

Modul Informatik-II

Kurs Informatik-3: Teil-4

www.engineering.zhaw.ch/de/engineering/studium/bachelor/informatik/studium-zurich.html

Prof. Dr. Olaf Stern
Leiter Studiengang Informatik
+41 58 934 82 51
olaf.stern@zhaw.ch

Lernziele: (Allgemein)

- Die Studierenden kennen die *grundlegende Architektur von Rechnern* und die wichtigsten *Architekturelemente*.
- Sie sind vertraut mit der *elementaren Arbeitsweise eines Computers* und der *hardwarenahen Programmierung*. Sie können diese an einfachen Beispiel erläutern.
- Die Studierenden kennen die grundsätzlichen Aufgaben eines *Betriebssystems*. Sie können die *typischen* Verfahren und *Algorithmen*, die bei der *Entwicklung* von *Betriebssystemen* zur Anwendung gelangen, beschreiben.

Lernziele: (Allgemein)

- Die Kurse der Module Informatik I und Informatik II (der Modulgruppen "Grundlagen der Informatik I+II") vermitteln den Studierenden die *Grundlagen der Informatik, die jede / jeder Studierende unabhängig von der Wahl der Wahlpflichtmodule im Fachstudium erlangen sollte.*
- Die vermittelten Grundlagen *werden in den Modulen im Fachstudium vorausgesetzt.*

Lernziele: Spezifisch Teil-4

- Die Studierenden können verschiedene *Grundtypen* von *Speichern* und ihre *Unterschiede* angeben.
- Sie können den *grundlegenden Aufbau* und die *Funktionsweise eines Speichers* (inklusive *Speicherhierarchie*) zusammenfassen und sind insbesondere vertraut mit der *DRAM*-Technik.

Themenüberblick Teil-4

Technische Informatik / Rechnerarchitektur

- Einführung / Übersicht
- Grundlegende Rechnerarchitektur
- Prozessoren
- Befehle – die „Wörter“ des Rechners
- „Mini-Power-PC“
- **Speicher**
 - **Speicherarten und Speicheraufbau**
 - **Speicherhierarchie**
 - Cache (Puffer, Zwischenspeicher)
- „Mini-Power-PC“ (Fortsetzung)

Lerninhalte Teil-4

- **Speicher**
 - **Speicherklassifizierung**
 - **RAM**
 - SRAM
 - DRAM
 - **ROM**
 - Verschiedene Typen
 - **Arbeitsspeicher**
 - **Speicherzugriff**
 - **Speicherhierarchie**

Architekturelemente - Speicher

Allgemein

- Grundsätzlich kann Speicher bezüglich vieler Eigenschaften klassifiziert bzw. unterschieden werden:
 - Zugriffsarten: direkt, sequentiell, assoziativ / parallel
 - Wieder-Beschreibbarkeit
 - Zugriffszeiten
 - Kosten / Grösse / Dichte
 - Einsatz: Arbeitsspeicher, Massenspeicher, Archiv, ...
 - Zuverlässigkeit
 - ...

neben einer Vielzahl weiterer physikalischer / elektrischer Parameter.

Architekturelemente - Speicher

Allgemein

- Die häufigsten allgemeinen Bezeichnungen sind:

a) **RAM** – *Random Access Memory*
(deutsch: *Direktzugriffsspeicher*)

und

b) **ROM** – *Read Only Memory*
(deutsch: *Festwertspeicher*)

Architekturelemente - Speicher

RAM – Random Access Memory

- Auf den Inhalt (Information) einer einzelnen Speicherzelle kann **direkt** - ohne andere Zellen zuvor zu bearbeiten - **lesend** und **schreibend** zugegriffen werden.
- Dieser Speicher wird auch als „*Speicher mit **wahlfreiem Zugriff***“ bezeichnet.

[Speicher ohne **wahlfreiem** Zugriff, z. B. Tapes (Bandlaufwerke), weisen sehr grosse Zugriffszeiten auf und sind daher nur als sekundäre Speicher nutzbar; in der Regel dafür aber sehr preiswert.]

Architekturelemente - Speicher

ROM – Read Only Memory

- Der Inhalt (Information) einer einzelnen Speicherzelle, ist im normalen Betrieb **nur lesbar**; die Speicherzelle kann **nicht beschrieben** werden⁺.
- Ursprünglich wurde der Inhalt eines *ROM*-Speichers zum Fertigungszeitpunkt festgelegt („*programmiert*“), z. B. durch eine Maske während des Herstellungsprozesses.

Einsatz: z. B. BIOS, Spezialprozessoren

⁺Gilt heute so absolut nicht mehr (z. B. *Flash-EEPROM*).

Architekturelemente - Speicher

RAM vs. ROM

- Die Begriffe *RAM* und *ROM* sind insofern irritierend bzw. nicht vollständig, da zugleich
 - *RAM*-Speicher in der Regel auch beliebig wiederholt beschrieben werden kann
 - und
 - auf *ROM*-Speicher in der Regel auch wahlfrei zugegriffen werden kann!

Architekturelemente - Speicher

RAM – Random Access Memory

- **RAM** wird weiter unterteilt in:
 - **DRAM** – **Dynamisches RAM**:
 - Der **Speicherinhalt** geht **verloren**, wenn
 - a) die **Versorgungsspannung abgeschaltet** oder
 - b) der **Speicherinhalt nicht periodisch aufgefrischt** wird: „**Refresh**“ (ca. alle 32 - 64ms erforderlich).
 - Wesentlicher **Vorteil** ist die **preiswerte Herstellung** und der **sehr geringe Platzbedarf** (=> **sehr hohe Dichte**).
 - **Einsatz: In Arbeitsspeicher (gross); langsam!**

[Realisierung: Über einen einzelnen Kondensator bzw. Transistor als Ladungsträger => Details siehe Kurs Elektronik.]

Architekturelemente - Speicher

RAM – Random Access Memory

- **RAM** wird weiter unterteilt in: (Forts.)

- **SRAM** – *Statisches RAM*:

- Der **Speicherinhalt** geht **verloren**, wenn
 - a) die **Versorgungsspannung abgeschaltet** wird.

Diese kann z. B. mit Hilfe einer Batterie jahrelang aufrecht erhalten werden; ein **Refresh** ist **nicht erforderlich**.

- Ein **weiterer Vorteil** sind die sehr **kurzen Zugriffszeiten**, jedoch benötigt eine **SRAM-Speicherzelle** deutlich **mehr Platz, Strom** und ist **wesentlich teurer** in der **Herstellung**.
 - **Einsatz:** In Registern und **Cache-Speichern**; **sehr schnell**

HS 2013/14 [Realisierung: über *bistabile Kippstufen* => Details siehe Kurs Elektronik]

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **Masken-ROM**,
 - **PROM** (*Programmable ROM*),
 - **EPROM** (*Erasable Programmable ROM*),
 - **EEPROM** (*Electrically Erasable Programmable ROM*) und
 - **Flash-EEPROM** (auch kurz Flash-Speicher genannt).

Anmerkung: Ob bei den neueren Techniken noch von „**Read-Only**“ im engeren Sinne gesprochen werden kann, ist fraglich!

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **Masken-ROM**: 1x beim Fertigungszeitpunkt programmierbar
 - **Preisgünstig** (Massenherstellung)
 - **Nicht flexibel**

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **PROM**: 1x programmierbar (durch Überspannung)
 - **Flexibler**
 - **Beim Programmieren wird Metaldampf freigesetzt**
 - **Spezielle SW (und HW) erforderlich**

Anmerkungen: Ein **PROM** entspricht technisch einem **PLA**. **PROMs** werden heute nicht mehr verwendet (durch **EPROMs** ersetzt).

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **EPROM**: Mehrfach löschar mit UV-Licht
 - **Flexibel**
 - **Teuer, UV-Licht für Löschvorgang erforderlich**
 - **Löschvorgang langwierig (etliche Minuten)**

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **EEPROM**: Mehrfach löschbar mit Software
 - **Sehr flexibel**
 - **Löschvorgang sehr schnell** (wenige Sekunden)
 - **Beständigkeit der Daten von > 10 Jahren wird garantiert**
 - Anzahl Schreibzyklen begrenzt (ca. 1'000'000)
 - **Schreibvorgang langsam (einige ms)**

Anmerkung: Schreiben bedeutet immer, dass **zuvor gelöscht** werden muss!

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **Flash-EEPROM**: Mehrfach löschar mit Software
 - ... (alle Eigenschaften wie beim **EEPROM**), aber
 - **Kostengünstiger**
 - Die Anzahl der Schreibzyklen ist begrenzt (zwischen 5'000 und 1'000'000, je nach Architektur und Hersteller).
 - **Schreiben erfolgt schneller als bei EEPROM (einige μs).**
 - **Lesen und Schreiben erfolgt Gruppenweise (nicht wahlfrei pro Byte).**
 - **Geringerer Temperaturbereich (noch) gegenüber EEPROM.**

Architekturelemente - Speicher

ROM – Read Only Memory

- Inzwischen gibt es eine recht grosse Anzahl verschiedener Arten von **ROMs**. Die wichtigsten sind:
 - **Flash-EEPROM**: (Forts.)
 - Aktuelle **Architekturen** sind **NAND-** und **NOR-Flashs** (neben weiteren Unterscheidungen).
 - **NAND-Flashs** haben **höhere Kapazitäten**, weisen jedoch **langsamere Zugriffszeiten** auf.
 - Populär geworden sind **Flash-Speicher** durch **USB-Sticks**, **CompactDisks** und andere „**Speicherkarten**“ für diverse digitale Geräte (insb. mobile).
 - [Technologischer **Unterschied** zu den **SRAMs** ist, dass **zum Schreiben immer zuvor gelöscht werden muss!**]

Architekturelemente - Speicher

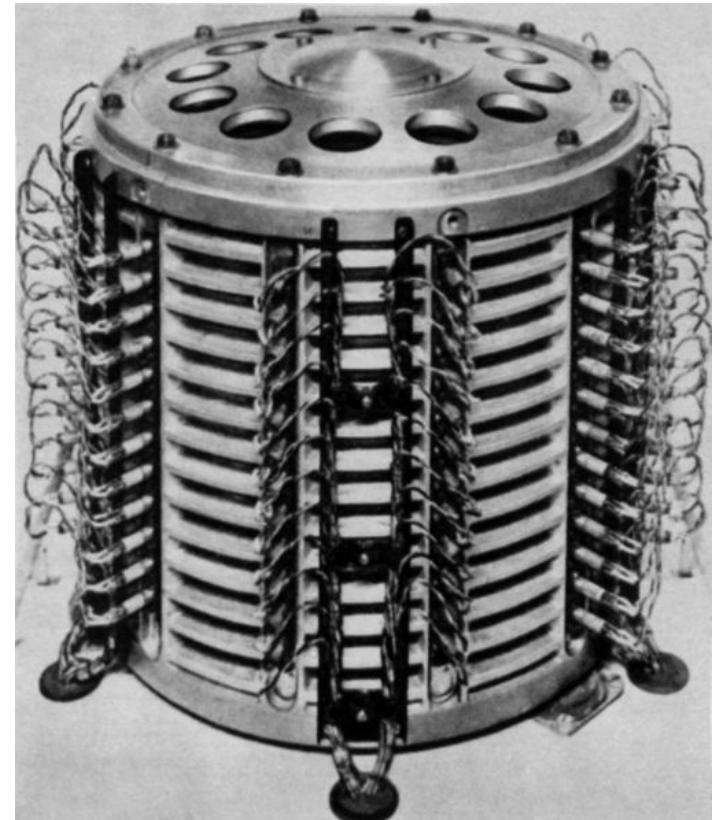
Arbeitsspeicher

- Auch als *Hauptspeicher* oder *main memory* bezeichnet
- Auf den Arbeitsspeicher wird immer direkt (*RAM*) zugegriffen – Zugriffsgeschwindigkeit!
- Historie:
 - Die ersten Rechner (Rechner der 1. Generation) besaßen nur wenige Register und keinen Arbeitsspeicher.

Architekturelemente - Speicher

Arbeitsspeicher

- **Historie: (Forts.)**
 - **Rechner der 2. Generation nutzten „Trommelspeicher“**
(Magnetischer Speicher, ähnlich einer Festplatte)



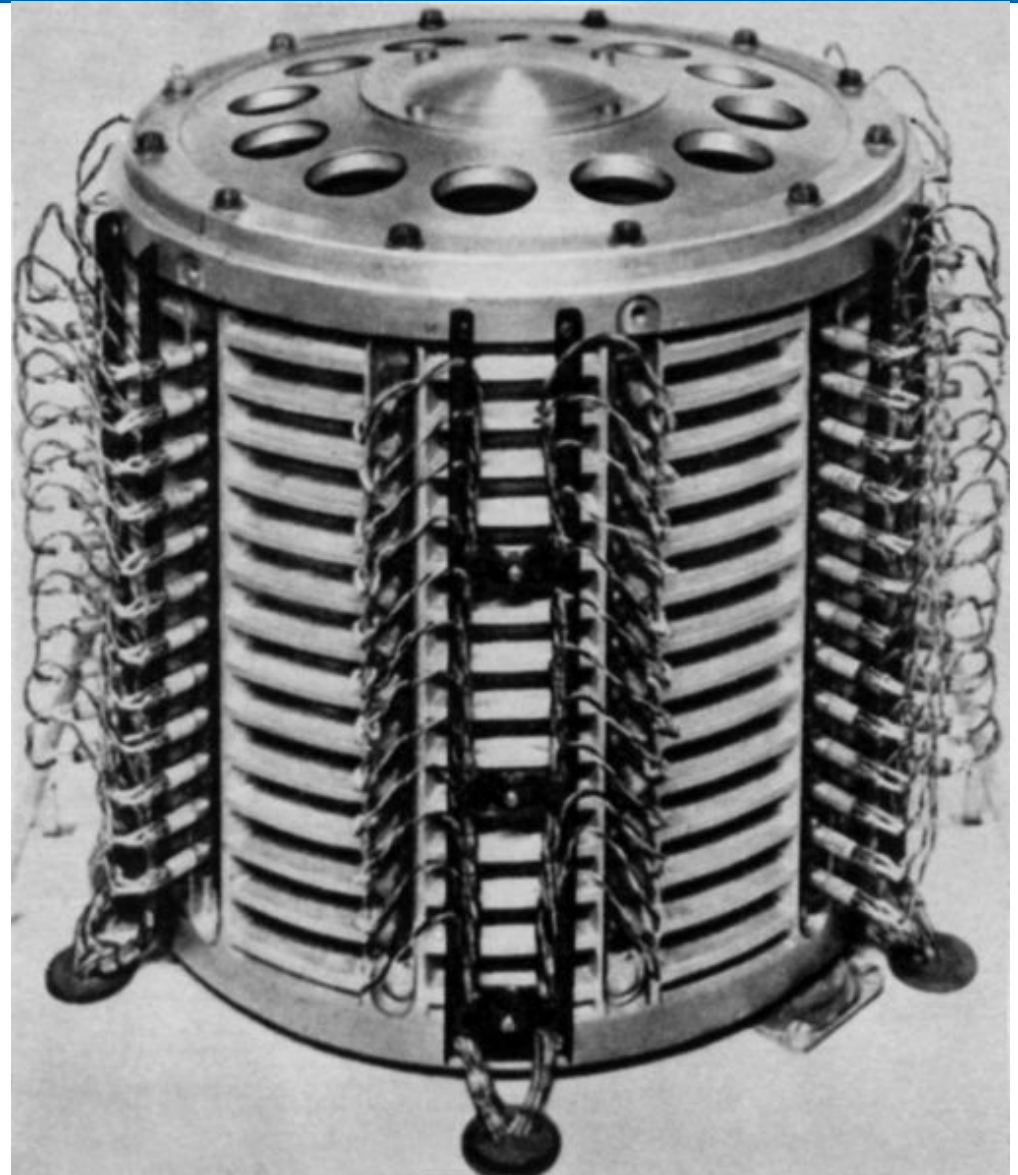
Quelle: Wikipedia „<http://de.wikipedia.org/wiki/Trommelspeichern>“, Stand Aug. 2011

Architekturelemente - Speicher

Arbeitsspeicher

„Trommelspeicher“

(Magnetischer Speicher,
ähnlich einer Festplatte)



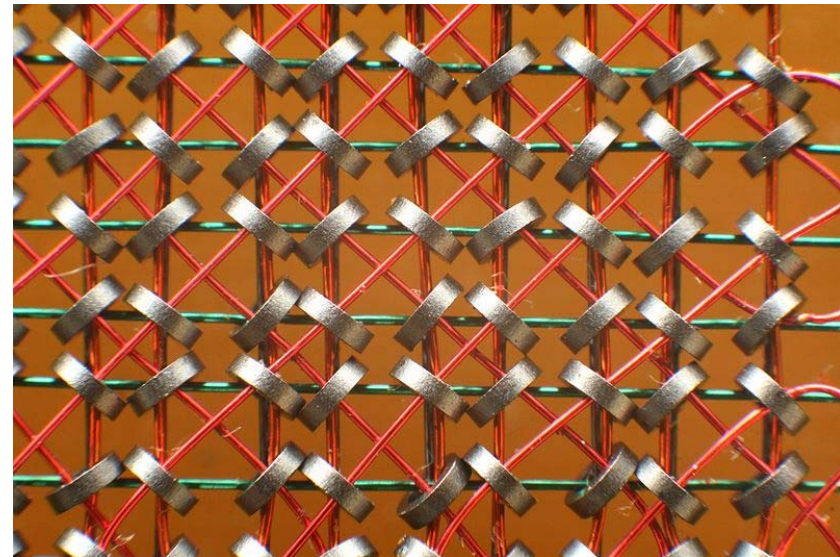
Quelle:

Wikipedia „<http://de.wikipedia.org/wiki/Trommelspeichern>“, Stand Aug. 2011

Architekturelemente - Speicher

Arbeitsspeicher

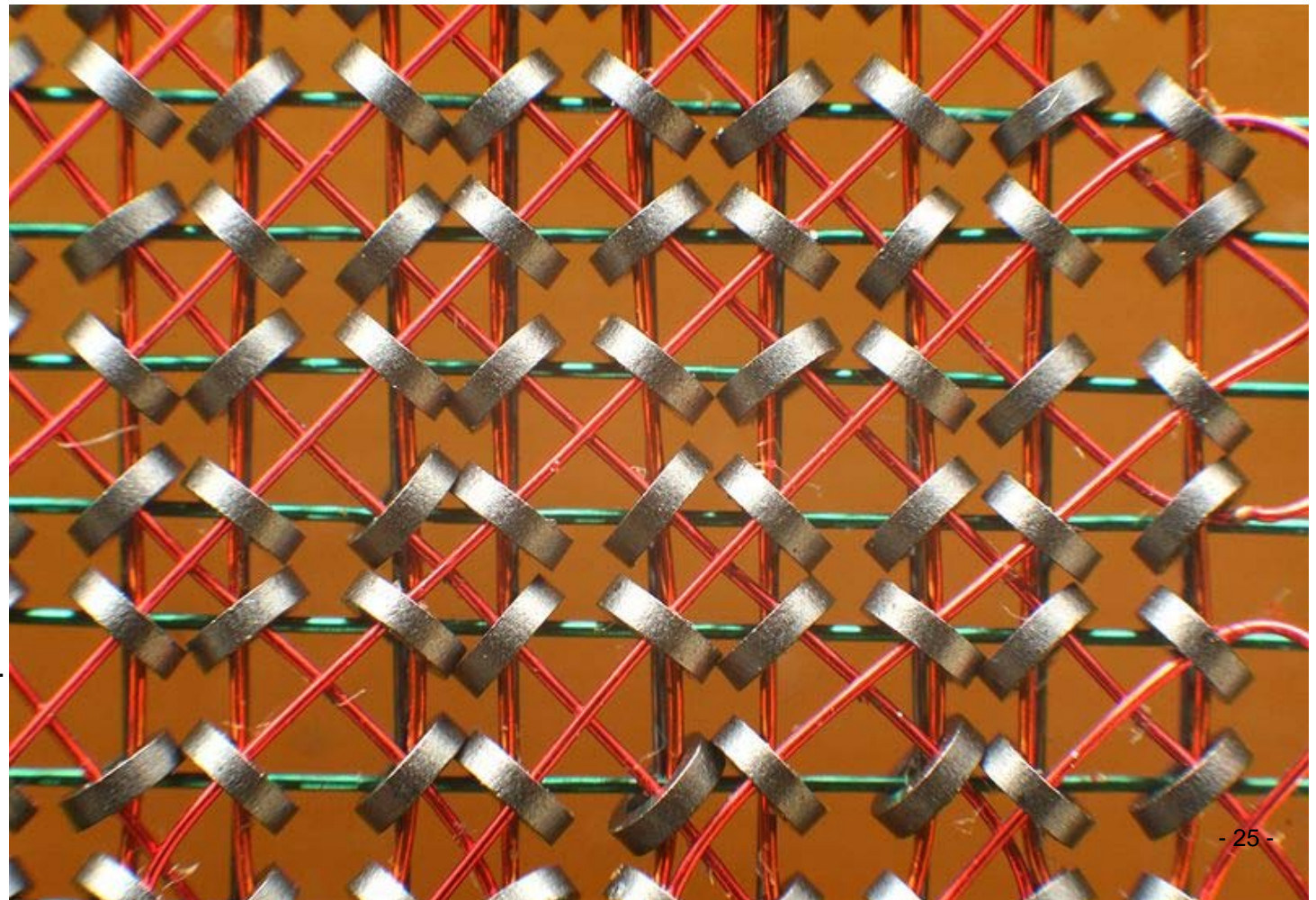
- **Historie: (Forts.)**
 - Rechner der 2. Generation nutzen „*Trommelspeicher*“ und später „*Kernspeicher*“: (magnetischer Speicher)



Quelle: Wikipedia „<http://de.wikipedia.org/wiki/Kernspeicher>“, Stand Aug. 2011

Architekturelemente - Speicher

Arbeitsspeicher „Kernspeicher“



Quelle:

Wikipedia „<http://de.wikipedia.org/wiki/Trommelspeichern>“,
Stand Aug. 2011

HS 2013/14

Zürcher Fachhochschule

Architekturelemente - Speicher

Arbeitsspeicher

■ Historie: (Forts.)

- Mit der Einführung von integrierten Schaltkreisen, *ICs*, (3. Generation) wurden der Arbeitsspeicher durch *ICs* ersetzt.
=> Die Zugriffszeiten lagen bereits bei nur ca. *100 ns*.
- Seit Ende der 70er-Jahre existieren **dynamische** Arbeitsspeicher (**DRAMs**).
- Ab Beginn der 80er-Jahre waren diese auf Grund der höheren Integrationsdichte kostengünstig herstellbar und trieben u. a. die Entwicklung der *PCs* voran.

Die Zugriffszeiten lagen seinerzeit häufig unter den Taktraten der Mikroprozessoren (wenige *MHz*)!

Architekturelemente - Speicher

Arbeitsspeicher

- Historie: (Forts.)

- Während bis heute die Kapazität der **DRAMs** **erheblich gesteigert** werden konnte (1970 Intel Typ 1103 mit **1 KiBit** auf heute über mehrere **GiBit**), gilt dieses **nicht für die Zugriffszeiten!**

Diese liegen weiterhin im **ns-Bereich**, heute bei ca. **5 - 20ns** für ein **einzelnes Bit/Byte/Wort** (erster Zugriff).

Mit verschiedenen **Verfeinerungen der Zugriffsmethoden**, z. B. „**Burst**“, konnten die **durchschnittlichen Zugriffszeiten** zwar auf **wenige ns gesenkt werden**, liegen damit dennoch erheblich **über den Taktraten aktueller Mikroprozessoren** (einige **GHz**)!

Architekturelemente - Speicher

Arbeitsspeicher

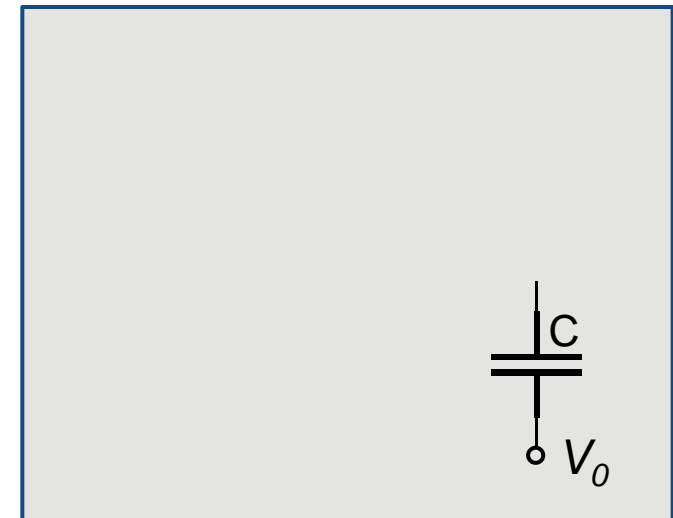
- **Historie: (Forts.)**
 - Heute wird der Arbeitsspeicher von Rechnern hauptsächlich mit **DRAMs** realisiert.
 - Für spezielle Anwendungen im mobilen Umfeld und insbesondere in eingebetteten Systemen (*Embedded Systems*) werden häufig **Flash-Speicher** verwendet.

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- **Aufbau eines *DRAM*-Speichers: Speicherzelle**
(vereinfachte schematische Darstellung)

DRAM-Speicherzelle besteht eigentlich
nur aus einem Kondensator, ...



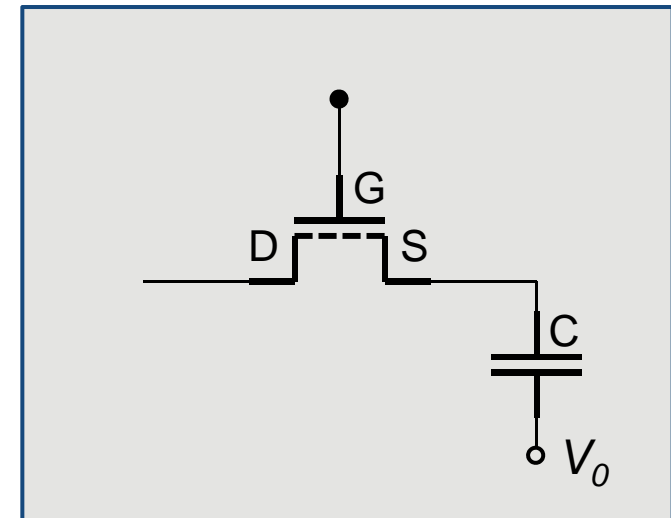
Anmerkung: V_0 kann $0V$ betragen, beträgt aber i. d. R. 50% der Betriebsspannung

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- **Aufbau eines *DRAM*-Speichers: *Speicherzelle*** (Forts.)
(vereinfachte schematische Darstellung)

***DRAM*-Speicherzelle** besteht **eigentlich** nur aus **einem Kondensator**, der über einen **Transistor geschaltet** wird ...



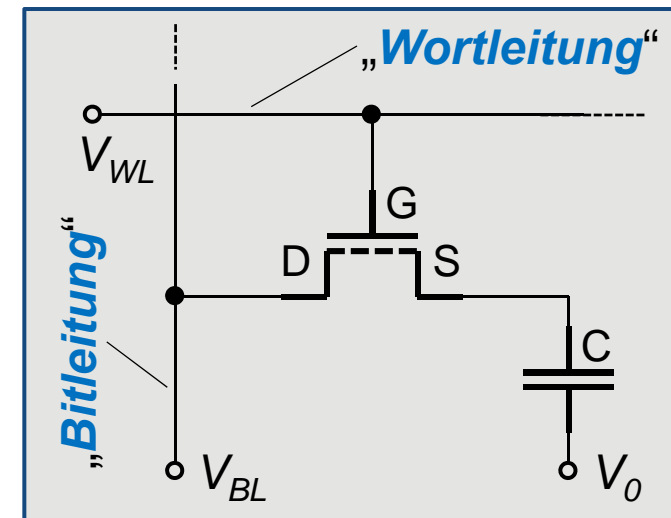
Anmerkung: V_0 kann $0V$ betragen, beträgt aber i. d. R. 50% der Betriebsspannung

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

DRAM-Speicherzelle besteht eigentlich nur aus einem **Kondensator**, der über einen **Transistor geschaltet** wird, und der zugehörigen **Ansteuerung** über eine sogenannte „**Bitleitung**“ und „**Wortleitung**“.



Anmerkung: V_0 kann $0V$ betragen, beträgt aber i. d. R. 50% der Betriebsspannung

Architekturelemente - Speicher

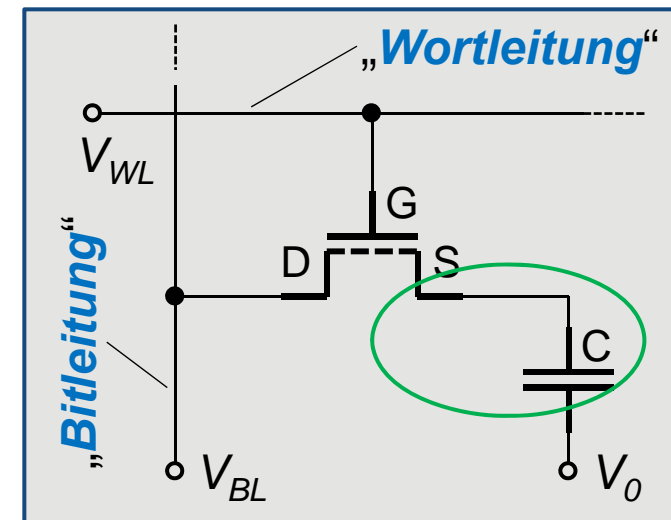
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

a) „*Ruhezustand*“:

Die *Wortleitung* befindet sich auf **niedrigem Potential**

=> Der **Transistor sperrt** und die **Ladung des Kondensators bleibt erhalten**



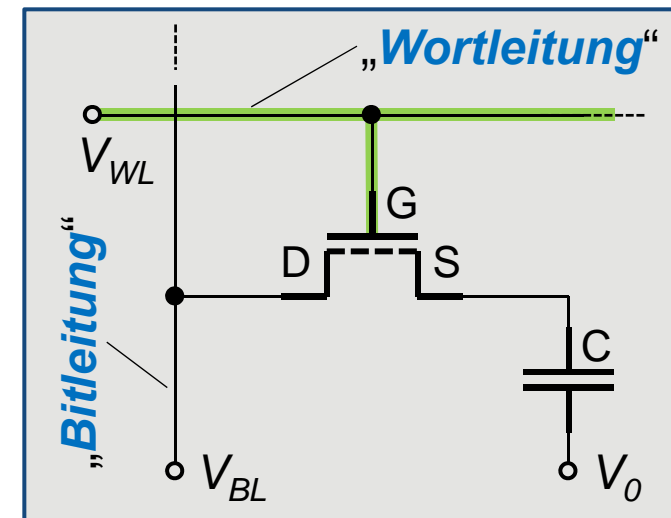
Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzelle* (Forts.)
(vereinfachte schematische Darstellung)

b) „Schreiben“:

Die *Wortleitung* befindet sich auf **hohem Potential**; ...



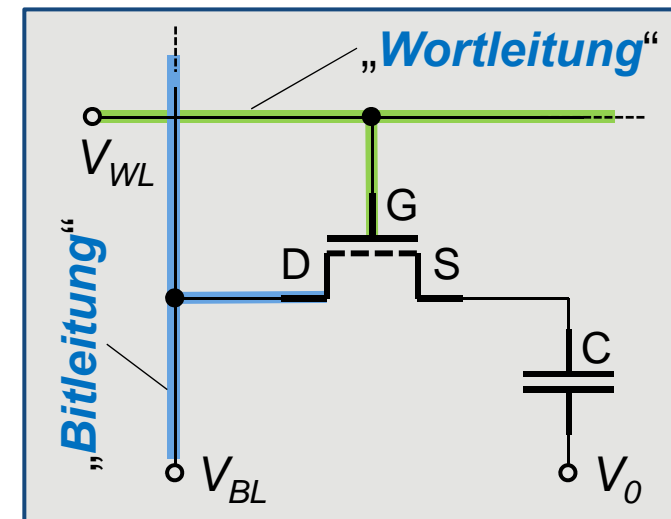
Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

b) „Schreiben“: (Forts.)

Die **Wortleitung** befindet sich auf **hohem Potential**; auf der „**Bitleitung**“ liegt der zu **speichernde Wert** an.



Architekturelemente - Speicher

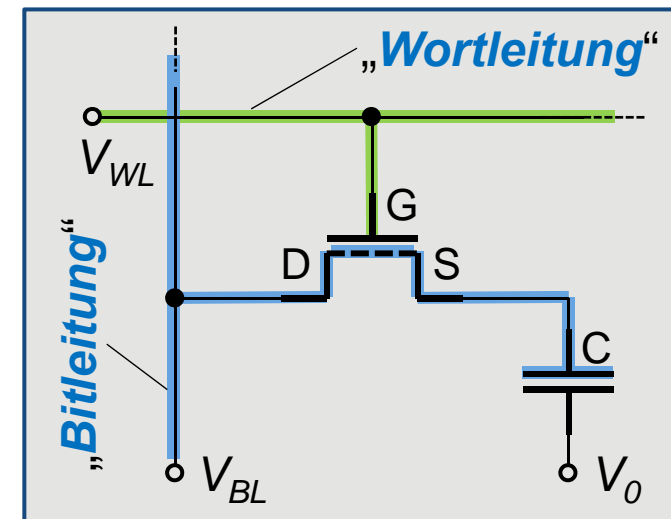
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzelle* (Forts.)
(vereinfachte schematische Darstellung)

b) „Schreiben“: (Forts.)

Die *Wortleitung* befindet sich auf **hohem Potential**; auf der „*Bitleitung*“ liegt der zu **speichernde Wert** an.

=> Der **Transistor** leitet und die **Ladung** auf der „*Bitleitung*“ wird übernommen, d. h. im **Kondensator C** gespeichert.



Architekturelemente - Speicher

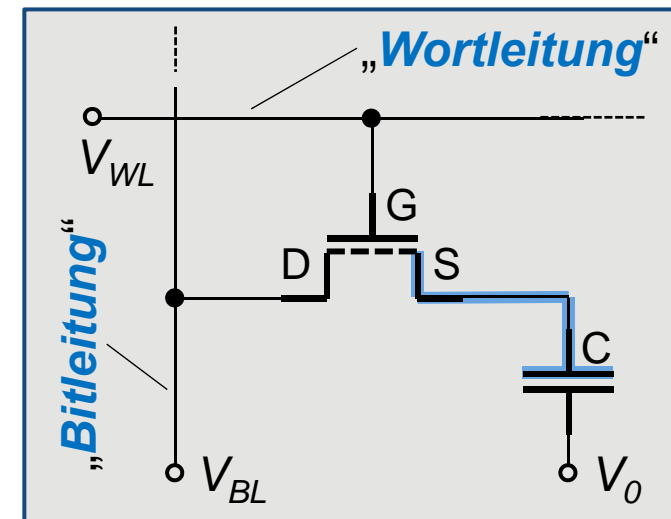
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzelle* (Forts.)
(vereinfachte schematische Darstellung)

b) „Schreiben“: (Forts.)

Die *Wortleitung* befindet sich auf **niedrigem Potential**; auf der „*Bitleitung*“ liegt ein unbestimmter Wert an.

=> Der **Transistor sperrt** und die **Ladung** im **Kondensator C** bleibt unabhängig vom **Zustand** der *Bitleitung* erhalten.
(bis auf **Leckströme** => *Refresh*)



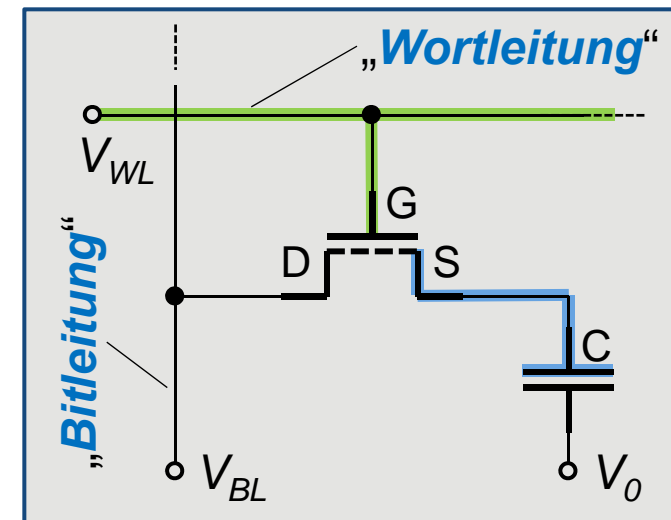
Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

b) „Lesen“:

Die **Wortleitung** befindet sich auf **hohem Potential**; der **Kondensator C** beinhaltet den **Wert der Speicherzelle**.



Architekturelemente - Speicher

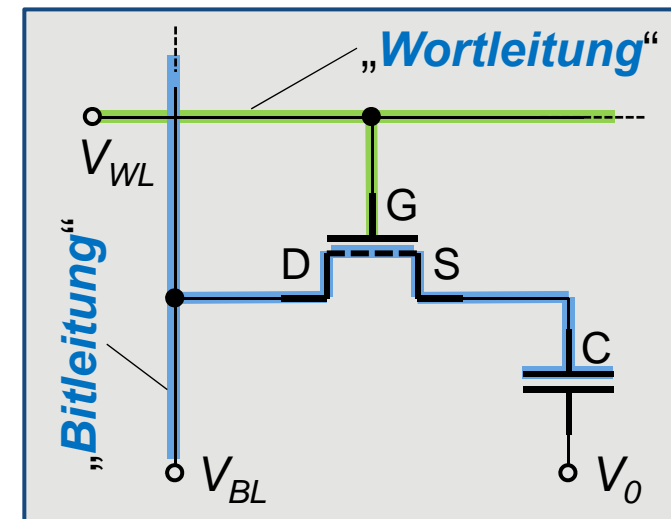
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

b) „Lesen“: (Forts.)

Die **Wortleitung** befindet sich auf **hohem Potential**; der **Kondensator C** beinhaltet den **Wert der Speicherzelle**.

=> Der **Transistor leitet** und die **Ladung** des **Kondensators C** wird auf die **Bitleitung** übernommen.



Architekturelemente - Speicher

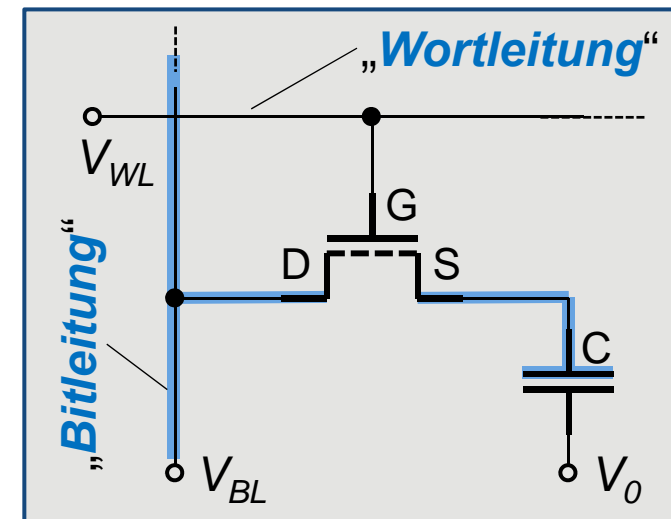
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: **Speicherzelle** (Forts.)
(vereinfachte schematische Darstellung)

b) „Lesen“: (Forts.)

Nachdem die **Wortleitung** deaktiviert ist
(befindet sich auf **niedrigem Potential**)

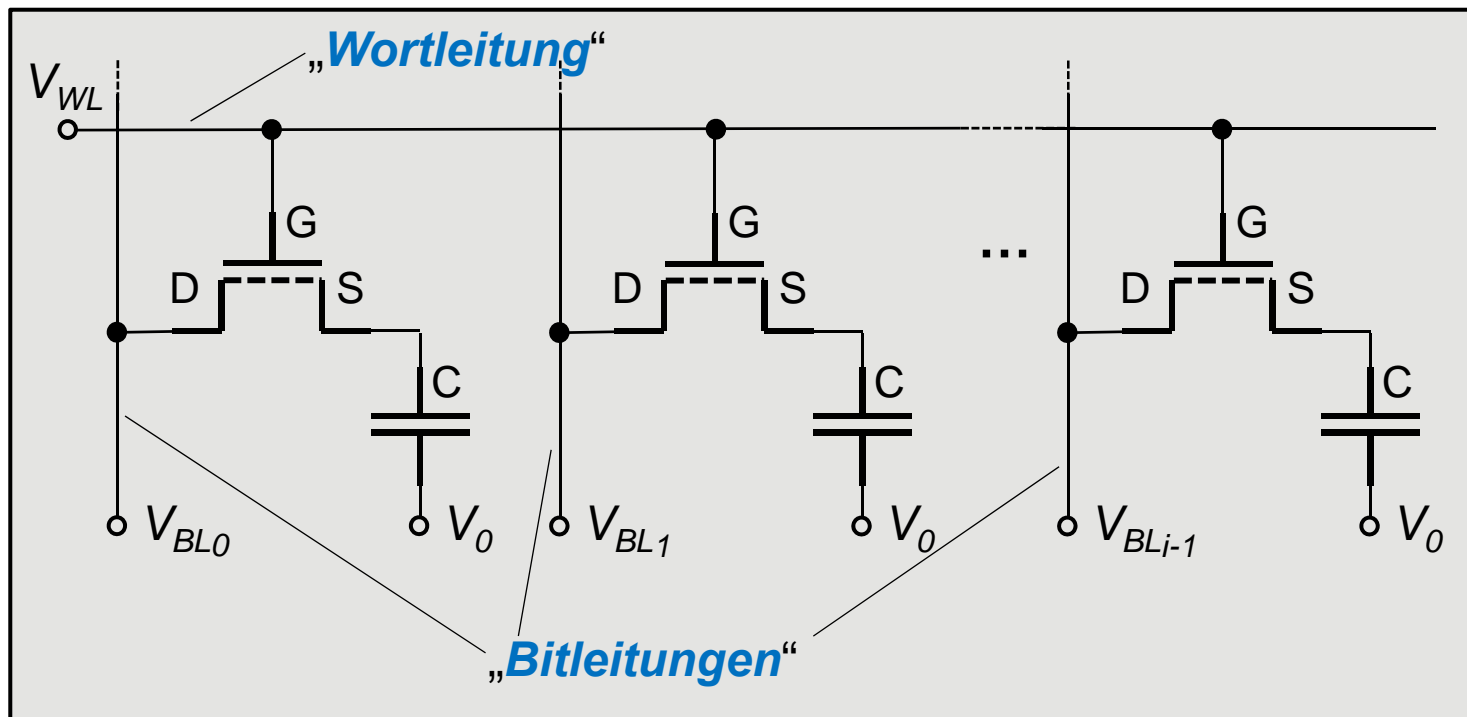
=> Der **Transistor sperrt** und die **Ladung**
auf der **Bitleitung** kann ausgelesen
werden (sie muss verstärkt werden!).



Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzeile*
(vereinfachte schematische Darstellung)



Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- **Aufbau eines *DRAM*-Speichers: *Speicherzeile*** (Forts.)
(vereinfachte schematische Darstellung)
 - **Alle** an eine *Wortleitung* angeschlossenen **Zellen** werden als „*Seite*“ (oder „*Page*“) bezeichnet.
 - Bei **Aktivierung** einer *Wortleitung* geben **alle** an die *Wortleitung* angeschlossenen **Zellen** ihren **Wert** (gespeicherten Inhalt) an die zugeordneten *Bitleitungen* weiter bzw. aus.

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzeile* (Forts.)
(vereinfachte schematische Darstellung)
 - ...
 - Die „*Aktivierung*“ der *Wortleitung*, d. h. mit hohem Potential belegen, dauert auf Grund der Kapazität und des Widerstandes der Leitung selber „*relativ*“ lange.
=> Verantwortlich für langsame Schaltdauer der *DRAMs*

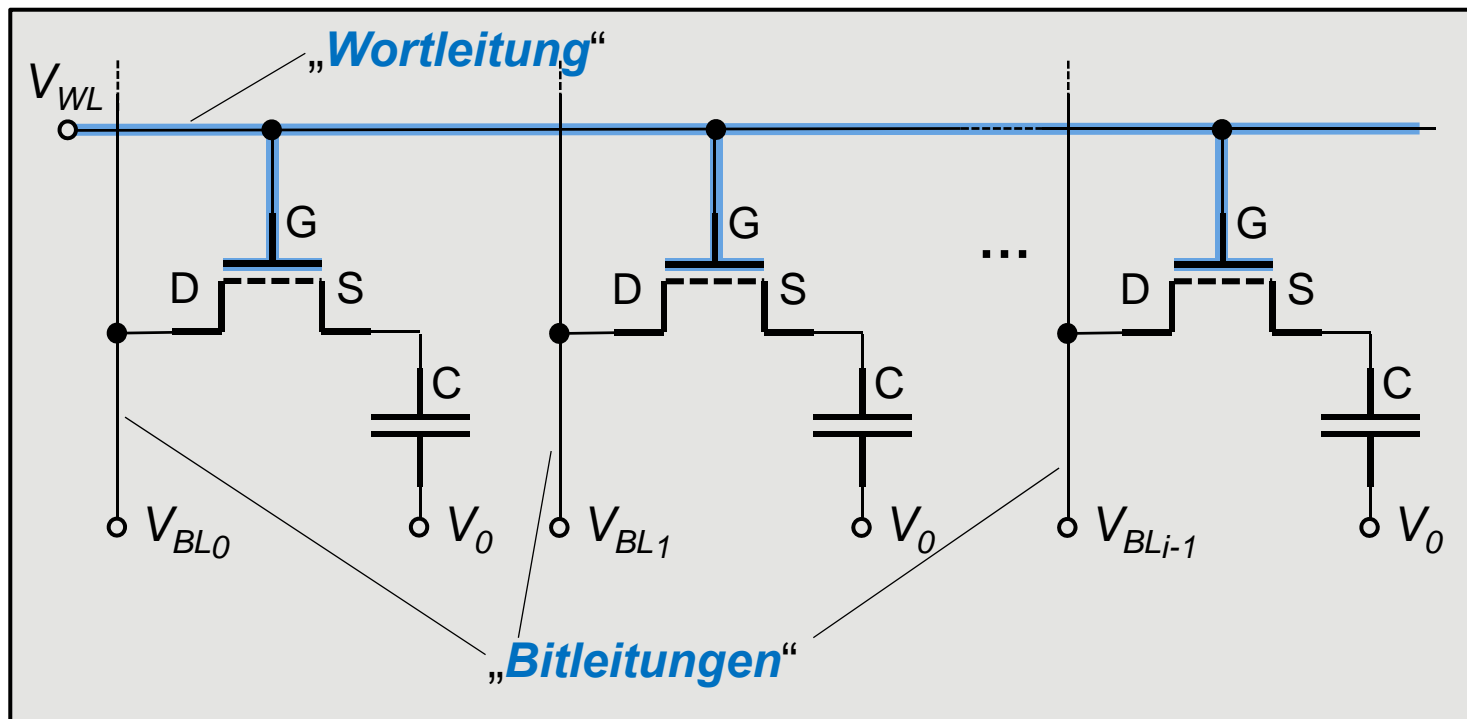
Physikalische Grenzen sind erreicht:

- **Kleinere Kapazitäten** erfordern **dünnere Leitungen** aber **dünnere Leitungen** verursachen **höheren Widerstand**.

Architekturelemente - Speicher

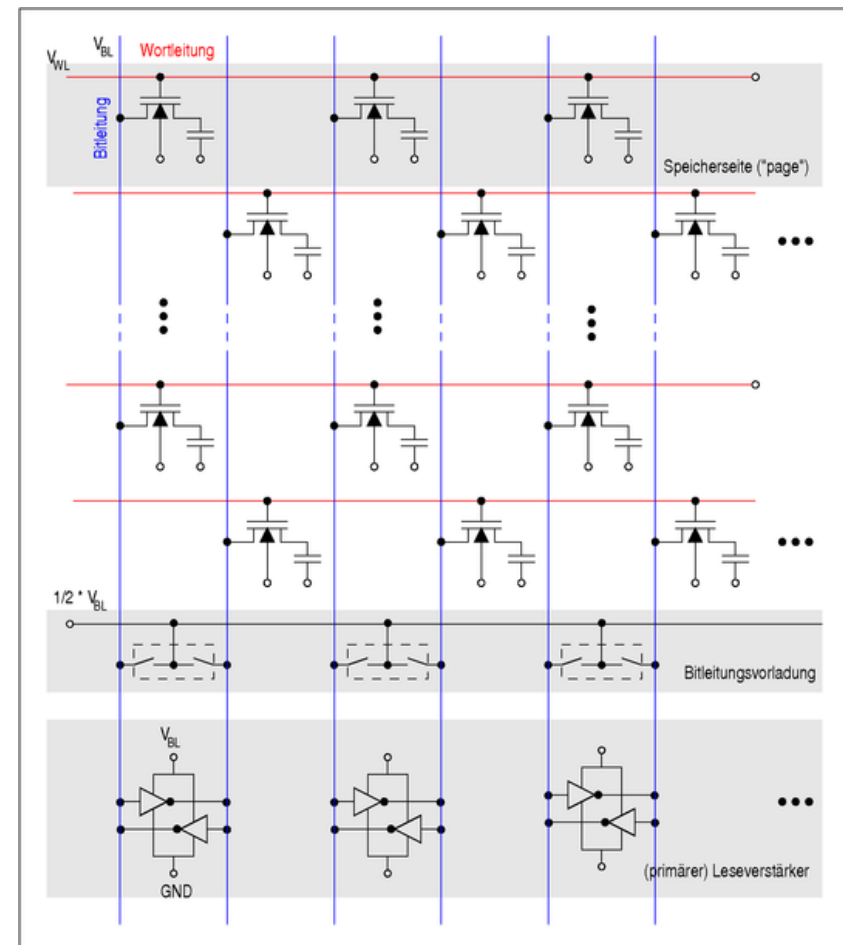
Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Speicherzeile* (Forts.)
(vereinfachte schematische Darstellung)



Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers:
Zellenfeld (vereinfachte Darstellung)

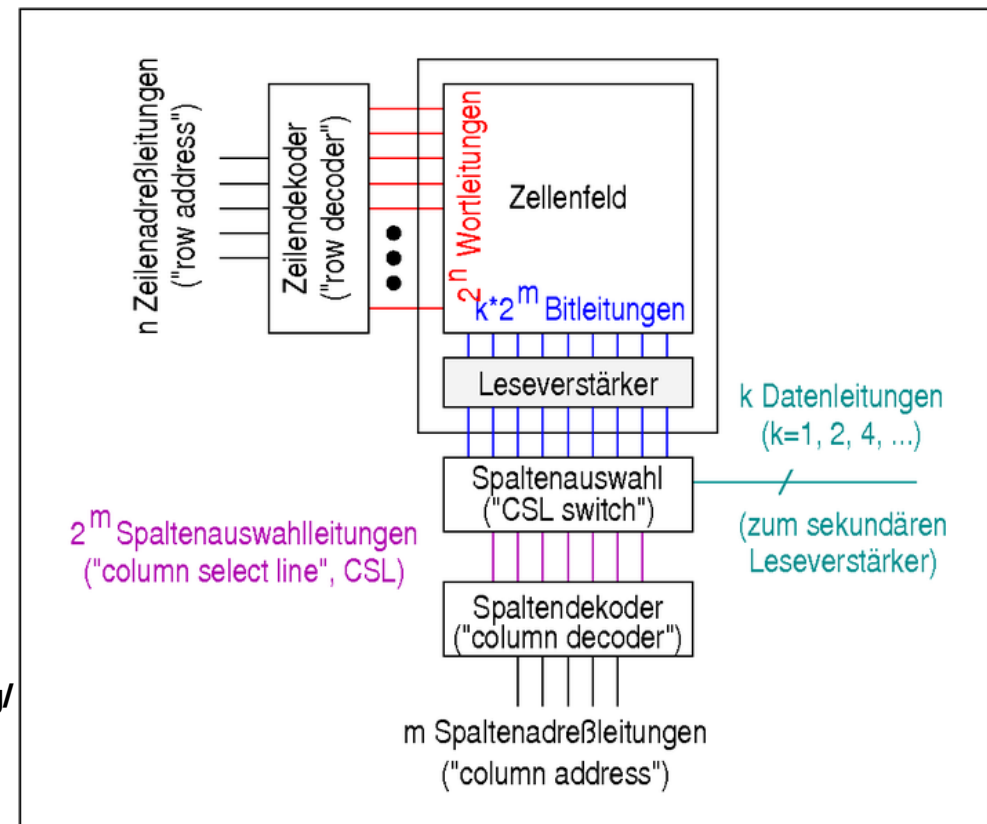


Quelle: Wikipedia „http://de.wikipedia.org/wiki/Dynamic_Random_Access_Memory“
Stand Aug. 2011

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: *Zellenfeld* inklusive *Zeilen-* und *Spaltenadresskodierung* (vereinfachte Darstellung)



Quelle: Wikipedia „http://de.wikipedia.org/wiki/Dynamic_Random_Access_Memory“
Stand Aug. 2011

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: Besonderheiten
 - Um die *PIN-Anzahl zu reduzieren* (ca. halbieren), werden die Adressleitungen gemultiplext.
 - => **Erfordert zwei Zyklen** für die vollständige Adresse!
 - *Burst-Mode*: Ausgabe aufeinanderfolgender Bits einer Zeile ohne vollständige Adressdekodierung (nur einmal zu Beginn).
 - => **Durchsatz ist höher!**

Architekturelemente - Speicher

Speicherzugriff – am Beispiel *DRAM*

- Aufbau eines *DRAM*-Speichers: Besonderheiten
 - *Refresh*: Erfolgt inzwischen (teilweise) intern bzw. automatisch; muss ansonsten durch externe Steuerlogik angestossen werden.

Die *Refresh*-Periode wird inzwischen teilweise temperaturabhängig durchgeführt.

Hintergrund ist der stark ansteigende *Leckstrom* der *DRAM*s bei höheren Temperaturen, z. B. verdoppelt er sich ca. bei einer Zunahme von 15°C auf 20°C.

=> **Stromsparen**; **Refresh ist sehr stromintensiv!**

Architekturelemente - Speicher

Speicheraufbau

- Idealerweise kann ein Rechner bzw. der Prozessor (Steuer- und Rechenwerk(e)) sehr schnell auf Speicher grosser Kapazität wahlfrei zugreifen, der zudem (fast) nichts kostet ...

... leider ist in der Realität:

- Schneller Speicher **sehr teuer !**
- Grosser Speicher **langsam** oder/und **sehr teuer !**

=> Speicherhierarchie

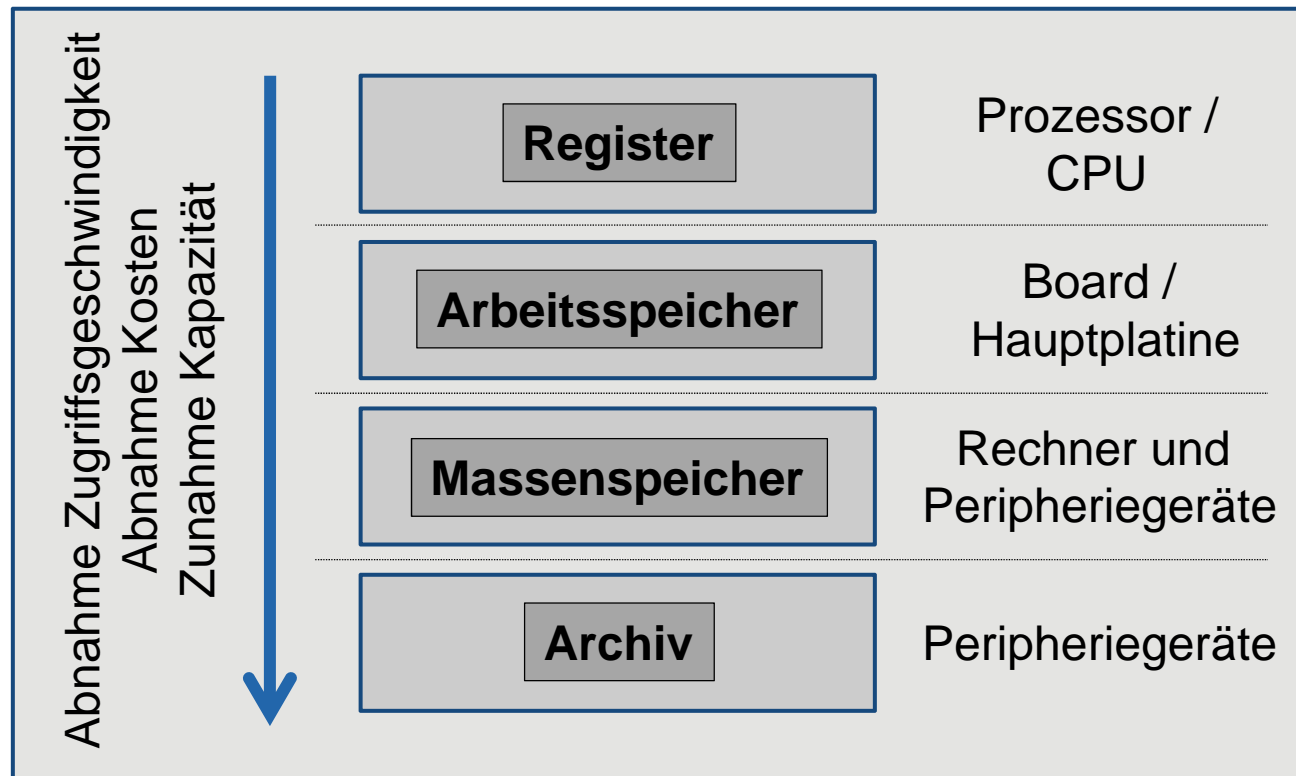
Architekturelemente - Speicher

Speicheraufbau – Speicherhierarchie

- Für aktive Inhalte, auf die schnell zugegriffen werden muss:
=> **Teurer sehr schneller** Speicher (Kapazität „**klein**“)
- Für weniger aktive Inhalte, auf die nur wenig / selten und langsam zugegriffen werden muss:
=> **Grosser preiswerter** Speicher

Architekturelemente - Speicher

Speicheraufbau – Speicherhierarchie



Architekturelemente - Speicher

Zürcher Hochschule
für Angewandte Wissenschaften

zhaw School of
Engineering

