Digitaltechnik
Wintersemester 2017/2018
2. Vorlesung

TECHNISCHE UNIVERSITÄT DARMSTADT

Inhalt

Einleitung

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Übungsbetrieb angelaufen
  - bisher 732 Anmeldungen im Moodle
  - 650 Zuordnungen zu Übungsgruppen
  - G22 und G23 (Mo 14:25-16:05) kaum nachgefragt
  - Mehr Interessenten für G24 (Fr 11:40-13:20)?

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Übungsbetrieb angelaufen
  - bisher 732 Anmeldungen im Moodle
  - 650 Zuordnungen zu Übungsgruppen
  - G22 und G23 (Mo 14:25-16:05) kaum nachgefragt
  - Mehr Interessenten für G24 (Fr 11:40-13:20)?
- Testate erst ab KW 44

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Übungsbetrieb angelaufen
  - bisher 732 Anmeldungen im Moodle
  - 650 Zuordnungen zu Übungsgruppen
  - G22 und G23 (Mo 14:25-16:05) kaum nachgefragt
  - Mehr Interessenten für G24 (Fr 11:40-13:20)?
- Testate erst ab KW 44
- Übungsausfall am 31.10.2017 (Reformationstag)
  - betrifft G05, G08, G09
  - kein Ersatztermin
  - Lösungsvorschlag und Sprechstunde nutzen
  - Zugehörige Testate (T2) wie geplant in KW 45

Rückblick auf letzte Vorlesung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beherrschen von Komplexität
  - Abstraktion
  - Diszplin
  - Hierarchie
  - Modularität
  - Regularität

- Digitale Abstraktion
  - Bits und Bitfolgen
  - Binäre Größenfaktoren (KiBi, MeBi, GiBi, TeBi)
  - Nibble, Bytes, Wort

Wiederholung:
Zweierpotenzen schnell schätzen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Etwa wie viele Farben definiert
  - 15 bit Real Color?
  - 24 bit True Color?
  - 42 bit Deep Color?

$2^5 \cdot 2^{10}$

32 K

$2^4 \cdot 2^{20}$

16 M

$2^2 \cdot 2^{40}$

4 T

Wiederholung: Zweierpotenzen schnell schätzen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Etwa wie viele Farben definiert
  - 15 bit Real Color?
  - 24 bit True Color?
  - 42 bit Deep Color?
- Wie viele Bits nötig zur Repräsentation von
  - 24 Übungsgruppen 5
  - 750 Studierenden (DT) 10
  - 26 360 Studierenden (TU Da) 15

Wiederholung: Zweierpotenzen schnell schätzen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Etwa wie viele Farben definiert
  - 15 bit Real Color? $2^5 \cdot 2^{10} \approx$ 32 Tausend
  - 24 bit True Color? $2^4 \cdot 2^{20} \approx$ 16 Millionen
  - 42 bit Deep Color? $2^2 \cdot 2^{40} \approx$ 4 Billionen
- Wie viele Bits nötig zur Repräsentation von
  - 24 Übungsgruppen 5 bit
  - 750 Studierenden (DT) 10 bit
  - 26 360 Studierenden (TU Da) 15 bit

Überblick der heutigen Vorlesung

- Zahlensysteme: Bitfolgen $\leftrightarrow$ (ganze) Zahlen
  - Dezimal-, Binär-, Hexadezimalzahlen
  - Darstellung
  - Umrechnung
  - Addition von Binärzahlen
  - Vorzeichenbehaftete Binärzahlen

- Logikgatter: Einfache Boolsche Funktionen
  - Wahrheitswertetabellen
  - Symbole und Schreibweisen
  - Anwendung

Harris 2013
Kap. 1.4 + 1.5
Seite 9 - 22

Darstellung von natürlichen Zahlen

0010

Definition von Zahlenmengen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$

Definition von Zahlenmengen

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$

Definition von Zahlenmengen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$
- Rationale Zahlen $\mathbb{Q} = \{\frac{a}{b} : a \in \mathbb{Z} \wedge b \in \mathbb{N} \wedge b \neq 0\}$

Definition von Zahlenmengen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$
- Rationale Zahlen $\mathbb{Q} = \{\frac{a}{b} : a \in \mathbb{Z} \land b \in \mathbb{N} \land b \neq 0\}$
- Reele Zahlen $\mathbb{R}$

Definition von Zahlenmengen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$
- Rationale Zahlen $\mathbb{Q} = \{\frac{a}{b} : a \in \mathbb{Z} \wedge b \in \mathbb{N} \wedge b \neq 0\}$
- Reele Zahlen $\mathbb{R}$
- Komplex, Transzendent, Algebraisch, ...

Definition von Zahlenmengen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$
- Rationale Zahlen $\mathbb{Q} = \{\frac{a}{b} : a \in \mathbb{Z} \wedge b \in \mathbb{N} \wedge b \neq 0\}$
- Reele Zahlen $\mathbb{R}$
- Komplex, Transzendent, Algebraisch, ...
- $|\mathbb{N}| = |\mathbb{Z}| = |\mathbb{Q}| < |\mathbb{R}|$

Definition von Zahlenmengen

- Natürliche Zahlen $\mathbb{N} = \{0, 1, 2, ...\}$
- Ganze Zahlen $\mathbb{Z} = \{..., -2, -1, 0, 1, 2, ...\}$
- Rationale Zahlen $\mathbb{Q} = \{\frac{a}{b} : a \in \mathbb{Z} \wedge b \in \mathbb{N} \wedge b \neq 0\}$
- Reele Zahlen $\mathbb{R}$
- Komplex, Transzendent, Algebraisch, ...
- $|\mathbb{N}| = |\mathbb{Z}| = |\mathbb{Q}| < |\mathbb{R}|$
- $\infty \notin \mathbb{N}$

Darstellungen von natürlichen Zahlen — Beispiele

Dezimal: 5347 $= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000$

Darstellungen von natürlichen Zahlen — Beispiele

Dezimal: 5347

$$\begin{aligned} 5347 &= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000 \\ &= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3 \end{aligned}$$

Darstellungen von natürlichen Zahlen — Beispiele

Dezimal: 5347

$$\begin{aligned} 5347 &= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000 \\ &= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3 \\ &=: 5347_{10} \end{aligned}$$

Darstellungen von natürlichen Zahlen — Beispiele

Dezimal: $5347 = 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000$

$= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3$

$=: 5347_{10}$

Binär: $1101_2 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3$

Darstellungen von natürlichen Zahlen — Beispiele

TECHNISCHE UNIVERSITÄT DARMSTADT

Dezimal: 5347

$$
\begin{aligned}
&= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000 \\
&= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3 \\
&=: 5347_{10}
\end{aligned}
$$

Binär: 1101_2

$$
\begin{aligned}
&= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 \\
&= 1 \cdot 1 + 0 \cdot 2 + 1 \cdot 4 + 1 \cdot 8 \\
&= 13_{10}
\end{aligned}
$$

Darstellungen von natürlichen Zahlen — Beispiele

TECHNISCHE UNIVERSITÄT DARMSTADT

Dezimal: 5347

$$\begin{aligned} 5347 &= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000 \\ &= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3 \\ &=: 5347_{10} \end{aligned}$$

Binär: 1101_2

$$\begin{aligned} 1101_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 \\ &= 1 \cdot 1 + 0 \cdot 2 + 1 \cdot 4 + 1 \cdot 8 \\ &= 13_{10} \end{aligned}$$

Hexadezimal: $1F3A_{16}$

$$1F3A_{16} = 10 \cdot 16^0 + 3 \cdot 16^1 + 15 \cdot 16^2 + 1 \cdot 16^3$$

Darstellungen von natürlichen Zahlen — Beispiele

Dezimal: 5347

$$
\begin{aligned}
5347 &= 7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000 \\
&= 7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3 \\
&=: 5347_{10}
\end{aligned}
$$

Binär: 1101_2

$$
\begin{aligned}
1101_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 \\
&= 1 \cdot 1 + 0 \cdot 2 + 1 \cdot 4 + 1 \cdot 8 \\
&= 13_{10}
\end{aligned}
$$

Hexadezimal: $1F3A_{16}$

$$
\begin{aligned}
1F3A_{16} &= 10 \cdot 16^0 + 3 \cdot 16^1 + 15 \cdot 16^2 + 1 \cdot 16^3 \\
&= 10 \cdot 1 + 3 \cdot 16 + 15 \cdot 256 + 1 \cdot 4096 \\
&= 7994_{10}
\end{aligned}
$$

Darstellungen von natürlichen Zahlen — Verallgemeinerung (Abstraktion)

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition: vorzeichenloses Stellenwertsystem

Für eine Basis $b \in \mathbb{N} \wedge b \geq 2$ ist $Z_b := \{0, 1, \ldots, b-1\}$ die Menge der verfügbaren Ziffern. Die Funktion $\mathrm{u}_{b,k}$ bildet eine Ziffernfolge der Breite $k \in \mathbb{N}$ auf eine natürliche Zahl ab:

$$\mathrm{u}_{b,k} : (a_{k-1} \ldots a_1 a_0) \in Z_b^k \mapsto \sum_{i=0}^{k-1} a_i \cdot b^i \in \mathbb{N}$$

Vorzeichenlosen Stellenwertsystem

- polyadisches Zahlensystem

Vorzeichenlosen Stellenwertsystem

TECHNISCHE UNIVERSITÄT DARMSTADT

- polyadisches Zahlensystem
- Niedrigstwertige Stelle (LSB): a_0
- Höchstwertige Stelle (MSB): a_{k-1}

Vorzeichenlosen Stellenwertsystem

- polyadisches Zahlensystem
- Niedrigstwertige Stelle (LSB): a_0
- Höchstwertige Stelle (MSB): a_{k-1}
- Kleinste darstellbare Zahl: $\sum_{i=0}^{k-1} 0 \cdot b^i = 0$

Vorzeichenlosen Stellenwertsystem

TECHNISCHE UNIVERSITÄT DARMSTADT

- polyadisches Zahlensystem
- Niedrigstwertige Stelle (LSB): a_0
- Höchstwertige Stelle (MSB): a_{k-1}
- Kleinste darstellbare Zahl: $\sum_{i=0}^{k-1} 0 \cdot b^i = 0$
- Größte darstellbare Zahl: $\sum_{i=0}^{k-1} (b-1) \cdot b^i = b^k - 1$

Vorzeichenlosen Stellenwertsystem

TECHNISCHE UNIVERSITÄT DARMSTADT

- polyadisches Zahlensystem
- Niedrigstwertige Stelle (LSB): a_0
- Höchstwertige Stelle (MSB): a_{k-1}
- Kleinste darstellbare Zahl: $\sum_{i=0}^{k-1} 0 \cdot b^i = 0$
- Größte darstellbare Zahl: $\sum_{i=0}^{k-1} (b-1) \cdot b^i = b^k - 1$
- Anzahl der darstellbaren Werte: $\left|Z_b^k\right| = \left|Z_b\right|^k = b^k$

Vorzeichenlosen Stellenwertsystem

- polyadisches Zahlensystem
- Niedrigstwertige Stelle (LSB): a_0
- Höchstwertige Stelle (MSB): a_{k-1}
- Kleinste darstellbare Zahl: $\sum_{i=0}^{k-1} 0 \cdot b^i = 0$
- Größte darstellbare Zahl: $\sum_{i=0}^{k-1} (b-1) \cdot b^i = b^k - 1$
- Anzahl der darstellbaren Werte: $\left|Z_b^k\right| = \left|Z_b\right|^k = b^k$
- Eineindeutig (bijektiv) auf Wertebereich $\{0, \ldots, b^k - 1\}$ für festes k

Häufig verwendete Basen

	Dual/Binär	Oktal	Dezimal	Hexadezimal
b	2	8	10	16
Z_b	$\{0, 1\} := \mathbb{B}$	$\{0, \dots, 7\}$	$\{0, \dots, 9\}$	$\{0, \dots, 9, A, B, C, D, E, F\}$
Literale	11010011_2	323_8	211_{10}	$D3_{16}$
	0b11010011	0o323	0d211	0xD3
		0323	211	

Häufig verwendete Basen

	Dual/Binär	Oktal	Dezimal	Hexadezimal
b	2	8	10	16
Z_b	$\{0, 1\} := \mathbb{B}$	$\{0, \ldots, 7\}$	$\{0, \ldots, 9\}$	$\{0, \ldots, 9, A, B, C, D, E, F\}$
Literale	11010011_2	323_8	211_{10}	$D3_{16}$
	0b11010011	0o323	0d211	0xD3
		0323	211	

- Weniger gebräuchlich:
  - $b = 20$ wenn man mit Händen *und* Füßen rechnet
  - $b = 60$ zur Angabe von Zeit bzw. Längen-/Breitengrade
  - $b = 12$ ein „Dutzend“

Umrechnen zwischen Zahlensystemen

0010101001000011001001110001101000011011
0100111010001011000001101111011010101100
0000010110111110101111100101001011111001
0111010000010101100000001110000000110100
0100010110101110000100110100110101111101
0111101111110101000001110010110110111110
1101110000101101110110010010001000100000
0000001010111011110011010000111110111011
1000100001010110001001111010010111110011
0100010011001100100010000100001001111001
1000010110010001001110101111101001110110
0110000011000101100011000101100010011000
1000101101010100110001110100100100011011
1101111110000111010101101000010000101100
0111111101001000100001000100000010011101
1001101011110000100000001100011111101110

Handwerkszeug: Zweiterpotenzen

TECHNISCHE UNIVERSITÄT DARMSTADT

16^0	$=$	8^0	$=$	2^0	$=$	1					2^{10}	$=$	1 Ki
				2^1	$=$	2					2^{11}	$=$	2 Ki
				2^2	$=$	4	16^3	$=$	8^4	$=$	2^{12}	$=$	4 Ki
		8^1	$=$	2^3	$=$	8					2^{13}	$=$	8 Ki
16^1			$=$	2^4	$=$	16					2^{14}	$=$	16 Ki
				2^5	$=$	32			8^5	$=$	2^{15}	$=$	32 Ki
		8^2	$=$	2^6	$=$	64	16^4			$=$	2^{16}	$=$	64 Ki
				2^7	$=$	128	16^5			$=$	2^{20}	$=$	1 Mi
16^2			$=$	2^8	$=$	256			8^{10}	$=$	2^{30}	$=$	1 Gi
		8^3	$=$	2^9	$=$	512	16^{10}			$=$	2^{40}	$=$	1 Ti

Handwerkszeug: Nibble-Werte

TECHNISCHE UNIVERSITÄT DARMSTADT

0000_2	=		=	
0001_2	=		=	
0010_2	=		=	
0011_2	=		=	
0100_2	=		=	
0101_2	=		=	
0110_2	=	6_{10}	=	6_{16}
0111_2	=		=	
1000_2	=		=	
1001_2	=		=	
1010_2	=	10_{10}	=	A_{16}
1011_2	=		=	
1100_2	=		=	
1101_2	=		=	
1110_2	=		=	
1111_2	=		=	

Handwerkszeug: Nibble-Werte

$$
\begin{aligned}
0000_2 &= 0_{10} &= 0_{16}\\
0001_2 &= 1_{10} &= 1_{16}\\
0010_2 &= 2_{10} &= 2_{16}\\
0011_2 &= 3_{10} &= 3_{16}\\
0100_2 &= 4_{10} &= 4_{16}\\
0101_2 &= 5_{10} &= 5_{16}\\
0110_2 &= 6_{10} &= 6_{16}\\
0111_2 &= 7_{10} &= 7_{16}\\
1000_2 &= 8_{10} &= 8_{16}\\
1001_2 &= 9_{10} &= 9_{16}\\
1010_2 &= 10_{10} &= A_{16}\\
1011_2 &= 11_{10} &= B_{16}\\
1100_2 &= 12_{10} &= C_{16}\\
1101_2 &= 13_{10} &= D_{16}\\
1110_2 &= 14_{10} &= E_{16}\\
1111_2 &= 15_{10} &= F_{16}
\end{aligned}
$$

Binär/Hexadezimal → Dezimal

- Polyadische Abbildung anwenden:
- $u_{2,5}(10011_2) =$ $1 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^4$

  $1 + 2 + 16$

  $= 19$

Binär/Hexadezimal → Dezimal

TECHNISCHE UNIVERSITÄT DARMSTADT

- Polyadische Abbildung anwenden:
- $u_{2,5}(10011_2) =$
- $u_{16,3}(4AF_{16}) =$ $15 \cdot 16^0 + 10 \cdot 16^1 + 4 \cdot 16^2$

Binär/Hexadezimal $\rightarrow$ Dezimal

TECHNISCHE UNIVERSITÄT DARMSTADT

- Polyadische Abbildung anwenden:
- $u_{2,5}(10011_2) = 2^4 + 2^1 + 2^0 = 19_{10}$
- $u_{16,3}(4AF_{16}) = 4 \cdot 16^2 + 10 \cdot 16^1 + 15 \cdot 16^0 = 1199_{10}$

Binär $\leftrightarrow$ Hexadezimal

TECHNISCHE UNIVERSITÄT DARMSTADT

- Nibble-weise umwandeln
- bei least significant bit beginnen
- führende Nullen weglassen oder ergänzen (je nach geforderter Bitbreite)

- $11\ 1010\ 0110\ 1000_2 =$

$3\ A\ 6\ 8_{16}$

Binär $\leftrightarrow$ Hexadezimal

- Nibble-weise umwandeln
- bei least significant bit beginnen
- führende Nullen weglassen oder ergänzen (je nach geforderter Bitbreite)

- $11\ 1010\ 0110\ 1000_2 =$

- $7BF_{16} =$

Binär $\leftrightarrow$ Hexadezimal

TECHNISCHE UNIVERSITÄT DARMSTADT

- Nibble-weise umwandeln
- bei least significant bit beginnen
- führende Nullen weglassen oder ergänzen (je nach geforderter Bitbreite)

- $11\ 1010\ 0110\ 1000_2 = 3A68_{16}$

- $7BF_{16} = 111\ 1011\ 1111_2$

Dezimal $\rightarrow$ Binär

TECHNISCHE UNIVERSITÄT DARMSTADT

- Methode 1
  (links nach rechts):
  Maximale Zweierpotenzen abziehen

53_{10}

$= 32 + 21$

$16 + 5$

$4 + 1$

110101_2

- Methode 2
  (rechts nach links):
  Halbieren mit Rest

53_{10}

$2 \cdot 26 + 1$

$2 \cdot 13 + 0$

$2 \cdot 6 + 1$

$2 \cdot 3 + 0$

$2 \cdot 1 - 1$

$2 \cdot 0 + 1$

Dezimal → Binär

TECHNISCHE UNIVERSITÄT DARMSTADT

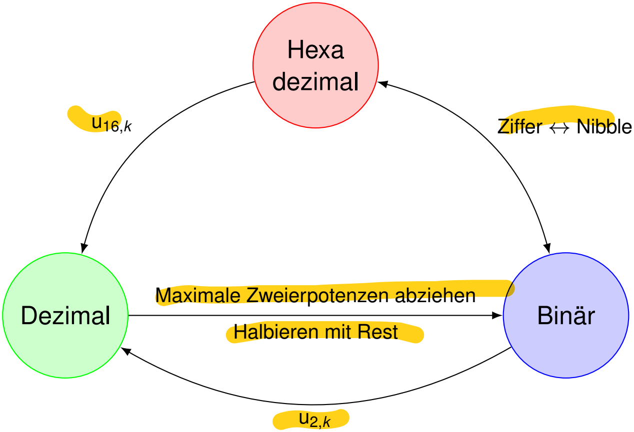
- Methode 1
  (links nach rechts):
  Maximale Zweierpotenzen abziehen

$$
\begin{aligned}
& 53_{10} \\
&= 32 + 21 \\
&= 32 + 16 + 5 \\
&= 32 + 16 + 4 + 1 \\
&= 2^5 + 2^4 + 2^2 + 2^0 \\
&= 110101_2
\end{aligned}
$$

- Methode 2
  (rechts nach links):
  Halbieren mit Rest

$$
\begin{aligned}
& 53_{10} \\
&= 2 \cdot 26 + 1 \\
&= 2 \cdot (2 \cdot 13 + 0) + 1 \\
&= 2 \cdot (2 \cdot (2 \cdot 6 + 1) + 0) + 1 \\
&= 2 \cdot (2 \cdot (2 \cdot (2 \cdot 3 + 0) + 1) + 0) + 1 \\
&= 2 \cdot (2 \cdot (2 \cdot (2 \cdot (2 \cdot 1 + 1) + 0) + 1) + 0) + 1
\end{aligned}
$$

Umrechnen zwischen Zahlensystemen



Zweierpotenzen verinnerlichen!

Addition von vorzeichenlosen Binärzahlen

Schriftliche Addition

TECHNISCHE UNIVERSITÄT DARMSTADT

- Dezimal:

		1	1		Übertrag
	3	7	3	4	Summand
+	5	1	6	8	Summand
	8	9	0	2	Summe

Schriftliche Addition

- Dezimal:

					Übertrag
	3	7	3	4	Summand
+	5	1	6	8	Summand
					Summe

- Binär:

	0	1	1		Übertrag
	1	0	1	1	Summand
+	0	0	1	1	Summand
	1	1	1	0	Summe

Schriftliche Addition

- Dezimal:

		1	1		Übertrag
	3	7	3	4	Summand
+	5	1	6	8	Summand
=	8	9	0	2	Summe

- Binär:

		1	1		Übertrag
	1	0	1	1	Summand
+	0	0	1	1	Summand
=	1	1	1	0	Summe

Addition mit Überlauf

TECHNISCHE UNIVERSITÄT DARMSTADT

- Binär:

	1	1	1			Übertrag
		1	0	1	1	Summand
+		0	1	1	0	Summand
	1	0	0	0	1	Summe

- Digitale Systeme arbeiten i.d.R. mit festen Bitbreiten
  - Langzahlarithmetik nur in Software (Bitbreite nur durch verfügbaren Arbeitsspeicher beschränkt)
  - Overflow-flag zum Signalisieren arithmetischer Ausnahmen in Hardware

Addition mit Überlauf

TECHNISCHE UNIVERSITÄT DARMSTADT

- Binär:

	1	1	1			Übertrag
		1	0	1	1	Summand
+		0	1	1	0	Summand
=	1	0	0	0	1	Summe

Überlauf

- Digitale Systeme arbeiten i.d.R. mit festen Bitbreiten
  - Langzahlarithmetik nur in Software (Bitbreite nur durch verfügbaren Arbeitsspeicher beschränkt)
  - Overflow-flag zum Signalisieren arithmetischer Ausnahmen in Hardware
- Operation (bspw. Addition) läuft über, wenn Ergenis nicht mit der verfügbaren Bitbreite dargestellt werden kann
- für 4 bit Addierer gilt: 11 + 6 = 1

Vorzeichenbehaftete Binärzahlen

Darstellungen von ganzen Zahlen — Dezimal

$$
\begin{aligned}
-5347_{10} &= (7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000) \cdot -1 \\
&= \left(7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3\right) \cdot (-1)^1
\end{aligned}
$$

Darstellungen von ganzen Zahlen — Dezimal

$$
\begin{aligned}
-5347_{10} &= (7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000) \cdot -1 \\
&= \left(7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3\right) \cdot (-1)^1
\end{aligned}
$$

$$
\begin{aligned}
+5347_{10} &= (7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000) \cdot 1 \\
&= \left(7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3\right) \cdot (-1)^0
\end{aligned}
$$

Darstellungen von ganzen Zahlen — Dezimal

$$-5347_{10} = (7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000) \cdot -1$$
$$= \left(7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3\right) \cdot (-1)^1$$

$$+5347_{10} = (7 \cdot 1 + 4 \cdot 10 + 3 \cdot 100 + 5 \cdot 1000) \cdot 1$$
$$= \left(7 \cdot 10^0 + 4 \cdot 10^1 + 3 \cdot 10^2 + 5 \cdot 10^3\right) \cdot (-1)^0$$

- Vorzeichen
  - Ist spezielle Ziffer an höchstwertiger Stelle
  - Kann auch als 0/1 repräsentiert werden

Darstellung von ganzen Zahlen — Verallgemeinerung (Abstraktion)

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition: Betrag und Vorzeichen

Für eine Basis $b \in \mathbb{N} \wedge b \geq 2$ ist $Z_b := \{0, 1, \ldots, b-1\}$ die Menge der verfügbaren Ziffern. Die Funktion $\text{bv}_{b,k}$ bildet eine Ziffernfolge der Breite $k \in \mathbb{N}$ auf eine ganze Zahl ab:

$$\text{bv}_{b,k} : (a_{k-1} \ldots a_1 a_0) \in \{0, 1\} \times Z_b^{k-1} \mapsto (-1)^{a_{k-1}} \cdot \sum_{i=0}^{k-2} a_i \cdot b^i \in \mathbb{Z}$$

Ganze Zahlen als Betrag und Vorzeichen

- Niedrigstwertige Stelle: a_0
- Höchstwertige Stelle: a_{k-1}
- Kleinste darstellbare Zahl: $(-1)^1 \cdot \sum_{i=0}^{k-2}(b-1)\cdot b^i = -(b^{k-1}-1)$
- Größte darstellbare Zahl: $(-1)^0 \cdot \sum_{i=0}^{k-2}(b-1)\cdot b^i = +(b^{k-1}-1)$
- Anzahl der darstellbaren Werte: $2 \cdot b^{k-1} - 1$
- Nicht eindeutig (doppelte Darstellung für Null: ± 0)

Binärdarstellung mit Betrag und Vorzeichen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$bv_{2,4}(1110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^1 = -6_{10}$$

Binärdarstellung mit Betrag und Vorzeichen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$\mathrm{bv}_{2,4}(1110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^1 = -6_{10}$$

$$\mathrm{bv}_{2,4}(0110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^0 = +6_{10}$$

Binärdarstellung mit Betrag und Vorzeichen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$\mathrm{bv}_{2,4}(1110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^1 = -6_{10}$$

$$\mathrm{bv}_{2,4}(0110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^0 = +6_{10}$$

- *Inkompatibel* mit binärer (unsigned) Addition:

$$\begin{array}{cccccccl}
 & 1 & 1 & 1 & & & & \\
 & & 1 & 1 & 1 & 0 & & = -6_{10} \\
+ & & 0 & 1 & 1 & 0 & & = +6_{10} \\
\hline
 & 1 & 0 & 1 & 0 & 0 & &
\end{array}$$

Binärdarstellung mit Betrag und Vorzeichen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$bv_{2,4}(1110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^1 = -6_{10}$$

$$bv_{2,4}(0110_2) = \left(0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2\right) \cdot (-1)^0 = +6_{10}$$

- *Inkompatibel* mit binärer (unsigned) Addition:

$$\begin{array}{cccccccl}
 & 1 & 1 & 1 & & & & \\
 & & 1 & 1 & 1 & 0 & & = -6_{10} \\
+ & & 0 & 1 & 1 & 0 & & = +6_{10} \\
\hline
= & 1 & 0 & 1 & 0 & 0 & & = 4_{10} \ \lightning
\end{array}$$

Darstellung von ganzen Zahlen — Digitaler „Goldstandard“

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition: Zweierkomplement

Die Funktion s_k bildet eine Bitfolge der Breite $k \in \mathbb{N}$ auf eine ganze Zahl ab:

$$s_k : (a_{k-1} \dots a_1 a_0) \in \{0,1\}^k \mapsto a_{k-1} \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} a_i \cdot 2^i \in \mathbb{Z}$$

Ganze Zahlen als Zweierkomplement

- Niedrigstwertige Stelle: a_0
- Höchstwertige Stelle: a_{k-1}
- Kleinste darstellbare Zahl: $1 \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} 0 \cdot 2^i = -2^{k-1}$
- Größte darstellbare Zahl: $0 \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} 1 \cdot 2^i = 2^{k-1} - 1$
- Anzahl der darstellbaren Werte: 2^k
- Eineindeutig (bijektiv) auf Wertebereich $\{-2^{k-1}, \ldots, 2^{k-1} - 1\}$ für festes k

Binärdarstellung im Zweierkomplement

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$s_4(1010_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot -2^3 = -6_{10}$$

Binärdarstellung im Zweierkomplement

- Beispiele

$$s_4(1010_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot -2^3 = -6_{10}$$

$$s_4(0110_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot -2^3 = +6_{10}$$

Binärdarstellung im Zweierkomplement

- Beispiele

$$s_4(1010_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot -2^3 = -6_{10}$$

$$s_4(0110_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot -2^3 = +6_{10}$$

- *Kompatibel* mit binärer (unsigned) Addition:

$$\begin{array}{llccccl}
\mathrm{0V} & & 1 & 1 & 1 & & \\
 & & 1 & 0 & 1 & 0 & = -6_{10} \\
 & + & 0 & 1 & 1 & 0 & = +6_{10} \\
\hline
 & & 0 & 0 & 0 & 0 &
\end{array}$$

Binärdarstellung im Zweierkomplement

TECHNISCHE UNIVERSITÄT DARMSTADT

- Beispiele

$$s_4(1010_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot -2^3 = -6_{10}$$

$$s_4(0110_2) = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot -2^3 = +6_{10}$$

- *Kompatibel* mit binärer (unsigned) Addition:

$$\begin{array}{cccccc l}
 & \color{red}{1} & 1 & 1 & & & \\
 & & 1 & 0 & 1 & 0 & = -6_{10} \\
+ & & 0 & 1 & 1 & 0 & = +6_{10} \\
\hline
= & \color{red}{1} & 0 & 0 & 0 & 0 & = 0_{10} \checkmark
\end{array}$$

- *Kein Überlauf* bei Addition positiver und negativer Zahl gleicher Breite

Dezimal $\rightarrow$ Zweierkomplement

TECHNISCHE UNIVERSITÄT DARMSTADT

- Methode 1 (links nach rechts): Größtmögliche Zweierpotenzen abziehen

  -53_{10}

- Methode 2 (rechts nach links): Betrag negieren = Komplement und Inkrement = Bits kippen und Eins addieren

  -53_{10}

- in beiden Fällen auf korrekte/geforderte Bitbreite achten
- ggf. müssen führende Null(en) schon für Betragsdarstellung eingefügt werden

Dezimal $\rightarrow$ Zweierkomplement

TECHNISCHE UNIVERSITÄT DARMSTADT

- Methode 1 (links nach rechts): Größtmögliche Zweierpotenzen abziehen

$$\begin{aligned}-53_{10} &= -64 + 11\\ &= -64 + 8 + 3\\ &= -64 + 8 + 2 + 1\\ &= -2^6 + 2^3 + 2^1 + 2^0\\ &= 1001011_2\end{aligned}$$

- Methode 2 (rechts nach links): Betrag negieren = Komplement und Inkrement = Bits kippen und Eins addieren

$$\begin{aligned}-53_{10} &= \sim 53_{10} + 1\\ &= \sim 0110101_2 + 1\\ &= \quad 1001010_2 + 1\\ &= \quad 1001011_2\end{aligned}$$

01111

- in beiden Fällen auf korrekte/geforderte Bitbreite achten
- ggf. müssen führende Null(en) schon für Betragsdarstellung eingefügt werden

$$\begin{aligned}-0 &= \sim 0000 + 1\\ &= \quad 1111 + 1\\ &= 1\,\underline{0000}\end{aligned}$$

Negieren = Komplement und Inkrement

TECHNISCHE UNIVERSITÄT DARMSTADT

$$s_k(\sim a_{k-1} \dots a_0) = \sim a_{k-1} \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} \sim a_i \cdot 2^i$$

Negieren = Komplement und Inkrement

TECHNISCHE UNIVERSITÄT DARMSTADT

$\sim a = 1 - a$

$$s_k(\sim a_{k-1} \dots a_0) = \sim a_{k-1} \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} \sim a_i \cdot 2^i$$

$$= (1 - a_{k-1}) \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} (1 - a_i) \cdot 2^i$$

Negieren = Komplement und Inkrement

TECHNISCHE UNIVERSITÄT DARMSTADT

$$s_k(\sim a_{k-1} \dots a_0) = \sim a_{k-1} \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} \sim a_i \cdot 2^i$$

$$= (1 - a_{k-1}) \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} (1 - a_i) \cdot 2^i$$

$$= \underbrace{\left(1 \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} 1 \cdot 2^i\right)} - \underbrace{\left((a_{k-1}) \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} a_i \cdot 2^i\right)}$$

Negieren = Komplement und Inkrement

TECHNISCHE
UNIVERSITÄT
DARMSTADT

$$s_k(\sim a_{k-1} \dots a_0) = \sim a_{k-1} \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} \sim a_i \cdot 2^i$$

+1

$$= (1 - a_{k-1}) \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} (1 - a_i) \cdot 2^i$$

$$= \underbrace{\left(1 \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} 1 \cdot 2^i\right)}_{-1} - \underbrace{\left((a_{k-1}) \cdot (-2^{k-1}) + \sum_{i=0}^{k-2} a_i \cdot 2^i\right)}_{s_k(a_{k-1} \dots a_0)}$$

Bitbreitenerweiterung

- notwendig, um unterschiedliche breite Bitfolgen zu addieren

Bitbreitenerweiterung

TECHNISCHE UNIVERSITÄT DARMSTADT

- notwendig, um unterschiedliche breite Bitfolgen zu addieren
- *zero extension*:
  - Auffüllen mit führenden Nullen für vorzeichenlose Darstellung

$$\mathrm{u}_{2,k+1}(0a_{k-1}\ldots a_0) = 0\cdot 2^k + \sum_{i=0}^{k-1} a_i\cdot 2^i = \mathrm{u}_{2,k}(a_{k-1}\ldots a_0)$$

Bitbreitenerweiterung

TECHNISCHE UNIVERSITÄT DARMSTADT

- notwendig, um unterschiedliche breite Bitfolgen zu addieren
- *zero extension*:
  - Auffüllen mit führenden Nullen für vorzeichenlose Darstellung

$$u_{2,k+1}(0a_{k-1}\dots a_0) = 0\cdot 2^k + \sum_{i=0}^{k-1} a_i \cdot 2^i = u_{2,k}(a_{k-1}\dots a_0)$$

- *signed extension*:
  - Auffüllen mit Wert des Vorzeichen-Bits für Zweierkomplement Darstellung

$$\begin{aligned}
s_{k+1}(a_{k-1}a_{k-1}\dots a_0) &= a_{k-1}\cdot \underbrace{(-2^k)}_{2\cdot(-2^{k-1})} + a_{k-1}\cdot 2^{k-1} + \sum_{i=0}^{k-2} a_i\cdot 2^i \\
&= a_{k-1}\cdot\left(-2^{k-1} - 2^{k-1} + 2^{k-1}\right) + \sum_{i=0}^{k-2} a_i\cdot 2^i \\
&= s_k(a_{k-1}\dots a_0)
\end{aligned}$$

Bitbreitenerweiterung — Beispiel

- -5_{10} im Zweierkomplement von 4 auf 8 Bit erweitern:

Bitbreitenerweiterung — Beispiel

TECHNISCHE UNIVERSITÄT DARMSTADT

- -5_{10} im Zweierkomplement von 4 auf 8 Bit erweitern:

$$
\begin{aligned}
5_{10} &= 0101_2 \\
\Rightarrow -5_{10} &= \sim 0101_2 + 1 \\
&= 1010_2 + 1 \\
&= 1011_2 \\
&= 11111011_2
\end{aligned}
$$

$$
\begin{aligned}
\sim 11111011_2 + 1 &= 00000100_2 + 1 \\
&= 00000101_2 \\
&= 5_{10}
\end{aligned}
$$

Vergleich der binären Zahlendarstellungen für k=4

TECHNISCHE UNIVERSITÄT DARMSTADT

$\mathbb{Z}$	Vorzeichenlos: $u_{2,k}$	Betrag/Vorzeichen: $bv_{2,k}$	Zweierkomplement: s_k
	$\{0, \dots, 2^k - 1\}$	$\{-2^{k-1}+1, \dots, 2^{k-1} - 1\}$	$\{-2^{k-1}, \dots, 2^{k-1} - 1\}$
15	1111		
14	1110		
13	1101		
12	1100		
11	1011		
10	1010		
9	1001		
8	1000		
7	0111	0111	0111
6	0110	0110	0110
5	0101	0101	0101
4	0100	0100	0100
3	0011	0011	0011
2	0010	0010	0010
1	0001	0001	0001
0	0000	0000 1000	0000
−1		1001	1111
−2		1010	1110
−3		1011	1101
−4		1100	1100
−5		1101	1011
−6		1110	1010
−7		1111	1001
−8			1000

Weitere Rechenbeispiele

TECHNISCHE UNIVERSITÄT DARMSTADT

Wertebereiche

- $u_{2,7}$
- $bv_{2,6}$
- $s_{2,10}$

Wertebereiche

TECHNISCHE UNIVERSITÄT DARMSTADT

- $u_{2,7} \mapsto \{0, \dots, 127\}$
- $bv_{2,6} \mapsto \{-31, \dots, 31\}$
- $s_{2,10} \mapsto \{-512, \dots, 511\}$

Binär → (Hexa-)Dezimal

- $u_{2,6}(110011_b) =$

- $bv_{2,6}(110011_b) =$

- $s_{2,6}(110011_b) =$

- $hex(110011_b) =$

Binär → (Hexa-)Dezimal

- $u_{2,6}(110011_b) = 32 + 16 + 2 + 1 = 51_{10}$

- $bv_{2,6}(110011_b) = -(16 + 2 + 1) = -19_{10}$

- $s_{2,6}(110011_b) = -32 + 16 + 2 + 1 = -13_{10}$

- $hex(110011_b) = 33_{16}$

Dezimal → Zweierkompliment, Addition

- 8 Bit Zweierkomplement von 60_{10} =
- 6 Bit Zweierkomplement von -20_{10} =
- binär addieren:
- Überlauf?

Dezimal $\rightarrow$ Zweierkompliment, Addition

TECHNISCHE UNIVERSITÄT DARMSTADT

- 8 Bit Zweierkomplement von 60_{10} = 00111100_2
- 6 Bit Zweierkomplement von -20_{10} = 101100_2
- binär addieren: $00111100_2 + 11101100_2 = 00101000_2 = 40_{10}$
- Überlauf? Nur bei reinen unsigned-Addierern ohne Überlauf-Maskierung bei ungleichen Vorzeichen

Nabla-Katalog für Rechnerarchitektur

https://nabla.algo.informatik.tu-darmstadt.de

Binäre Addition

Addition und Subtraktion von Bits

Binary number: addition

Bereitgestellt von Wikipedia.org

W

- Leicht 0 0
- Mittel 0 0
- Schwer 0 0

Start

Erstellungsdatum: Freitag, 26. Mai 2017

Autor: Lukas Weber

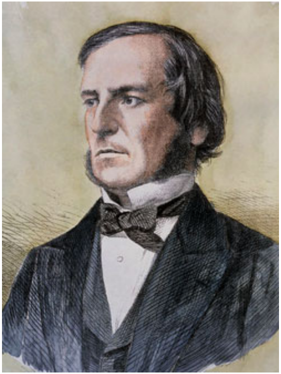
Logikgatter

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Schichtenmodell eines Computers

Anwendungs-software	Programme
Betriebs-systeme	Gerätetreiber
Architektur	Befehle Register
Mikro-architektur	Datenpfade Steuerung
Logik	Addierer Speicher
Digital-schaltungen	UND Gatter Inverter
Analog-schaltungen	Verstärker Filter
Bauteile	Transistoren Dioden
Physik	Elektronen

George Boole, 1815 - 1864



- In einfachen Verhältnissen geboren
- Brachte sich selbst Mathematik bei
- Professor am Queen's College in Irland
- „An Investigation of the Laws of Thought" (1854)

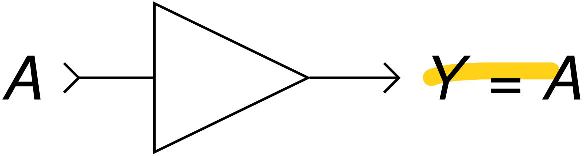
$\Rightarrow$ Grundlegende Logische Variablen und Operationen

Logische Operationen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Verknüpfen binäre Werte: $\mathbb{B}^n \to \mathbb{B}^k$
- zunächst $k = 1$
- Beispiele für
  - $n = 1$: NOT
  - $n = 2$: AND, OR, XOR
  - $n = 3$: MUX
- Charakterisierung durch Wahrheitswertetabellen

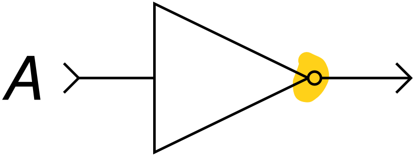
BUF : $\mathbb{B} \to \mathbb{B}$



A	Y
0	0
1	1

NOT : $\mathbb{B} \rightarrow \mathbb{B}$

TECHNISCHE
UNIVERSITÄT
DARMSTADT



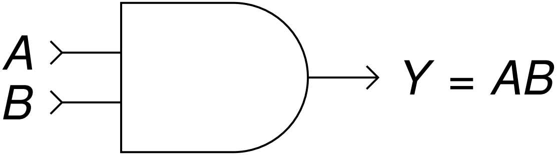
$$Y = \overline{A}$$

A	Y
0	1
1	0

Alternativ: $Y = !A = \sim A$

AND : $\mathbb{B}^2 \rightarrow \mathbb{B}$

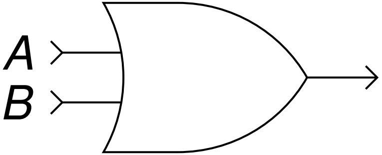
TECHNISCHE UNIVERSITÄT DARMSTADT



A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Alternativ: $Y = A \cdot B = A \,\&\, B = A \wedge B$

OR : $\mathbb{B}^2 \rightarrow \mathbb{B}$

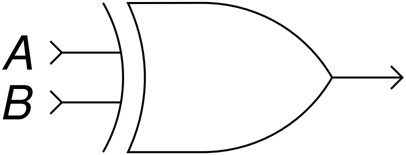


$$Y = A + B$$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

Alternativ: $Y = A|B = A \vee B$

XOR : $\mathbb{B}^2 \rightarrow \mathbb{B}$



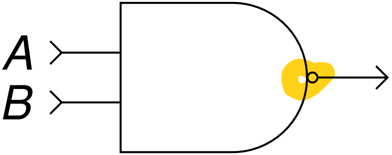
$$Y = A \oplus B$$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Alternativ: $Y = A\hat{}B$

NAND : $\mathbb{B}^2 \to \mathbb{B}$

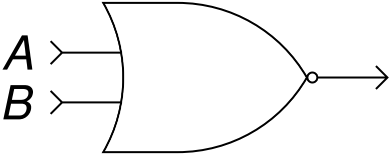
TECHNISCHE UNIVERSITÄT DARMSTADT



$$Y = \overline{AB}$$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

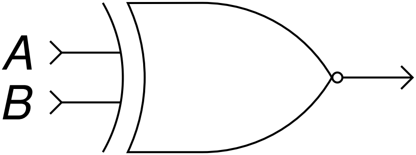
NOR : $\mathbb{B}^2 \to \mathbb{B}$



$$Y = \overline{A + B}$$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

XNOR : $\mathbb{B}^2 \rightarrow \mathbb{B}$



$$Y = \overline{A \oplus B}$$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Entspricht Test auf Gleichheit

XOR mit mehreren Eingängen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Paritätsfunktion
  - Erkennt gerade oder ungerade Anzahl von Eingängen mit Wert 1
- XOR
  - Ungerade Paritätsfunktion
  - Liefert 1 am Ausgang, wenn ungerade Anzahl von Eingängen den Wert 1 haben

A	B	C	$A \oplus B \oplus C$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Anwendung der Paritätsfunktion
Fehlererkennung

- Zur Fehlererkennung bei Übertragung oder Speicherung
  - „Sender“ := Schreiboperation
  - „Empfänger“ := Leseoperation
- Z. B. 7 Bit-Kodierung auf eine redundante 8 Bit-Kodierung erweitert
  - 8-tes Bit durch XOR-Schaltung erzeugt: Paritätsbit
  - genau dann = 1, wenn an Eingängen (7 Bit) ungerade Zahl von Einsen
  - erzeugte Kodierung hat immer gerade Zahl von Einsen (even parity)
- Bei Verfälschung in
  - nur einem Bit $\Rightarrow$ Zahl der Einsen nicht mehr gerade $\Rightarrow$ Fehler
  - gerader Anzahl von Bits: Fehler nicht erkennbar
- Alternativ: auch Paritätsbit für ungerade Zahl von Einsen (odd parity)

Anwendung der Paritätsfunktion
Fehlererkennung

- Parity Bit: Zählt Anzahl der 1en, bspw.
  - Sender: `10 111 001 PB: 1`
  - `Empfänger: 10 101 001 PB: 0`
  - `Empfänger(2): 10 001 001 PB: 1`
- 1 bit Fehler erkennbar, 2 bit Fehler nicht erkennbar
- Korrekturen nicht möglich ⇒ Wiederholte Übertragung des Datums
- Variante: weitere Prüfbits (Längs- und Querparität)

Zusammenfassung

1110101001111011010000000101100000101011
1110011011100101110011011101111000101111
0010101100000011000011010011001110101010
1010010010000101111010001010101001010100
1001011101010110111011110110000001000101
0111010111000011111010101001100000001010
0001101011110011001111011011110101110000
110111111010001111**1000**100011000011110010
1010110111011111011000010111111111010011
0110110001000001100100000001100100110111
1000001110110001100110001000110011111001
1010011010111100111011011001110111001100
0011101000111011100110111010010111111000
1111001111100110101110000101101110101011
0011110001101110100110000010100011011111
1000111010100000101111001010101110111000

Zusammenfassung und Ausblick

TECHNISCHE UNIVERSITÄT DARMSTADT

- Zahlensysteme: Bitfolgen $\leftrightarrow$ (ganze) Zahlen
  - Dezimal-, Binär-, Hexadezimalzahlen
  - Darstellung
  - Umrechnung
  - Addition von Binärzahlen
  - Vorzeichenbehaftete Binärzahlen
- Logikgatter
  - Darstellung
  - Wahrheitswertetabellen

Zusammenfassung und Ausblick

TECHNISCHE UNIVERSITÄT DARMSTADT

- Zahlensysteme: Bitfolgen $\leftrightarrow$ (ganze) Zahlen
  - Dezimal-, Binär-, Hexadezimalzahlen
  - Darstellung
  - Umrechnung
  - Addition von Binärzahlen
  - Vorzeichenbehaftete Binärzahlen
- Logikgatter
  - Darstellung
  - Wahrheitswertetabellen

- Nächste Vorlesung behandelt
  - Physikalische Realisierung von Logikgattern