Grundlagen der IT

Modul: Grundlegende Konzepte der IT

Semester I

- Vorlesungstermine: Kurs A
  - Fr, 29.10.2021, 13:00 – 15:30 Uhr

  - Fr, 05.11.2021, 13:00 – 15:30 Uhr
  - Fr, 12.11.2021, 13:00 – 15:30 Uhr
  - Fr, 19.11.2021, 13:00 – 15:30 Uhr
  - Fr, 26.11.2021, 13:00 – 15:30 Uhr

  - Fr, 03.12.2021, 13:00 – 15:30 Uhr
  - Fr, 10.12.2021, 13:00 – 15:30 Uhr
  - Fr, 17.12.2021, 13:00 – 15:30 Uhr
- Begleitmaterial:
  - „IT Grundlagen“ von Prof. Dr. A. Wiedemann

Agenda

Informatik

- Begriffsdefinition:
  - „Wissenschaft von der systematischen Darstellung, Speicherung, Verarbeitung und Übertragung von Informationen, besonders der automatischen Verarbeitung mithilfe von Digitalrechnern“

- Etymologie:
  - Kofferwort aus Information & Mathematik
  - Geprägt von Karl Steinbuch 1957

- Theoretische Informatik
  - Verifikation
  - Berechenbarkeitstheorie
  - Automatentheorie
  - Komplexitätstheorie

- Praktische Informatik
(Konzepte der Informatik auf
Probleme der Informatik anwenden)
  - Software Engineering
  - Dateisysteme
  - Sortier- & Suchalgorithmen

- Angewandte Informatik
(Konzepte der Informatik auf
Probleme anderer Gebiete anwenden)
  - Medizinische Informatik
  - Geoinformatik
  - Wirtschaftsinformatik

- Technische Informatik
  - Prozessortechnik
  - Rechnerkommunikation
  - Verteilte Systeme
  - Eingebettete Systeme

Geschichtliche Entwicklung

▶ Historische Entwicklung

Binäre Zahlendarstellung	Boolesche Algebra	Mathematisches Maschinenmodell (Turingmaschine)	Von-Neumann-Architektur
Gottfried Wilhelm Leibniz	**George Boole**	**Alan Turing**	**John von Neumann**
17. – 18. Jh.	**1847**	**1936**	**1945**

Was ist ein Computer?

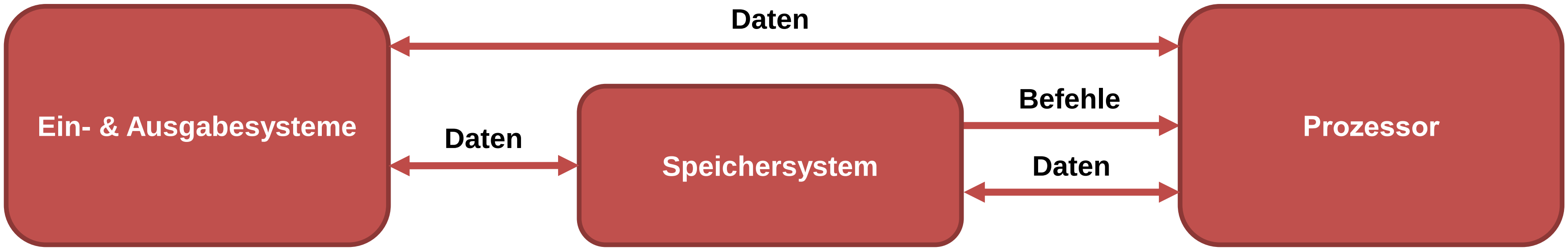
- Synonyme
  - Datenverarbeitungssystem
  - Rechner
  - Rechenanlage
  - Rechenautomat
  - Informationsverarbeitungssystem

- Definition eines Datenverarbeitungssystems nach DIN 44300
  - „Ein Datenverarbeitungssystem ist eine Funktionseinheit zur Verarbeitung und Aufbewahrung von Daten. Verarbeitung umfasst die Durchführung mathematischer, umformender, übertragender und speichernder Operationen.“

Was macht ein Computer?

- Mathematische Operationen
  - Arithmetische Operationen
  - Logische Operationen
- Umformende Operationen
  - Sortieren
  - (Ent-)Packen
  - (De-)Codieren
- Übertragende Operationen
  - Kommunikation mit anderen Systemen
  - Mensch-Maschine-Schnittstelle
- Speichernde Operationen
  - Ablegen & Wiederauffinden von Daten
  - Löschen von Daten

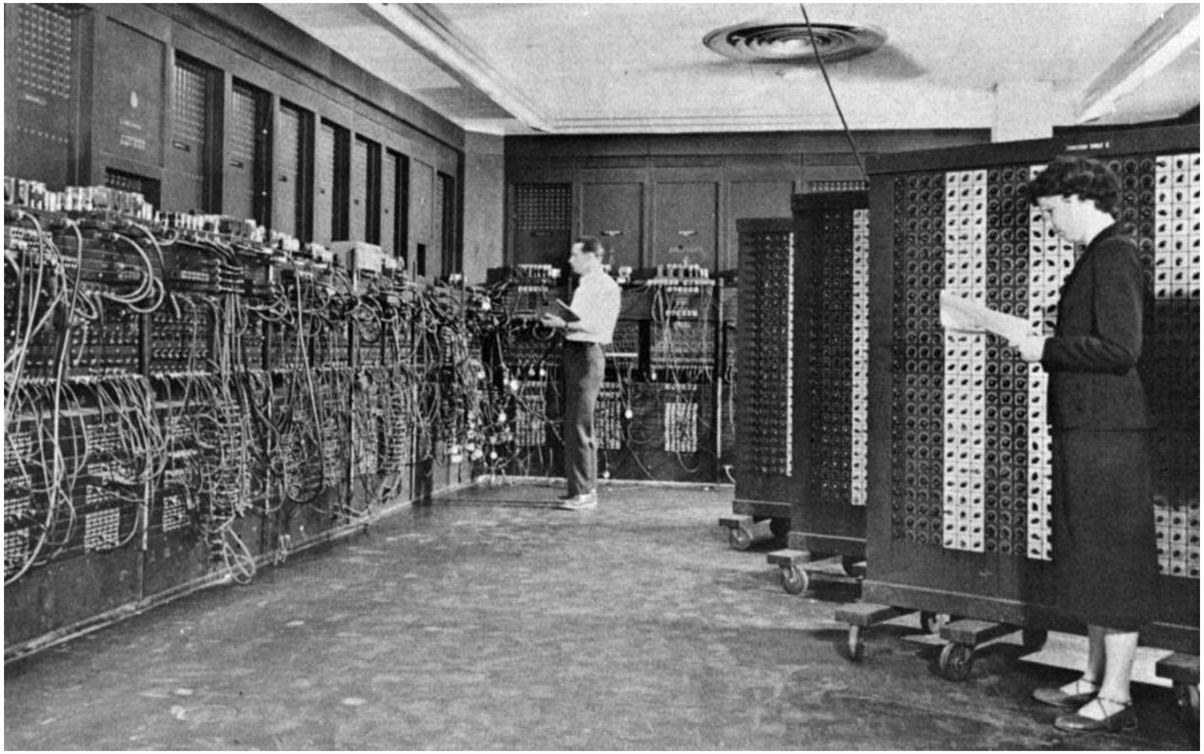
- Komponenten eines Computers



► Rechnergenerationen

Gen.	Zeitdauer	Technologie	Operationen/s
1	1946 – 1954	Vakuumröhren	40 000
2	1955 – 1964	Transistor	200 000
3	1965 – 1971	Small Scale Integration (SSI) Medium Scale Integration (MSI)	1 000 000
4	1972 – 1977	Large Scale Integration (LSI)	10 000 000
5	Ab 1978	Very Large Scale Integration (VLSI)	100 000 000

- 1. Generation
  - ENIAC von John P. Eckert & John W. Mauchly
  - 17 000 Röhren, 150 kW Leistungsaufnahme
  - Erster vollständig elektronischer Computer



- 2. Generation
  - Transistoren und Mikroelektronik
  - Höhere Programmiersprachen: ALGOL, COBOL, FORTRAN

2. Generation

Assembler vs. Pascal

```
 ASSUME  CS:CODE, DS:DATA;- dem Assembler die Zuordnung der Segmentregister zu den Segmenten mitteilen

DATA    SEGMENT                  ;Beginn des Datensegments
Meldung db   "Hallo Welt"        ;- Zeichenkette „Hallo Welt"
        db   13, 10              ;- Neue Zeile
        db   "$"                 ;- Zeichen, das INT 21h, Unterfunktion 09h als Zeichenkettenende verwendet
DATA    ENDS                     ;Ende des Datensegments

CODE    SEGMENT                  ;Beginn des Codesegments
Anfang:                          ;- Einsprung-Label fuer den Anfang des Programms
        mov ax, DATA             ;- Adresse des Datensegments in das Register „AX" laden
        mov ds, ax               ;  In Segmentregister „DS" uebertragen (DS-Register kann nicht direkt mit Konstante beschrieben werden)
        mov dx, OFFSET Meldung   ;- die zum Datensegment relative Adresse des Textes in das „DX" Datenregister laden
                                 ;  die vollstaendige Adresse von „Meldung" befindet sich nun im Registerpaar DS:DX
        mov ah, 09h              ;- die Unterfunktion 9 des Betriebssysteminterrupts 21h auswaehlen
        int 21h                  ;- den Betriebssysteminterrupt 21h aufrufen (hier erfolgt die Ausgabe des Textes am Schirm)
        mov ax, 4C00h            ;- die Unterfunktion 4Ch (Programmbeendigung) des Betriebssysteminterrupts 21h festlegen
        int 21h                  ;- diesen Befehl ausfuehren, damit wird die Kontrolle wieder an das Betriebssystem zurueckgegeben
CODE    ENDS                     ;Ende des Codesegments

END     Anfang                   ;- dem Assembler- und Linkprogramm den Programm-Einsprunglabel mitteilen
                                 ;- dadurch erhaelt der Befehlszaehler beim Aufruf des Programmes diesen Wert
```

```
program Hallo(output);
begin
  writeln('Hallo Welt')
end.
```

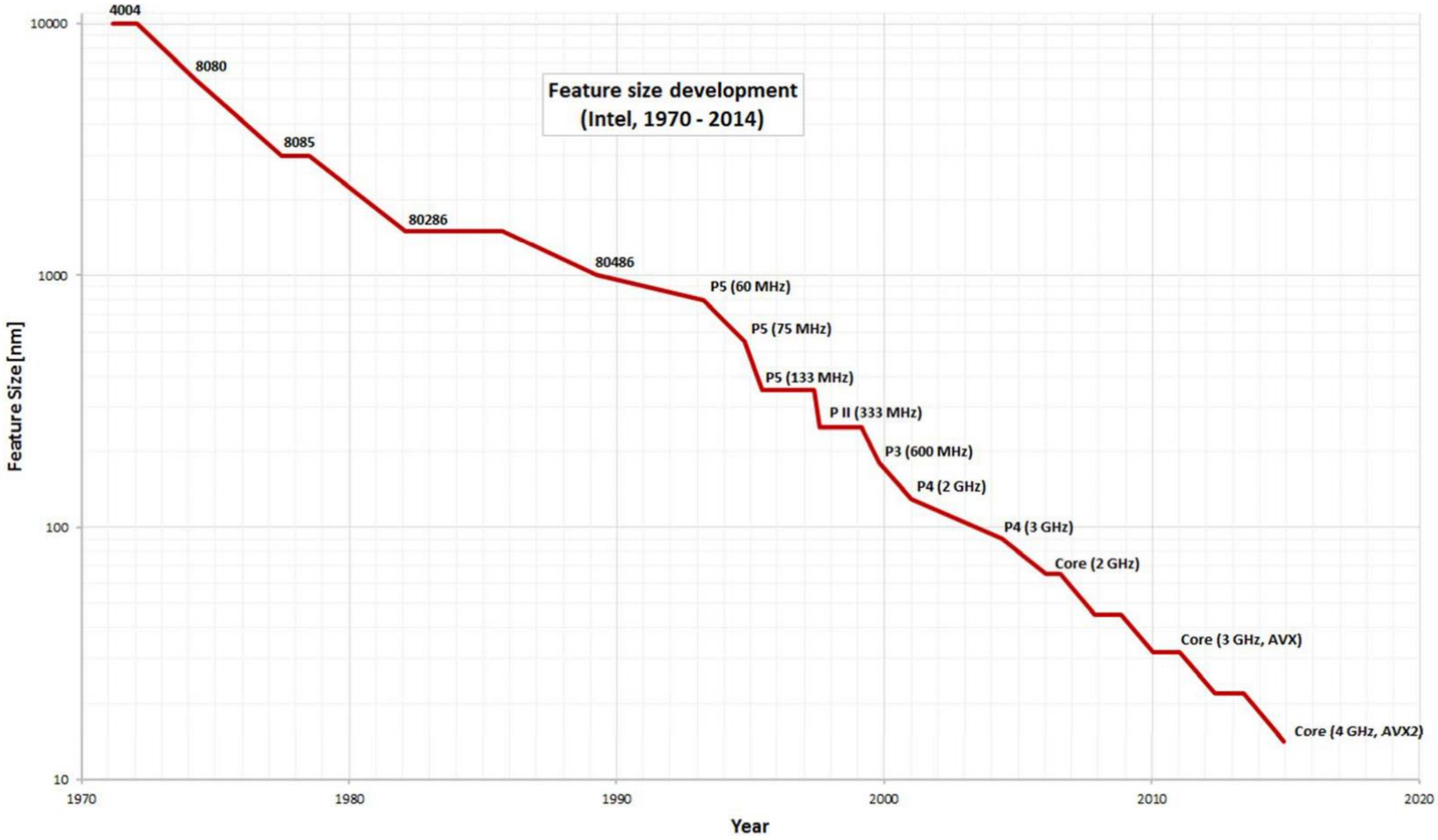
DHBW
Duale Hochschule
Baden-Württemberg
Mannheim

- 3. Generation
  - Integrierte Schaltkreise (ICs)
  - Halbleiterspeicher auf Basis von ICs
  - Betriebssysteme
  - Parallelrechner

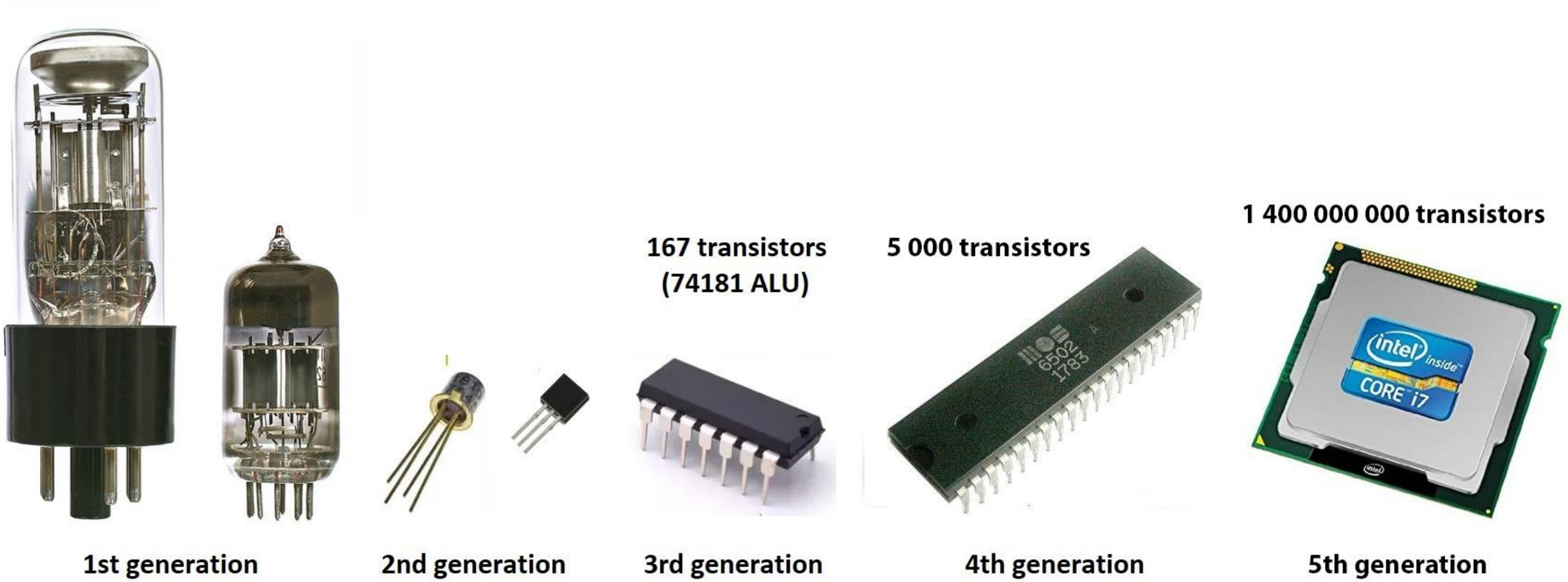
- 4. & 5. Generation
  - Unterscheidung bezügl. Logikfamilie (TTL, CMOS)
  - Kompletter Prozessor auf einzelnem Chip: Mikroprozessor
  - Speicher & E/A-Bausteine auf gleichem Chip: Mikrocomputer
  - Weitere Komponenten (A/D-Wandler, Sensoren, …): Mikrocontroller

- Aktuelle Großrechner:
  - SUMMIT am Oak Rich National Laboratory, aktuell leistungsfähigster Großrechner der Welt
    - Inbetriebnahme: 2016
    - 200 Peta-FLOPs = 200.000 Billionen Gleitkommaoperationen pro Sekunde
    - Verbund aus 4086 Servern mit jeweils zwei 22-Core-IBM-Prozessoren sowie 6 NVIDIA-Tesla-Grafik-Prozessoren
    - 10 Petabyte Arbeits-speicher

- Fertigungsgrößen



- Anzahl Transistoren pro Bauteil



Zahlen- & Stellenwertsysteme

- Herausbildung unterschiedlicher Systeme & Verfahren im Laufe der Geschichte.

- Man unterscheidet zwischen 3 Systemen:
  - Additionssysteme
  - Hybride Zahlensysteme
  - Stellenwertsysteme

- Definition:
  - „Ein Zahlensystem wird dazu verwendet, Zahlen darzustellen. Eine bestimmte Zahl wird gemäß den in dem verwendeten System bestehenden Regeln gebildet.“

- Definition Stellenwertsystem:
  - „Bei einem Stellenwertsystem mit der Basis B und den B Ziffern im Intervall $0, 1, \ldots, B-1$ hängt der Wert einer Ziffer nicht nur von der Form des Zeichens sondern auch von der Stelle in der Zahl ab. Die Ziffern einer Zahl eines Stellenwertsystems stellen die Koeffizienten der Potenzen der Basis dar. Dabei steht die niedrigwertigste Ziffer (in der Regel) ganz rechts.“

- In der Praxis verwendete Zahlensysteme:
  - Binärsystem
  - Oktalsystem
  - Dezimalsystem
  - Hexadezimalsystem
  - Hexagesimalsystem

Dezimal	Oktal	Binär	Hexadezimal
0	0	0	0
1	1	1	1
2	2	10	2
3	3	11	3
4	4	100	4
5	5	101	5
6	6	110	6
7	7	111	7
8	10	1000	8
9	11	1001	9
10	12	1010	A
11	13	1011	B
12	14	1100	C
13	15	1101	D
14	16	1110	E
15	17	1111	F
16	20	10000	10
17	21	10001	11

Binär- & Hexadezimalsystem

Binärsystem

- Dezimalsystem:
  - Stellenwertsystem zu Basis 10
  - Ziffern 0 – 9
  - Darstellung: 2345_d oder 2345_{10}

- Binärsystem:
  - Stellenwertsystem zur Basis 2
  - lediglich zwei Ziffern 0 und 1
  - Darstellung: 1101_b oder 1101_2

Binärsystem

- Analog zum Dezimalsystem hat jede Ziffer einer Binärzahl einen Wert, der von der Stelle der Ziffer in der Zahl abhängt. In diesem Fall zur Basis 2.

- Beispiele:
  - $10_2 = (1 \times 2^1) + (0 \times 2^0) = 2_{10}$
  - $11_2 = (1 \times 2^1) + (1 \times 2^0) = 3_{10}$
  - $100_2 = (1 \times 2^2) + (0 \times 2^1) + (0 \times 2^0) = 4_{10}$

- Kommazahlen werden - wie im Dezimalsystem - mit negativen Potenzen der Basis dargestellt:
  - $10011.1101_2 = 2^4 + 2^1 + 2^0 + 2^{-1} + 2^{-2} + 2^{-4}$

Binärsystem

- Umrechnung Binärsystem → Dezimalsystem
  - Ganzzahl:

    $z = 10100111_2$

    $= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

    $= 2^7 + 2^5 + 2^2 + 2^1 + 2^0$

    $= 128 + 32 + 4 + 2 + 1$

    $= 167_{10}$
  - Kommazahl:

    $z = 0.1001101_2$

    $= 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} + 1 \times 2^{-5} + 0 \times 2^{-6} + 1 \times 2^{-7}$

    $= 2^{-1} + 2^{-4} + 2^{-5} + 2^{-7}$

    $=1/2 + 1/16 + 1/32 + 1/128$

    $= 0.6015625_{10}$

Binärsystem

- Umrechnung Dezimalsystem → Binärsystem
  - Ganzzahl: $z = 47_{10}$

$47 / 2 = 23$	Rest 1	$\Rightarrow R_0 = 1$
$23 / 2 = 11$	Rest 1	$\Rightarrow R_1 = 1$
$11 / 2 = 5$	Rest 1	$\Rightarrow R_2 = 1$
$5 / 2 = 2$	Rest 1	$\Rightarrow R_3 = 1$
$2 / 2 = 1$	Rest 0	$\Rightarrow R_4 = 0$
$1 / 2 = 0$	Rest 1	$\Rightarrow R_5 = 1$

$47_{10} = 10\ 1111_2$

Binärsystem

- Umrechnung Dezimalsystem → Binärsystem
  - Dezimalzahl: $z = 0.828125_{10}$

$0.828125 \times 2 = 1.65625 \quad \Rightarrow$ Output: 1
$0.65625 \times 2 = 1.3125 \quad \Rightarrow$ Output: 1
$0.3125 \times 2 = 0.625 \quad \Rightarrow$ Output: 0
$0.625 \times 2 = 1.25 \quad \Rightarrow$ Output: 1
$0.25 \times 2 = 0.5 \quad \Rightarrow$ Output: 0
$0.5 \times 2 = 1.0 \quad \Rightarrow$ Output: 1

$z = 0.828125_{10} = 0.110101_2$

  - Zur Überprüfung wird die Binärdarstellung wieder in das Dezimalsystem umgerechnet.

Binärsystem

- Addition:
  - funktioniert genau wie die Addition von Dezimalzahlen, außer dass bereits ein Übertrag entsteht, sobald das Ergebnis größer als 1 ist.
  - Logik der Addition im Binärsystem:

a	b	a+b	Übertrag
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- Beispiel:

```
           101
      +     11
Übertrag  111
--------------
      = 1000
```

Binärsystem

- Subtraktion:
  - geschieht nach dem gleichen Verfahren, wie man es von Dezimalzahlen kennt.
  - Logik der Subtraktion im Binärsystem:

a	b	a-b	Borgen
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

  - Beispiel:

```
          10110
      -    1101
Borgen   1   1
---------------
      =    1001
```

- Wird so in Prozessoren nicht angewendet. Mehr dazu in Kapitel 1.2.2.

Binärsystem

- Übungen

  - **1.** Berechne den Dezimalwert der folgenden Dualzahlen!
    a) 101110011_2
    b) 110101101_2
    c) 11110110_2
    d) 100001110_2

  - **2.** Übertrage die folgenden Dezimalzahlen in Dualzahlen!
    a) 123_{10}
    b) 408_{10}
    c) 230_{10}
    d) 169_{10}

  - **3.** Addiere die folgenden Dualzahlen!
    a) $110101_2+10111_2$
    b) $100101_2+11101_2$
    c) $11100_2+10001_2$
    d) $101010_2+101010_2$

  - **4.** Subtrahiere die folgenden Dualzahlen!
    a) $11001_2-10101_2$
    b) $11110_2-10010_2$
    c) $10101_2-10011_2$
    d) $11100_2-11011_2$

Binärsystem

- Multiplikation:
  - Die Multiplikation zweier Binärzahlen führt man nach dem gleichen Schema aus, wie man es vom Dezimalsystem kennt
  - Logik der Multiplikation im Binärsystem:

a	b	a*b
0	0	0
0	1	0
1	0	0
1	1	1

- Beispiel:

```
  10110 * 101
  -----------
        10110
       00000
      10110
  -----------
      1101110
```

- Left Shift: Multiplikation einer Binärzahl mit einer Zweierpotenz (2^n). Zahl wird um *n* Stellen nach links verschoben & mit Nullen aufgefüllt.

Binärsystem

- Division:
  - wird durch wiederholte Subtraktion durchgeführt.

  - Beispiele:

```
1111 : 101 = 0011

1
-----------
11
-----------
111
-101
-----------
 101
-101
-----------
   0     Rest
```

```
10111 : 10 = 1011,1
-10
-----------
 011
 -10
-----------
   11
  -10
-----------
    1   Rest
-----------
    10
   -10
-----------
     0
```

Binärsystem

- Wurzelziehen:
  - Geschieht durch fortlaufende Multiplikation.

  - Beispiel $\sqrt{2}_{10}$ == $\sqrt{10}_2$:

$10 \times 10_2$	$== 100_2$	$> 10_2$
$1 \times 1_2$	$== 1_2$	$< 10_2$
$1.1 \times 1.1_2$	$== 10.01_2$	$> 10_2$
$1.01 \times 1.01_2$	$== 1.1001_2$	$< 10_2$
$1.011 \times 1.011_2$	$== 1.111001_2$	$< 10_2$
$1.0111 \times 1.0111_2$	$== 10.00000001_2$	$> 10_2$
$1.01101 \times 1.01101_2$	$== 1.1111101001_2$	$< 10_2$

  - Ergebnis: $\sqrt{2}_{10} == \sqrt{10}_2 == 1.01101_2 == 1\ 13/32_{10} == 1.40625_{10}$

Binärsystem

- Komplementdarstellung:
  - Subtrahieren durch Addition ist mithilfe der Komplementbildung möglich.

  - Beispiel: 7 – 4 == 3
    - Gesucht wird nun die Zahl $\overline{4}$ für die gilt 4 + $\overline{4}$ == 10
    - Offensichtlich ist $\overline{4}$ == 6
    - Man nennt $\overline{4}$ == 6 die zu 4 komplementäre Zahl oder auch das Komplement zu 4.

  - Die charakteristische Eigenschaft des Komplements $\overline{z}$ einer Zahl z im Dezimalsystem ist, dass die Addition der Zahl z und ihr Komplement $\overline{z}$ die nächsthöhere Potenz der Basis 10 ergibt.

  - Am Beispiel: 7 – 4 == 3   7 + $\overline{4}$ == 3   7 + 6 == 13
Überlauf ignorieren: 7 + 6 == (1)3

Binärsystem

- Wenn die Subtraktion durch die Addition des Komplements ersetzt werden kann, benötigt das Rechenwerk der CPU lediglich ein Addierwerk, um alle elementaren arithmetischen Operationen auszuführen.
- Das Komplement einer Zahl kann auf beliebige Stellenwertsysteme verallgemeinert werden.

- Definitionen:
  - (a) Das Komplement einer Zahl ist die Ergänzung einer vorgegebenen Zahl.
  - (b) Das $B - 1$–Komplement einer n–stelligen Zahl im Stellenwertsystem zur Basis B ist die Ergänzung zur größten Zahl mit n Stellen.
  - (c) Das B–Komplement einer n–stelligen Zahl im Stellenwertsystem zur Basis B ist die Ergänzung zur kleinsten Zahl mit n+1 Stellen.

Binärsystem

- Komplementbildung im Binärsystem (B = 2):

  - Definition besagt: addiert man zur einer gegebenen Zahl das B − 1–Komplement (Einerkomplement), ergibt sich immer die größte Zahl mit der gleichen Stellenanzahl.

  - Beispiel: $z = 1010\ 0010_2$

  - Man erhält das Einerkomplement durch die Subtraktion von z von der größten darstellbaren Zahl mit 8 Stellen:

$$\begin{array}{r} 1111\ 1111 \\ -1010\ 0010 \\ \hline 0101\ 1101 \end{array}$$

  - Einfacher: jedes Bit einer gegebenen Zahl kippen ergibt das Einerkomplement

Binärsystem

- Komplementbildung im Binärsystem (B = 2):

  - Addiert man zu einer Zahl z das zugehörige Einerkomplement, erhält man die größte darstellbare Zahl mit gleicher Stellenanzahl:

$$\begin{array}{rr} z & 1010\ 0010 \\ +\overline{z} & 0101\ 1101 \\ \hline = & 1111\ 1111 \end{array}$$

  - Addiert man zu diesem Ergebnis noch 1, erhält man:

$$\begin{array}{rr} & 1111\ 1111 \\ + & 0000\ 0001 \\ \hline = & 1\ 0000\ 0000 \end{array}$$

  - Abgesehen von dem Überlauf in die n + 1te Stelle ist das Ergebnis also 0. Wenn also der Überlauf ignoriert wird, können wir schreiben:

    $z + \overline{z} + 1 = 0$ oder $\overline{z} + 1 = -z$.

  - Resultate:
    - $\overline{z} + 1$ ist eine Darstellung für $-z$.
    - Die Binärzahl $\overline{z} + 1$ heißt Zweierkomplement der Zahl z.

Binärsystem

- Komplementbildung im Binärsystem (B = 2):
  - Merke!
    - Die Subtraktion eine Binärzahl kann auf die Addition des Zweierkomplements zurückgeführt werden, wobei der Übertrag ignoriert wird.
    - Die Bildung des Zweierkomplements ist im Binärsystem besonders effektiv umsetzbar, da dies im wesentlichen eine Negation der Bits ist, mit anschließender Addition der Zahl 1.
  - Umrechnung einer Dezimalzahl x in das Zweierkomplement:
    - 1. Bestimme die Anzahl der Binärstellen n.
    - 2. Wenn x positiv, Umrechnung wie bei Binärcodierung
    - 3. Wenn x negativ
      - (a) Binärcode von $|x|$
      - (b) Einerkomplement bilden (Bits kippen)
      - (c) 1 addieren

Binärsystem

- Übungen

- **1.** Multiplizieren Sie folgende binäre Zahlen:
  a) 0101 * 0010 0101
  b) 1101 * 1101 1011
  c) 1001 1001 0101 * 0111 1101
  d) 1111 1010 0101 1101 * 1111 0100 1001
  e) 1111 0110 0000 1000 * 1111 0000 0101 0101

- 3. Subtrahieren Sie folgende duale Zahlen mittels Bildung des Zweierkomplements:
  a) 0101 1011 - 0010 0101
  b) 1101 0100 - 0101 1011
  c) 1001 1001 0101 1101 - 0101 1101
  d) 1111 1010 0101 1101 - 1101 0111 1010 1001
  e) 1111 0110 0000 1000 - 0111 0000 1011 0101

- **2.** Dividieren Sie die folgenden Binärzahlen:
  a) 1100 / 0100
  b) 1101 0010 / 0110
  c) 1111 0000 1100 / 1101 0110
  d) 0101 1010 0101 0000 / 1010 1010
  e) 1101 1111 1100 1000 / 1111 1000
  f) 1011 0010 0010 / 0111 1000
  g) 1000 1000 1101 / 0010 0000

Hexadezimalsystem

Hexadezimalsystem

- Binärsystem vorteilhaft zur Verarbeitung von Zahlen mittels Computern, für den Menschen aber eher schwerfällig
- Kompakte Notation präferiert, wenn mit Rohdaten in Computern gearbeitet werden muss

- Hexadezimalsystem:
  - Stellenwertsystem zur Basis B = 16
  - Dargestellt durch die hexadezimalen Ziffern:
    0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

- Beispiele: $2F4_{16}$, $FF5A_{16}$, $123AD2_{16}$

- Umrechnung vom Hexadezimalsystem in das Dezimalsystem und umgekehrt - oder ein anderes Stellenwertsystem - erfolgt nach den bekannten Verfahren

Hexadezimalsystem

- Beispiel Hexadezimal → Dezimal bei Ganzzahlen:

$$
\begin{aligned}
2F4_{16} &= 2_{16} \cdot 16^2 + F_{16} \cdot 16^1 + 4_{16} \cdot 16^0 \\
&= 2_{10} \cdot 16^2 + 15_{10} \cdot 16^1 + 4_{10} \cdot 16^0 \\
&= 756_{10}
\end{aligned}
$$

- Beispiel Hexadezimal → Binär bei Ganzzahlen:

$$2F4_{16} = 0010\ 1111\ 0100_2$$

  - Ziffern der Hexadezimalzahl werden einfach durch die jeweiligen Vierergruppen von Bits ersetzt

Hexadezimalsystem

- Beispiel Hexadezimal → Dezimal bei Dezimalzahlen:

  - Vor- & Nachkommateil getrennt voneinander umrechnen:

$2F4{,}5A4_{16}$

$$2F4_{16} = 2_{16} \cdot 16^2 + F_{16} \cdot 16^1 + 4_{16} \cdot 16^0$$
$$= 2_{10} \cdot 16^2 + 15_{10} \cdot 16^1 + 4_{10} \cdot 16^0$$
$$\underline{\mathbf{= 756_{10}}}$$

$$0{,}5A4_{16} \quad = 4_{16} \cdot 16^0 = 4_{10} \cdot 1 = 4$$
$$= A_{16} \cdot 16^1 = 10_{10} \cdot 16 = 160$$
$$= 5_{16} \cdot 16^2 = 5_{10} \cdot 256 = 1280$$

$\underline{\mathbf{1444_{10}}}$

  - Suchen der nächsthöheren Potenz zur Basis 16:

16^0	16^1	16^2	$\mathbf{16^3}$	16^4
1	16	256	**4096**	65 536

  - Nachkommateil durch entspr. Potenz zur Basis 16 teilen: 1444 : 4096 = $\underline{\mathbf{0{,}3525390625_{10}}}$

  - Vor- & Nachkommateil addieren: 756 + 0,3525390625 = $\underline{\mathbf{756{,}3525390625_{10}}}$

Hexadezimalsystem

- Umrechnung Dezimal → Hexadezimal:
  - gleiches Verfahren wie bei der Umrechnung einer Dezimalzahl in das Binärsystem
  - Division (mit ganzzahligem Rest) dabei jedoch mit 16 durchführen

  - Beispiel 3521_{10}:

$$
\begin{array}{rcll}
3521 \text{ div } 16 & = & 220 & \text{Rest } 1_{10} \Rightarrow R_0 = 1_{16} \\
220 \text{ div } 16 & = & 13 & \text{Rest } 12_{10} \Rightarrow R_1 = C_{16} \\
13 \text{ div } 16 & = & 0 & \text{Rest } 13_{10} \Rightarrow R_2 = D_{16}
\end{array}
$$

$$3521_{10} = DC1_{16}$$

Hexadezimalsystem

- Gründe für die Verwendung des Hexadezimalsystems:
  - Die Notation ist wesentlich kompakter als die Binärschreibweise.
  - In den meisten Computern werden die Daten in Vielfachen von 4 Bit-Gruppen organisiert, und daher im hexadezimalen Raster.
  - Es ist sehr einfach, zwischen dem Hexadezimalsystem und dem Binärsystem umzurechnen.

- Binäre Ziffern werden dazu in Viererblöcke gruppiert. Jeder möglichen Kombination von vier binären Ziffern wird dann ein Symbol vergeben:

0000 = 0	1000 = 8
0001 = 1	1001 = 9
0010 = 2	1010 = A
0011 = 3	1011 = B
0100 = 4	1100 = C
0101 = 5	1101 = D
0110 = 6	1110 = E
0111 = 7	1111 = F

Hexadezimalsystem

- Übung

  - **1.** Rechnen Sie folgende Zahlen in die jeweiligen Zahlensysteme um.

    a) 109_{10} = . . . $_2$, . . . $_{16}$

    b) $5D_{16}$ = . . . $_{10}$, . . . $_2$

    c) 2011_{10} = . . . $_{16}$, . . . $_2$

    d) 1001010101010101_2 = . . . $_{16}$, . . . $_{10}$

    e) 0.78515625_{10} = . . . $_2$, . . . $_{16}$

    f) 0.1_{10} = . . . $_{16}$, . . . $_2$

    g) 101.1101_2 = . . . $_{10}$, . . . $_{16}$

    h) $3FD.A_{16}$ = . . . $_{10}$, . . . $_2$