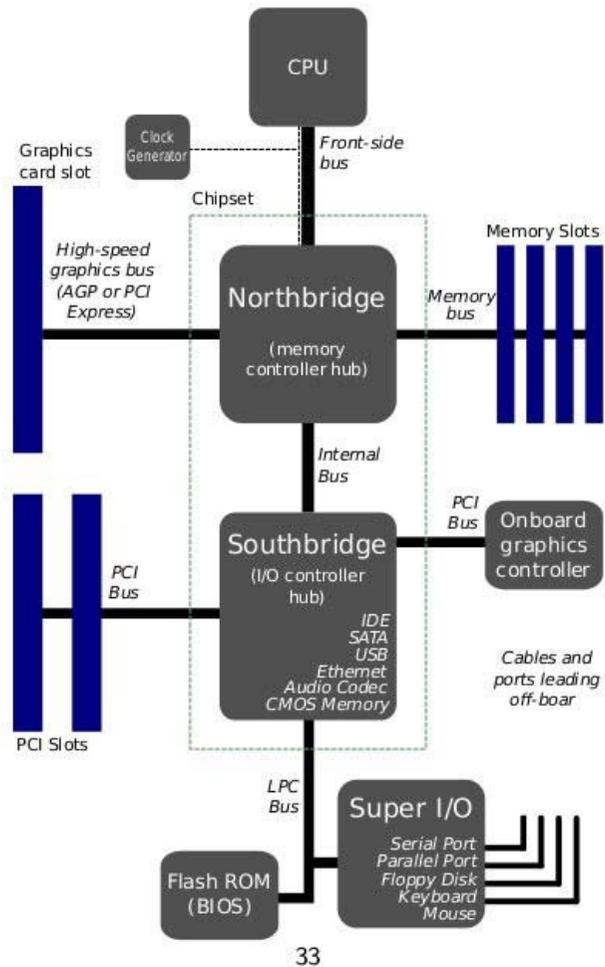
7.1 Systemstrukturen

■ Mehrbussysteme



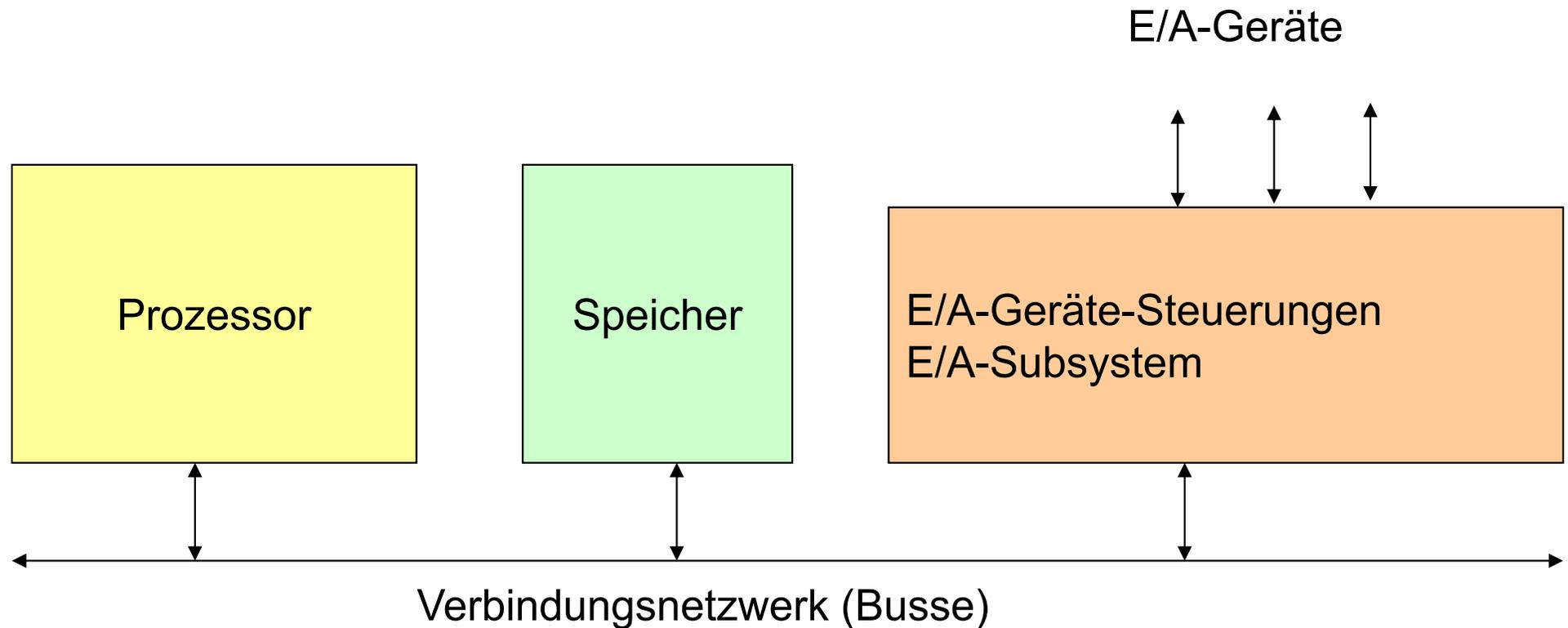
7.1 Systemstrukturen

Mehrbussysteme



7.2 Busse

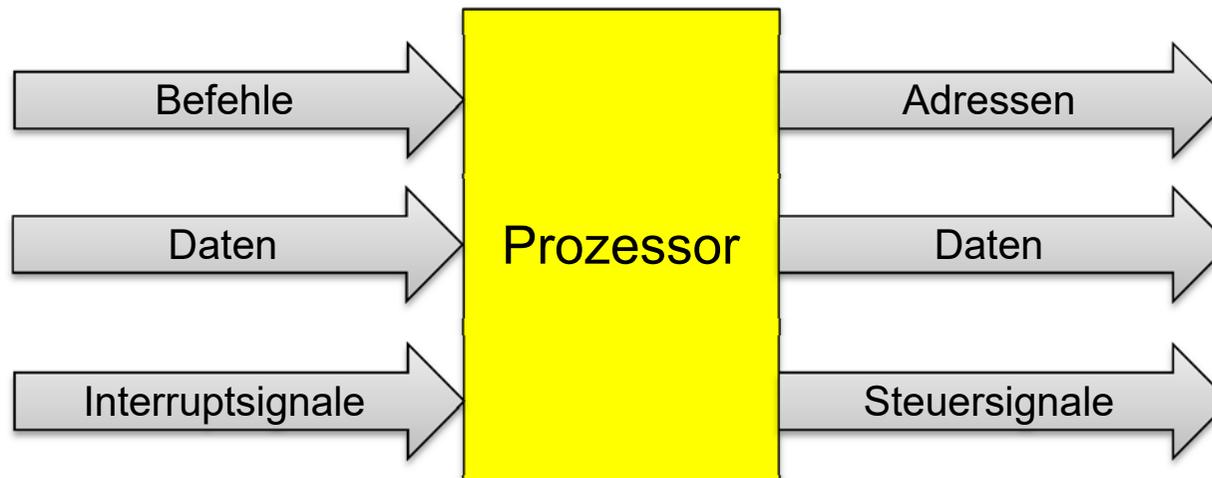
■ Einfaches Rechnermodell



7.2 Busse

■ Verbindungsstrukturen

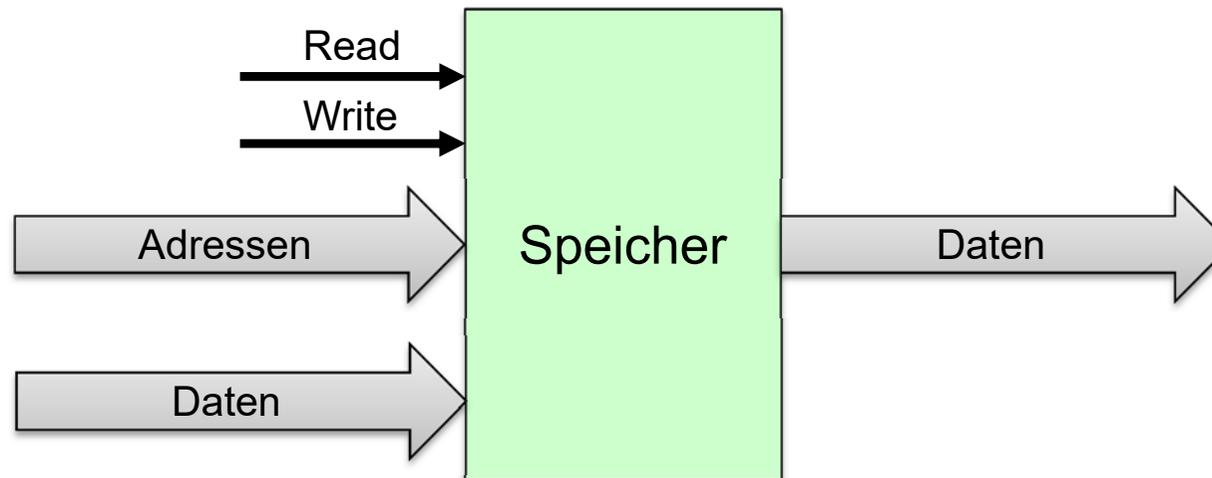
- Menge der Pfade, die die Komponenten des Rechners miteinander verbinden und über die Informationen ausgetauscht werden
- Komponenten mit den grundsätzlichen Ein- und Ausgabeformen



7.2 Busse

■ Verbindungsstrukturen

- Menge der Pfade, die die Komponenten des Rechners miteinander verbinden und über die Informationen ausgetauscht werden
- Komponenten mit den grundsätzlichen Ein- und Ausgabeformen

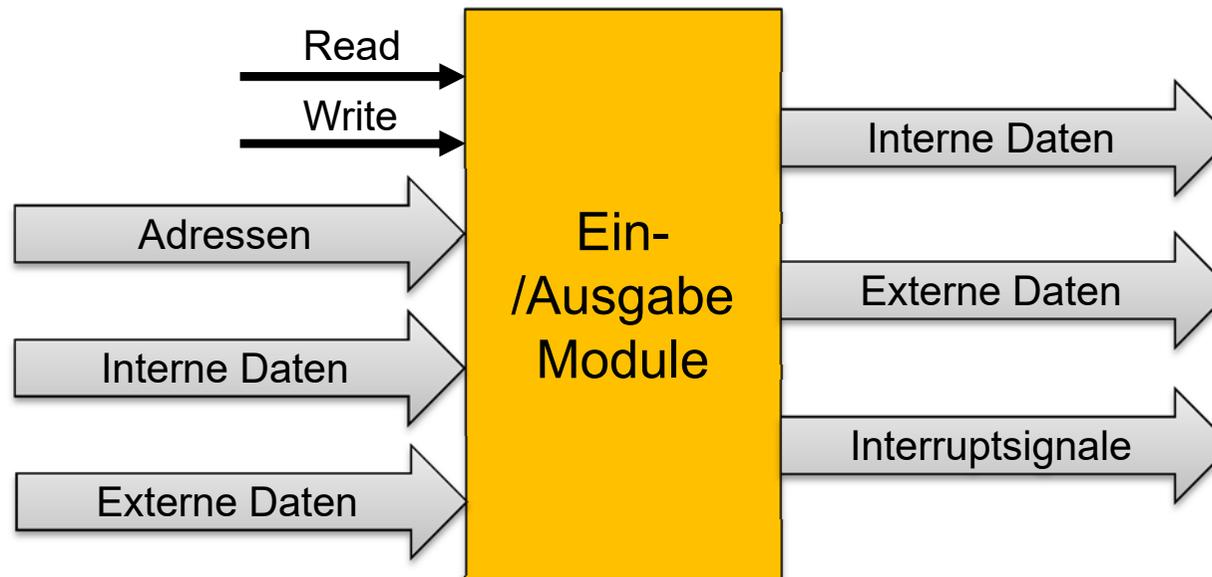


7.2 Busse

■ Verbindungsstrukturen

- Menge der Pfade, die die Komponenten des Rechners miteinander verbinden und über die Informationen ausgetauscht werden

- Komponenten mit den grundsätzlichen Ein- und Ausgabeformen



7.2 Busse

■ Verbindungsstrukturen

- Unterstützen folgende Arten von Übertragungen
 - Prozessor an Speicher: Prozessor schreibt Daten in den Speicher
 - Speicher an Prozessor: Prozessor liest Daten vom Speicher
 - Ein-/Ausgabe an Prozessor: Speicher liest Daten einer E/A-Geräts über die E/A-Schnittstelle
 - Prozessor an Ein-/Ausgabe: Prozessor sendet Daten an das E/A-Gerät
 - Ein-/Ausgabe an oder von Speicher: E/A-Schnittstelle tauscht direkt Daten mit dem Speicher (direkter Speicherzugriff, direct memory access, DMA)

7.2 Busse

■ Busspezifikation

■ Standardisierung

- Baugruppen verschiedener Hersteller können kombiniert werden
- Standardisierungsgremien
- Zusammenschluss von Rechnerherstellern
- Nationale oder internationale Organisationen

■ Spezifikationen beschreiben

- die mechanischen und elektrotechnischen Eigenschaften
- die Regeln für die Funktionsabläufe zwischen den Busteilnehmern (Busprotokolle)
- Realisierung durch Steuereinheiten

7.2 Busse

■ Busspezifikation

■ Funktionen der Bussteuereinheiten

■ Datenübertragung zwischen Master und Slave

- **Master:** bussteuernde, aktive Einheit (Prozessor, DMA-Einheit)
- **Slave:** passive Einheit (z. B. Speicher, Interface)
- Adressierung des Slave;
- Verwaltung der für die Übertragung erforderlichen Steuersignale

■ Interrupt-Verwaltung

- Priorisierung der von den verschiedenen Interruptquellen kommenden Anforderungssignale;
- Erzeugung des Gewährungssignals für die akzeptierte Quelle
- Anforderung des Datenbusses für die Übernahme von Statusinformationen dieser Quelle (z.B. Vektornummer)
- Ablauf wird durch einen oder mehrere Interrupt-Handler gesteuert

7.2 Busse

■ Busspezifikation

■ Funktionen der Bussteuereinheiten

■ Busarbitrierung (**Buszuteilung**) bei mehreren Mastern

- Priorisierung der von mehreren Mastern kommenden Anforderungssignale
- Verwaltung der Steuersignale für eindeutige Buszuteilung
- Steuerung durch zentralen Busverwalter (Arbiter)

■ Nutzfunktionen

- Stromversorgung
- Taktversorgung
- Systeminitialisierung

7.2 Busse

■ Busspezifikation

■ Bussignale

■ Leitungsbündel (bei System- oder Prozessorbus)

■ Datenbus

- Übertragung von Daten zwischen den Systemkomponenten
- Breite des Datenbus: Anzahl der Leitungen (z. B. 32, 64, 128, oder mehr)
- Über jede Leitung kann 1 Bit pro Zeiteinheit übertragen werden, deshalb bestimmt die Anzahl der Leitungen wieviele Bits pro Zeiteinheit übertragen werden können

■ Adressbus

- Über die Adressleitungen werden Adressinformationen übertragen
- Bestimmen die Quelle oder das Ziel der Daten auf dem Datenbus
- Die Breite des Adressbusses bestimmt die maximale Speicherkapazität
- Adressierung der Ein-/Ausgabeeinheiten (I/O Ports)

7.2 Busse

■ Busspezifikation

■ Bussignale

■ Leitungsbündel (bei System- oder Prozessorbus)

■ Steuerbus: Beispiele

- **Memory write (W):** Bewirkt das Schreiben der Daten auf dem Datenbus an den adressierten Ort
- **Memory read (R):** Bewirkt das Daten des adressierten Orts auf den Bus gelegt werden
- **I/O write:** Bewirkt, dass Daten über den adressierten I/O Port ausgegeben werden
- **I/O read:** Bewirkt, dass Daten vom adressierten I/O Port auf den Bus gelegt werden
- **Transfer ACK:** Zeigt an, dass Daten vom Bus übernommen oder auf den Bus gegeben worden sind

7.2 Busse

■ Busspezifikation

■ Bussignale

■ Leitungsbündel (bei System- oder Prozessorbus)

■ Steuerbus: Beispiele

- **Bus request:** zeigt an, dass eine Komponente auf den Bus zugreifen möchte
- **Bus grant:** zeigt an, dass ein anforderndes Modul die Steuerung über den Bus erhält
- **Interrupt request:** zeigt an, dass eine Interrupt-Anforderung anliegt
- **Interrupt ACK:** zeigt an, dass einer Interrupt-Anforderung stattgegeben wird
- **Clock (Takt):** Synchronisation von Operationen
- **Reset:** Initialisierung der Komponenten

7.2 Busse

■ Busspezifikation

- Bussignale
- Leitungsbündel (bei System- oder Prozessorbus)
- Steuerbus
 - Signalfluss
 - unidirektional: eine Leitung wirkt als Eingabe oder als Ausgang
 - bidirektional: eine Leitung wirkt einmal als Eingang und einmal als Ausgang;

■ Versorgungsbus

7.2 Busse

■ Busspezifikation

■ Bussignale

■ Zustände von Steuerleitung

- Aktiver Zustand: Signal ist gesetzt, ihre Funktion ist wirksam
- Inaktiver Zustand: Signal ist nicht gesetzt, ihre Funktion ist ohne Wirkung

■ 1-aktive Signale

- haben, wenn sie im Aktivzustand sind, den logischen Wert 1;

■ 0-aktive Signale (Signalname überstrichen, vorangestellter Stern)

- haben, wenn sie im aktiven Zustand sind, den logischen Wert 0,

Bus Grant
* Bus Grant

■ einige Signale (z.B. das 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;

7.2 Busse

■ Busspezifikation

■ Bussignale

■ Zustände von Steuerleitung

■ Tri-State-Logik

- bei aktiven Bausteinen, d.h. bei Bausteinen mit Masterfunktionen sind die meisten Signalausgänge mit Tri-State-Logik versehen;
- kann neben den beiden logischen Zuständen 0 und 1 einen hochohmigen Zustand (high-impedence state) annehmen;
- Tri-State-Logik ermöglicht dem Prozessor, sich vom Systembus abzukoppeln. so dass ihn ein anderer Master übernehmen kann;

■ Open-collector-Verhalten von Signalausgängen

- Open-collector-Ausgänge bilden, wenn sie miteinander verbunden (verdrahtet werden)
- bei 1-aktiven Signalen eine UND-Verknüpfung (verdrahtetes UND)
- bei 0-aktiven Signalen eine ODER-Verknüpfung (verdrahtetes ODER, wired-or)

7.2 Busse

■ Busspezifikation

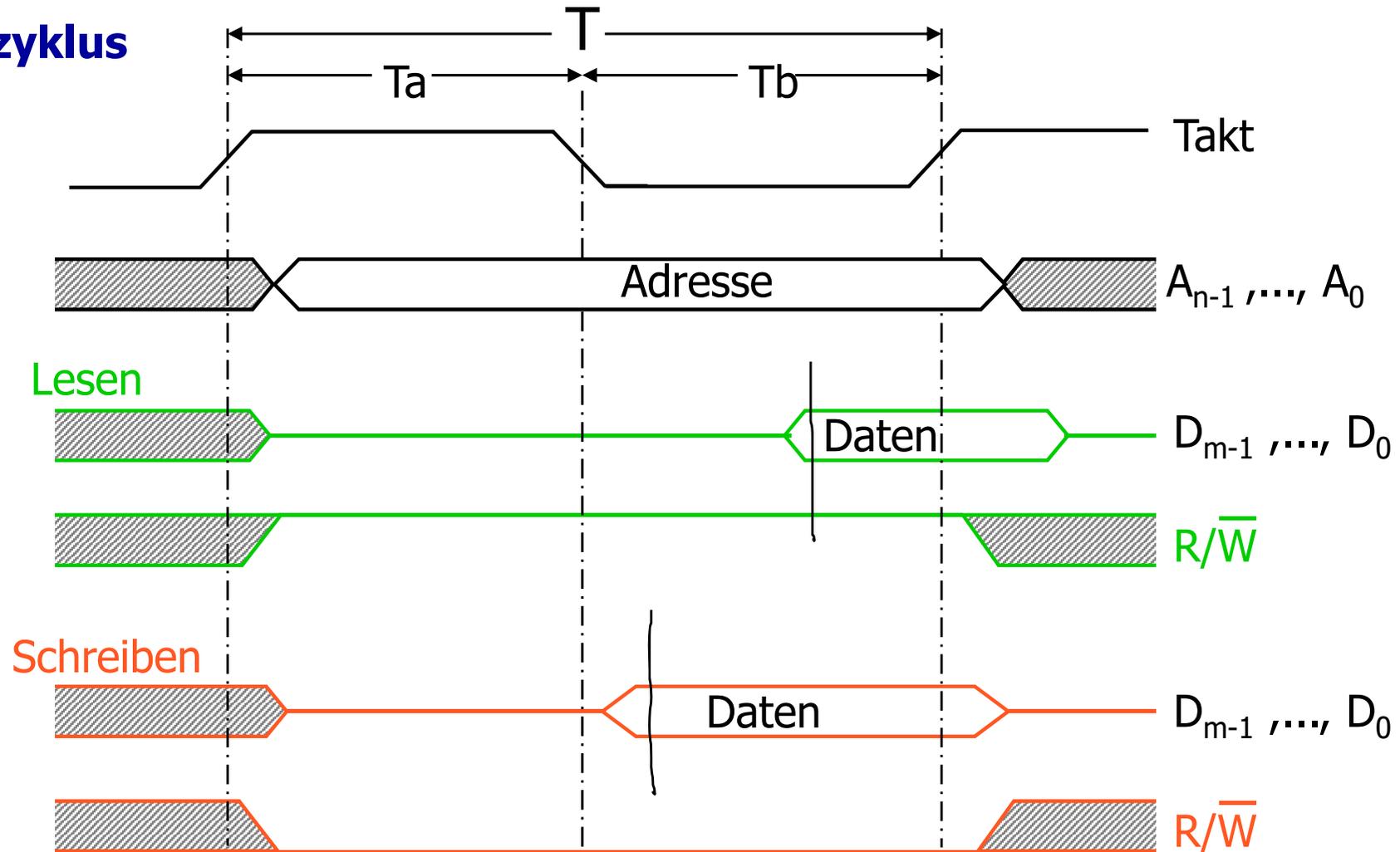
■ Bussignale: Zeitverhalten

■ Synchroner Bus

- Das Auftreten von Ereignissen auf dem Bus wird durch einen Takt festgelegt
- Alle Vorgänge synchron zum Takt nach einem starren Muster ablaufen → synchroner Systembus
- Übergabe und Übernahme der Daten geschieht zu festgelegten Taktflanken
- Nachteil: Alle am Bus angeschlossenen Komponenten müssen strenge Zeitvorgaben erfüllen
 - Langsamste Komponente bestimmt die zulässige Geschwindigkeit des Busses
 - Oder: der Bus schließt den Einsatz von „schnellen“ Komponenten aus (z.B. keine schnelle Speicher)

Zeitverhalten: synchroner Bus

T: Buszyklus



7.2 Busse

■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Synchroner Bus

- Adresse wird zu Beginn des Buszyklus (T_a) auf den Adressbus gelegt
- Auswahl der Übertragungsrichtung durch R/*W

- Lesen ($R/*W = 1$):
 - Speicher (oder andere Systemkomponente) liefert ihre Daten gegen Ende des Buszyklus (T_b)
 - Übernahme der Daten in den Prozessor mit der steigenden Flanke des Systemtakts

- Schreiben ($R/*W = 0$):
 - Prozessor legt die Daten zu Beginn der zweiten Takthälfte (T_b) auf den Systembus
 - Übernahme der Daten in den Speicher durch die steigende Flanke des Systemtakts am Ende der zweiten Takthälfte (T_b)

7.2 Busse

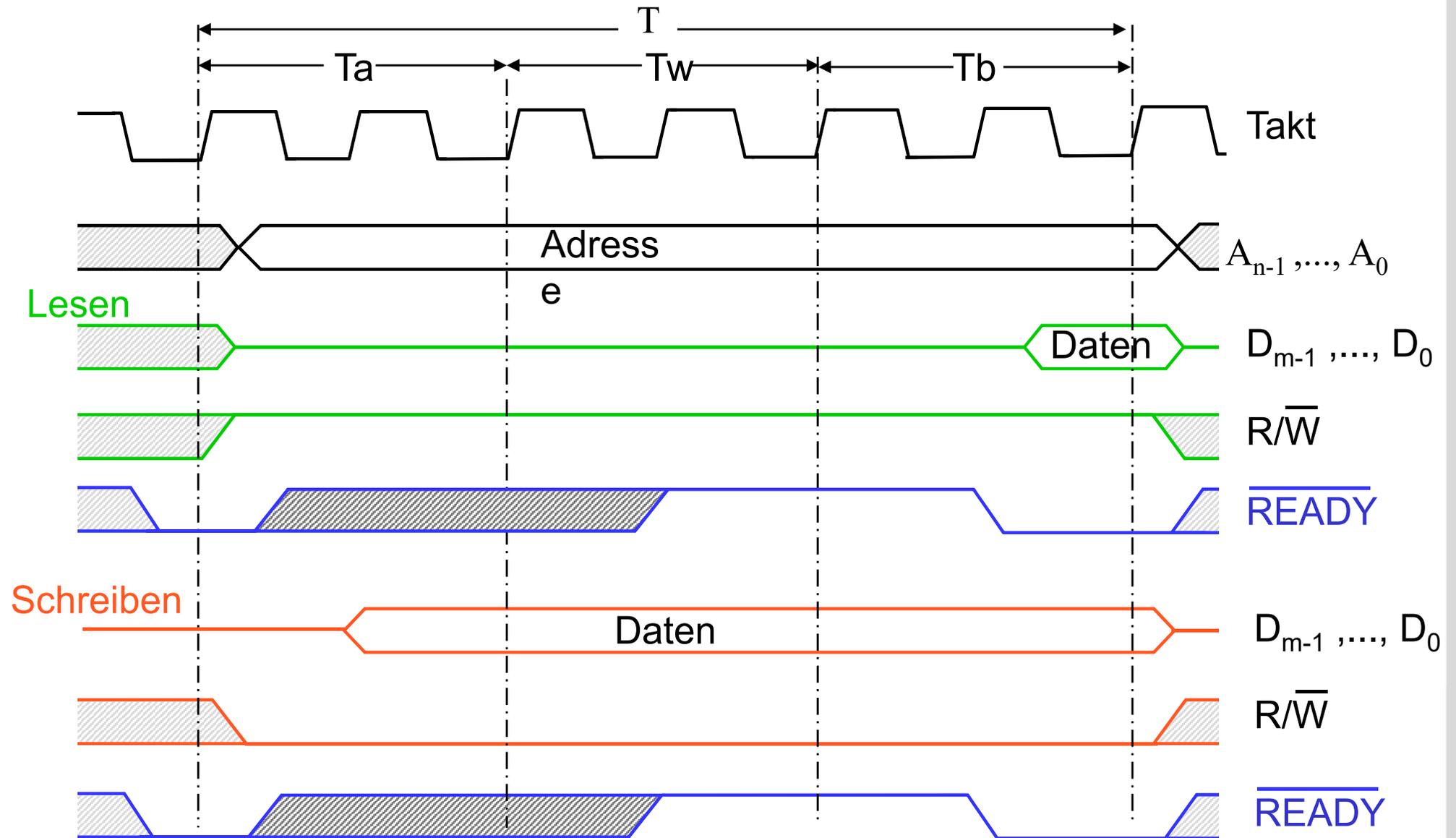
■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Semi-Synchroner Bus

- Bei Prozessoren mit höheren Taktfrequenzen
 - Schreib- bzw. Lesezugriffe benötigen mehrere Taktzyklen
 - Steuereingang (*READY) zur Synchronisation der Buszugriffe durch Einführung von Wartezyklen (wait states)
 - Unterschiedlich schnelle Speicher und Geräte können individuell bedient werden

Semi-synchroner Bus



7.2 Busse

■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Semi-Synchroner Bus

■ Timing:

■ Adresse zu Beginn des Buszyklus (T_a) auf den Adressbus

■ Buszyklus wird nur abgeschlossen, wenn rechtzeitig vor Ende von der Takthälfte (T_b) $*READY = 0$ ist

■ Ansonsten werden solange Wartezyklen (T_w , Dauer: z.B. halbe Buszykluslänge) eingefügt, bis $*READY = 0$ wird

■ Bezeichnung: Semi-synchrone Busse oder: Synchrone Busse mit Wartezyklen

7.2 Busse

■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Semi-Synchroner Bus

- Bus ist immer noch synchron (streng am Takt orientiert), die Dauer eines Buszyklus ist jedoch nicht mehr fest, sondern in Vielfachen von Taktzyklen variierbar → semi-synchroner Systembus
- Höherer Steueraufwand als synchroner Systembus
- Zeitverhalten ist auf verschieden schnelle Bausteine anpassbar
- Sind nur ausreichend schnelle Bausteine im System, kann *READY z.B. fest auf 0 gelegt werden

7.2 Busse

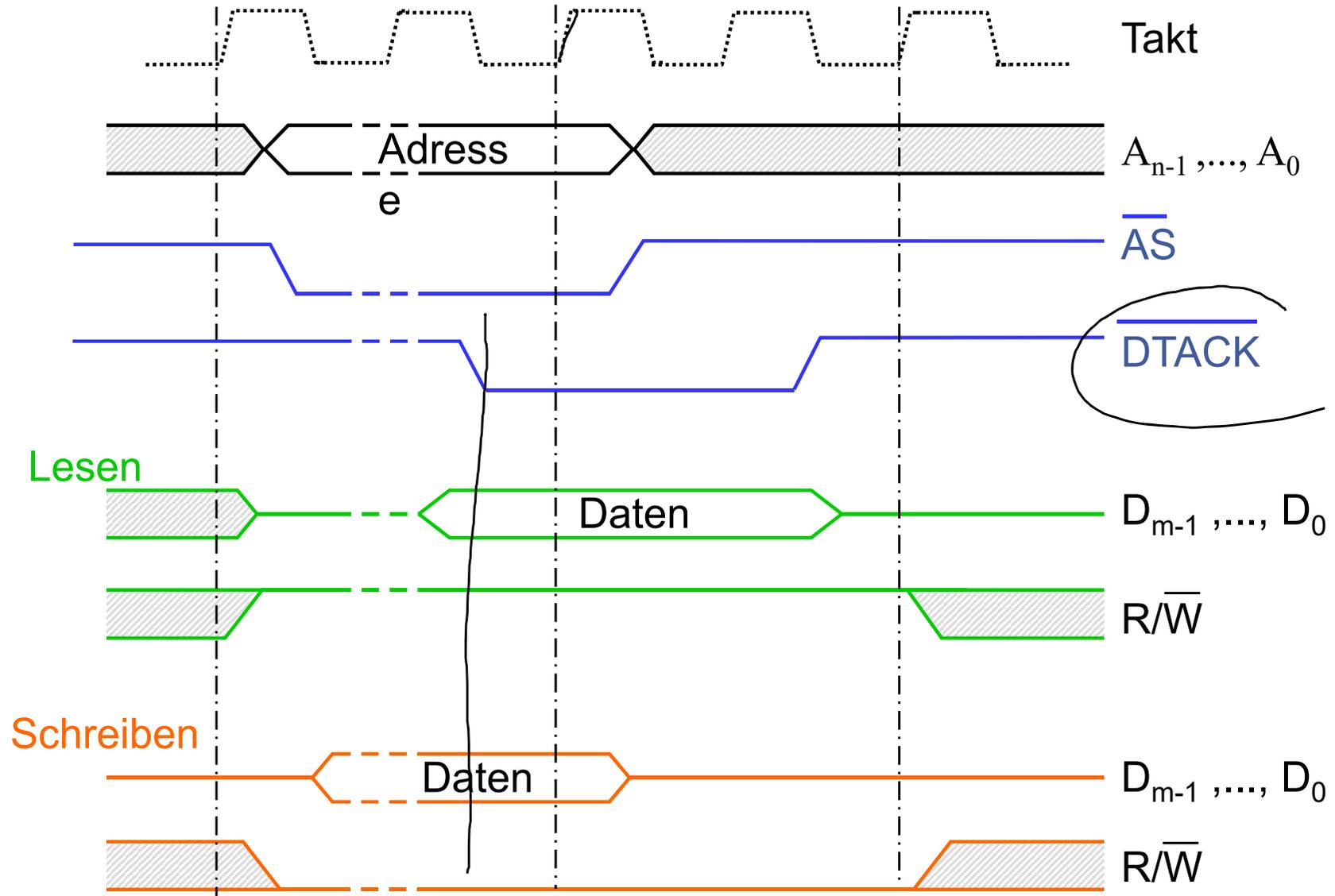
■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Asynchroner Bus

- Zeitliche Abläufe werden durch Handshake-Signale gesteuert
- Handshake-Signale:
 - ***AS** (Address Strobe) von CPU
 - ***DTACK** (Data Transfer Acknowledge) von Speicher/Komponente
- Vollständig asynchroner Übertragungsablauf, Anschluss von Komponenten mit fast beliebiger Zugriffszeit möglich
- Systemtakt spielt keine Rolle mehr für die Synchronisation der Übertragung (nur noch für die Synchronisation der Signale: synchrones Steuerwerk)

Zeitverhalten asynchroner Bus



7.2 Busse

■ Busspezifikation

■ Bussignale: Zeitverhalten

■ Asynchroner Bus

- Mit $*AS = 0$ (Address Strobe) zeigt die CPU an, dass sie eine gültige Adresse auf den Adressbus gelegt hat
- Mit $*DTACK = 0$ zeigt der Speicher an, dass er die Daten zur Verfügung gestellt (Lesen) oder übernommen (Schreiben) hat
- Zwischen $*AS = 0$ und $*DTACK = 0$ kann eine beliebige Zeitspanne liegen
- Wird $*DTACK = 0$, so nimmt die CPU die Adresse wieder vom Adressbus und setzt $*AS$ wieder zu 1
- Daraufhin nimmt der Speicher das Datum vom Datenbus und setzt $*DTACK$ wieder zu 1

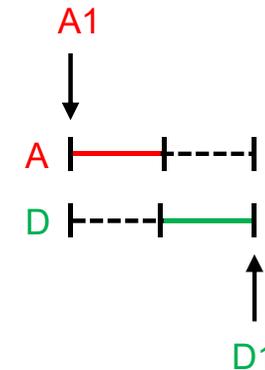
7.2 Busse

■ Busspezifikation

■ Busarten

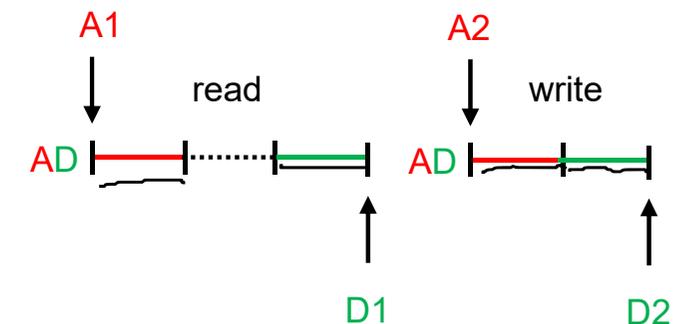
■ Split-Bus

- Getrennte Adress- und Datenleitungen
- Die beiden Teilbusse werden ausschließlich für die Übertragung von Adressen bzw. Daten benutzt



■ Mux-Bus

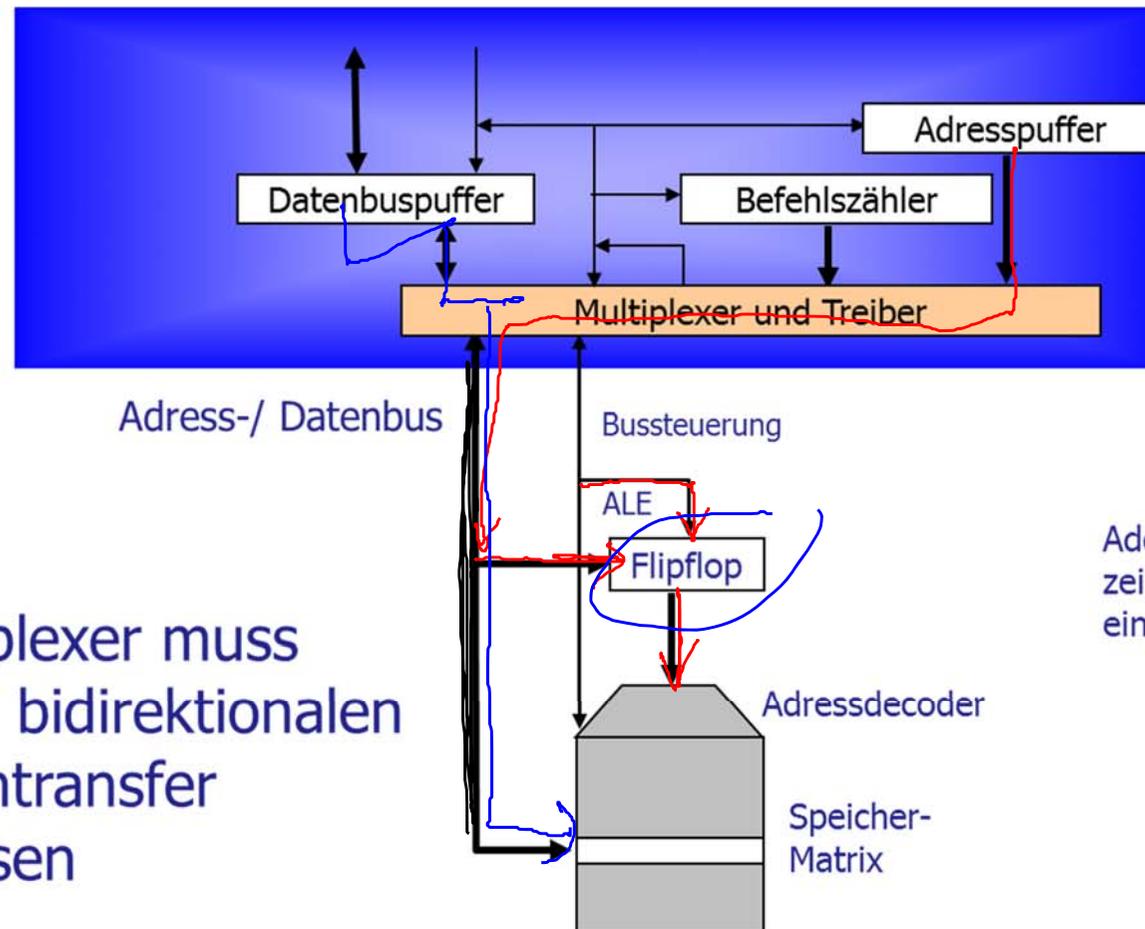
- Adress- und Datenleitungen sind zusammengefasst
- Die Adressen und Daten werden nacheinander im Multiplexbetrieb übertragen



7.2 Busse

■ Busspezifikation

- Busarten
- Multiplex-Bus-Schnittstelle



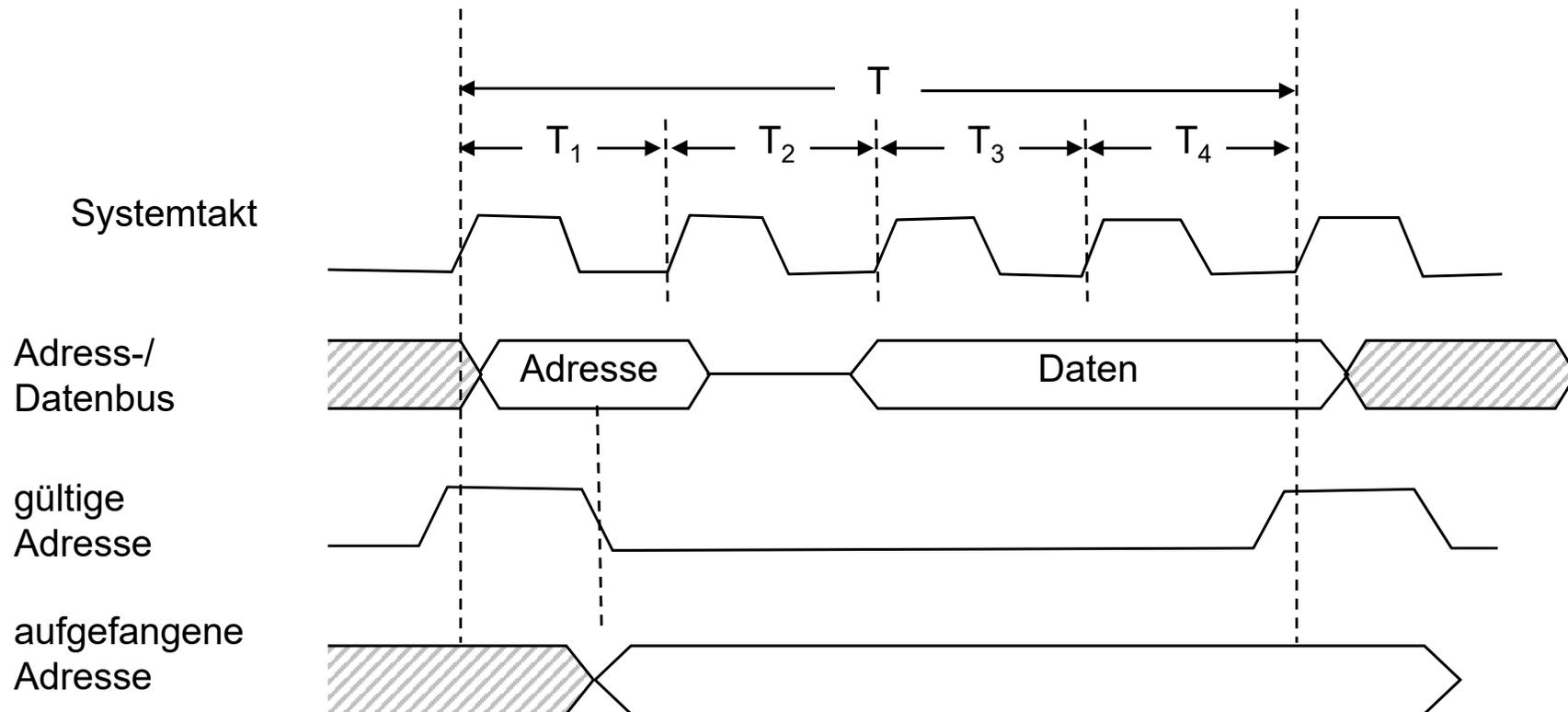
Multiplexer muss einen bidirektionalen Datentransfer zulassen

Address Latch Enable (ALE) zeigt das Vorliegen einer gültigen Adresse an

7.2 Busse

■ Busspezifikation

- Busarten
- Multiplex-Bus: Zeitverhalten



7.2 Busse

■ Busspezifikation

■ Busarten

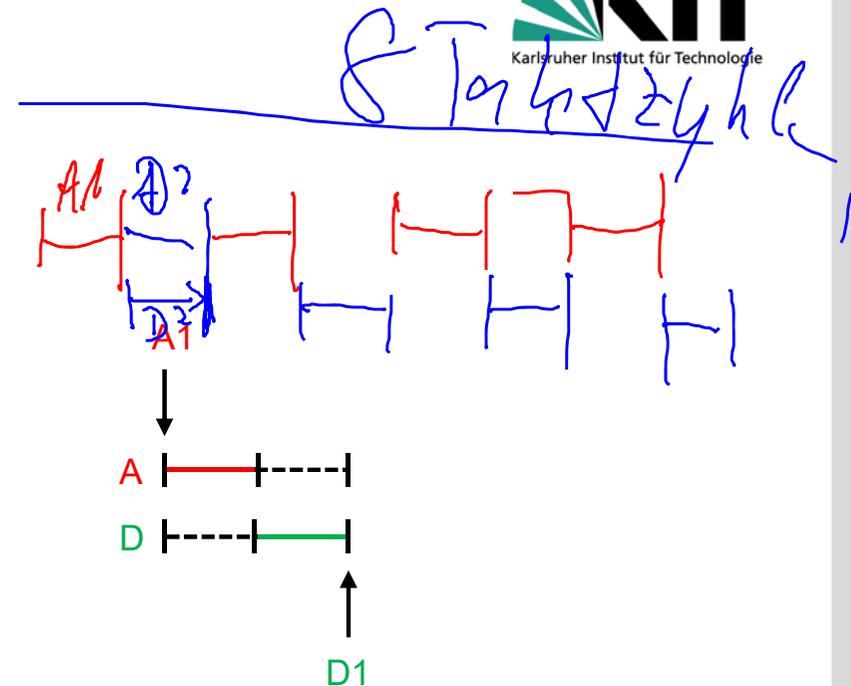
■ Multiplex-Bus: Zeitverhalten

- Gültige Adresse auf dem gemeinsamen Bus wird durch ein Signal angezeigt, z.B. *AS (Address Strobe) oder *VMA (Valid Memory Address) oder *ALE (Address Latch Enable)
- mit der aktiven Flanke dieses Signals (im Beispiel fallende Flanke) wird die Adresse in ein vor den Speicher geschaltetes Adress-Flipflop übernommen → stabile Adresse am Speicher
- Danach kann der Bus umgeschaltet werden und nun die Daten über die gleichen Leitungen geschickt werden

7.2 Busse

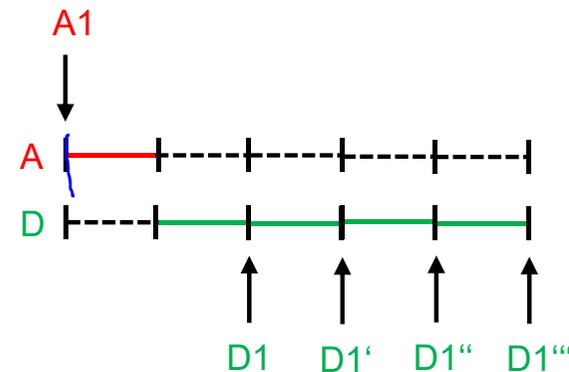
■ Busspezifikation

- Datentransport (Splitbus)
- Einzelbuszyklus
 - Führt eine Datenübertragung durch



■ Blockbuszyklus

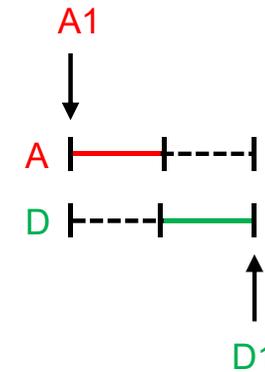
- Erweitert den Einzelbuszyklus um zusätzliche Datentransfers, benötigt aber nur einen Adresstransfer



7.2 Busse

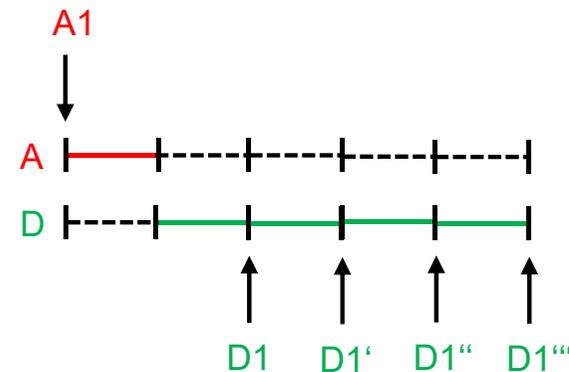
■ Busspezifikation

- **Datentransport (Splitbus)**
- **Einzelbuszyklus**
 - Führt eine Datenübertragung durch



■ Blockbuszyklus

- Erweitert den Einzelbuszyklus um zusätzliche Datentransfers, benötigt aber nur einen Adresstransfer



7.3 Systemaufbau

■ Organisation des Hauptspeichers

- lineare Liste von Speicherworten
- Aufgebaut aus Speicherbausteinen
 - Zugriffszeit hängt allein von der Art der verwendeten Speicherbausteine ab
 - Die Breite des Arbeitsspeichers entspricht i.A. der Breite des Datenbusses (8, 16, 32, 64 Bit)
 - Dies entspricht der maximalen Datenmenge, auf die in einem Buszyklus zugegriffen werden kann

- Die maximale Kapazität des Hauptspeichers ist durch die Breite des Adressbusses gegeben
 - 8-Bit Prozessoren mit 16-Bit Adressbus: 64 kBytes
 - 16-Bit Prozessoren mit 24-Bit Adressbus: 16 MBytes
 - 32-Bit Prozessoren mit 32-Bit Adressbus: 4 GBytes
 - 64-Bit Prozessoren mit 64-Bit Adressbus: 16 EBytes

7.3 Systemaufbau

■ Organisation des Hauptspeichers

■ Modularer Speicheraufbau

- Arbeitsspeicher wird oft auf mehrere Steckkarten verteilt, die über eine Grundplatine mit dem Systembus verbunden sind → Erweiterbarkeit

