



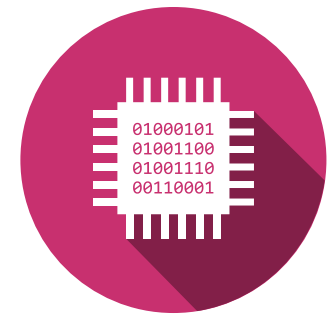
# Digitales Design (DiD)

## Numerische Darstellung und Codes

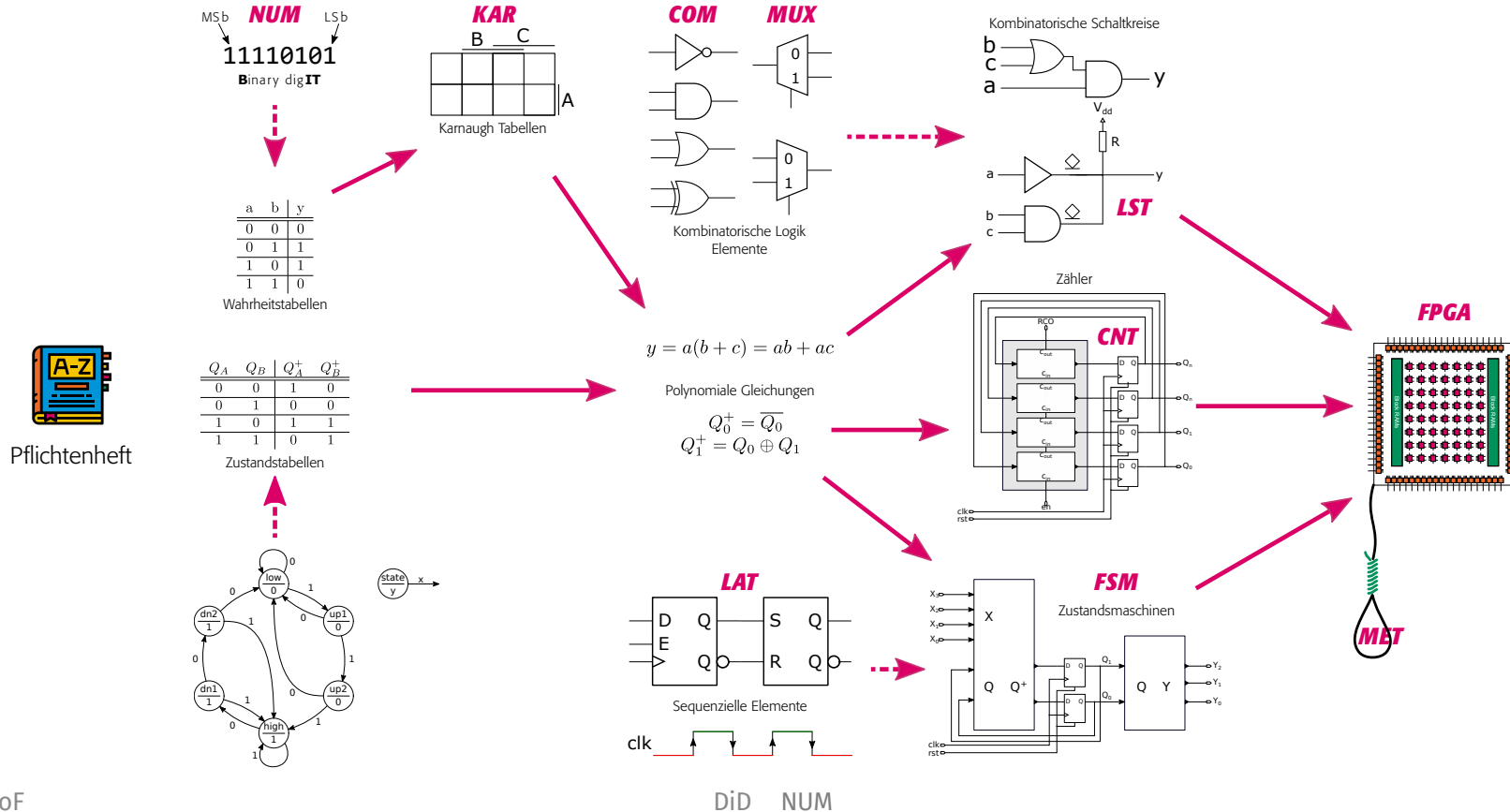
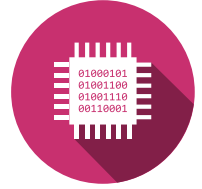
### NUM

Studiengang Systemtechnik  
Studiengang Energie und Umwelttechnik  
Studiengang Informatik und Kommunikationssysteme

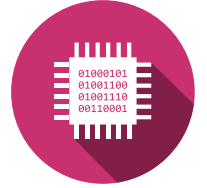
Silvan Zahno [silvan.zahno@hevs.ch](mailto:silvan.zahno@hevs.ch)  
Christophe Bianchi [christophe.bianchi@hevs.ch](mailto:christophe.bianchi@hevs.ch)  
François Corthay [francois.corthay@hevs.ch](mailto:francois.corthay@hevs.ch)



# Aktueller Inhalt des Themas im Kurs

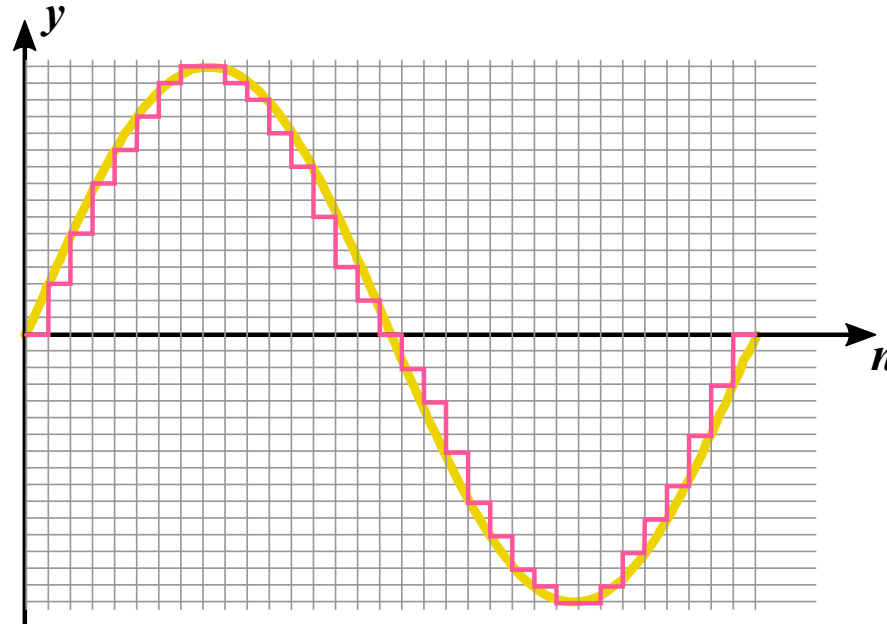
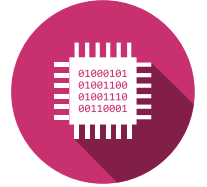


# Inhalt



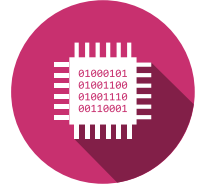
- **Zahlensysteme**
  - Dezimalsystem
  - Binärsystem
  - Hexadezimalsystem
- Umwandlung von Zahlensystemen
- Operationen auf Logikzahlen
- Codes
- Darstellung von Arithmetischen Zahlen (Vorzeichenbehaftet)

# Wechsel von Analog zu Digital

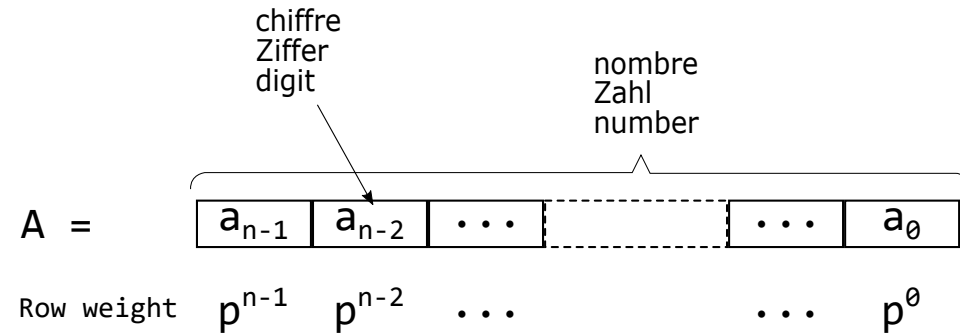


- Diskrete Zeiten und Amplituden
- Frequenzabhängiges Zeitintervall
- $n$ : Stichprobennummer (z.B. 32 Stichproben über den Zeitraum)
- $y$ : Signalamplitude (Beispiel: 32 mögliche Werte)

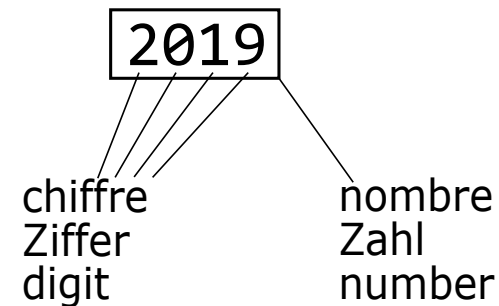
# Allgemeine Darstellung von Ganzen Zahlen



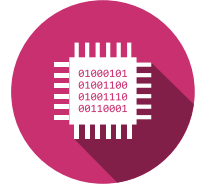
Nach dem Positionssystem besteht eine Zahl aus Ziffern



In der Basis 10 ,  $p=10$



# Dezimalsystem



Im Allgemeinen verwendet der Mensch die Basis 10

10 Symbole : 0,1,2,3,4,5,6,7,8 und 9

Jede Position entspricht einer Potenz von 10

⇒ 1, 10, 100, 1000, ...

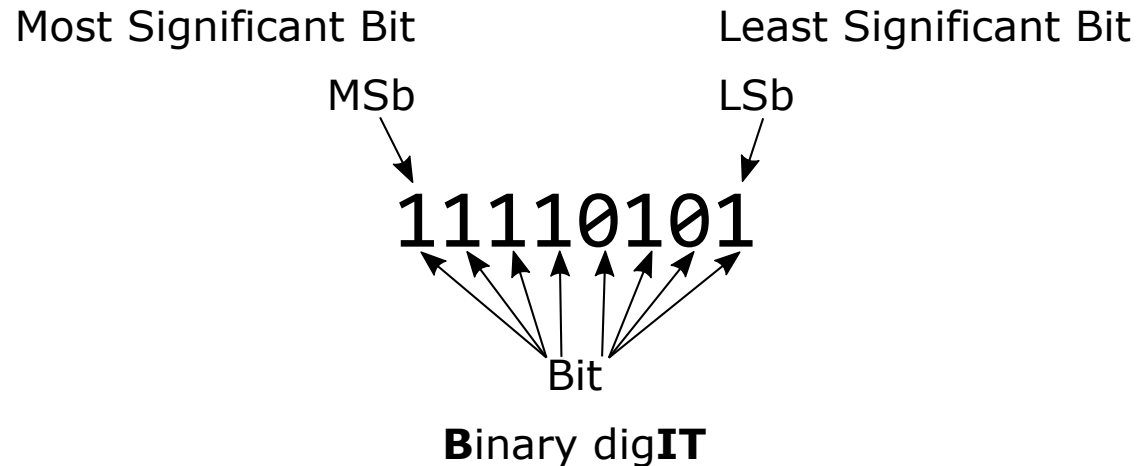
Beispiel:  $245_{10} = 2 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0$

# Binärsystem

## Das BIT



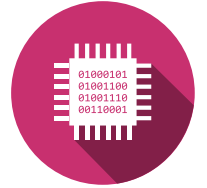
- Computer arbeiten in Basis 2
- 2 Symbole : 0 und 1
- Eine Binärzahl (0/1) wird ein BIT (**B**inary dig**IT**) genannt



- Beispiel:  $11110101_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$

# Binärsystem

## Das Byte



- 8 BIT bilden ein Byte (octet)
  - Rein historisch

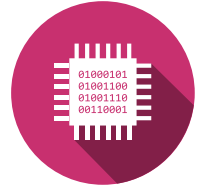
11110101

8 Bit = 1 Byte

- Using IEC standard:
  - 1 KiB = 1'024 bytes (Note: big K)
  - 1 MiB = 1'024 KiB = 1'048'576 bytes
  - 1 GiB = 1'024 MiB = 1'048'576 KiB = 1'073'741'824 bytes
- Using SI standard:
  - 1 kB = 1'000 bytes (Note: small k)
  - 1 MB = 1'000 kB = 1,000,000 bytes
  - 1 GB = 1'000 MB = 1'000'000 KB = 1'000'000'000 bytes



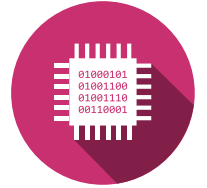
# Hexadezimalsystem



- 16 Symbole
- 2er Potenz ( $16 = 2^4$ )  
Bit-Gruppierung von 4
- Ermöglicht das Schreiben von Binärzahlen
- in kompakter Form

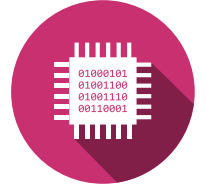
Decimal	Hexadecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

# Schreibweisen



- Dezimal
  - $9_{10}$ ,  $9_d$
- Binär
  - $0b1001$ ,  $1001_2$ ,  $1001_b$
- Hexadezimal
  - $0x9$ ,  $9_{16}$ ,  $9_h$

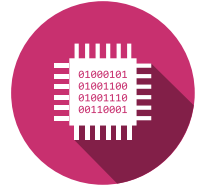
# Inhalt



- Zahlensysteme
- **Umwandlung von Zahlensystemen**
  - Binär  $\Leftrightarrow$  Dezimal
  - Dezimal  $\Leftrightarrow$  Binär
  - Hexadezimal  $\Leftrightarrow$  Binär
  - Binär  $\Leftrightarrow$  Hexadezimal
  - Hexadezimal  $\Leftrightarrow$  Dezimal
  - Dezimal  $\Leftrightarrow$  Hexadezimal
- Operationen auf Logikzahlen
- Codes
- Darstellung von Arithmetischen Zahlen (Vorzeichenbehaftet)

# Umwandlung

Binär  $\Rightarrow$  Dezimal

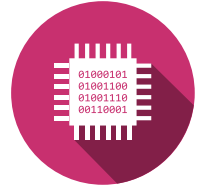


Addition der Gewichte der Positionen (2er-Potenz), bei denen es eine 1 gibt

$$\begin{array}{cccccccc} 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ \swarrow & \swarrow & \swarrow & \swarrow & & \swarrow & \swarrow & \\ 128 & +64 & +32 & +16 & & +4 & +1 & = 245_{10} \end{array}$$

# Umwandlung

## Dezimal $\Rightarrow$ Binär (kleine Zahlen)



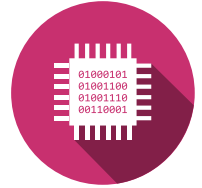
- Suchen Sie nach den zu berücksichtigenden 2er-Potenzen, so dass die Summe die angegebene Dezimalzahl ergibt.
- Ausgehend von höherwertigen Bits

$$77_{10} = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1$$

Diagram illustrating the binary conversion of the decimal number 77. The powers of 2 are listed above the binary digits 0 1 0 0 1 1 0 1. Dashed lines connect the powers of 2 to the corresponding binary digits: 128 to 0, 64 to 1, 32 to 0, 16 to 0, 8 to 1, 4 to 1, 2 to 0, and 1 to 1. Solid lines connect the powers of 2 to the corresponding binary digits: 128 to 0, 64 to 1, 32 to 0, 16 to 0, 8 to 1, 4 to 1, 2 to 0, and 1 to 1.

# Umwandlung

## Dezimal $\Rightarrow$ Binär (grosse Zahlen)



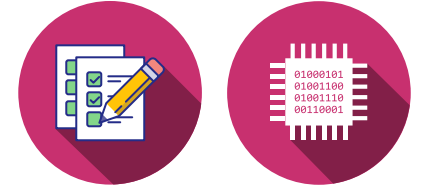
- Wiederholen Sie die Division durch 2 der angegebenen Dezimalzahl, bis der Quotient 0 ist.

- Der Rest jeder Division ergibt ein zusätzliches Bit der Zahl, beginnend mit dem niederwertigsten Bit.

245	/ 2	=	122	+	remainder	1	
122	/ 2	=	61	+	remainder	0	
61	/ 2	=	30	+	remainder	1	
30	/ 2	=	15	+	remainder	0	
15	/ 2	=	7	+	remainder	1	
7	/ 2	=	3	+	remainder	1	
3	/ 2	=	1	+	remainder	1	
1	/ 2	=	0	+	remainder	1	

245<sub>10</sub> = 111110101<sub>2</sub>

# Aufgabe



- 3.1.c) Führen Sie die Umwandlung der folgenden reinen Binärzahl durch :

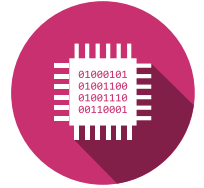
□  $01001010_2 = ?_{10}$

- 3.2.a) Führen Sie folgende Dezimalzahl-Konvertierung durch :

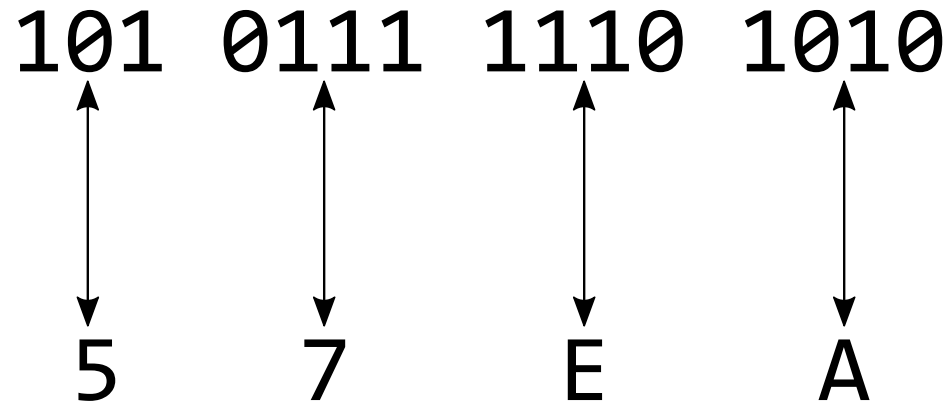
□  $125_{10} = ?_2$

# Umwandlung

## Hexadezimal $\Rightarrow$ Binär



- Gruppierung der Bits in Vierergruppen, beginnend mit dem niedrigstwertigen Bit und Umwandlung dieser Vierergruppen in ihr hexadezimal Äquivalent





# Aufgabe



- 3.4.c) Führen Sie die Umwandlung der folgenden reinen Binärzahl durch :

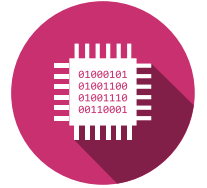
□  $11101011_2 = ?_{16}$

- 3.3.c) Führen Sie die Konvertierung der folgenden hexadezimalen Zahl durch :

□  $AB3D_{16} = ?_2$

# Umwandlung

## Hexadezimal $\Rightarrow$ Dezimal

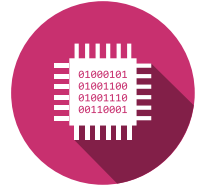


- Addition der Produkte, die durch jede hexadezimale Ziffer gebildet werden, und ihr entsprechendes Positionsgewicht

$$\begin{array}{cc} \text{F} & \text{5} \\ \downarrow & \downarrow \\ 15 * 16^1 + 5 * 16^0 = 245_{10} \end{array}$$

# Umwandlung

## Dezimal $\Rightarrow$ Hexadezimal

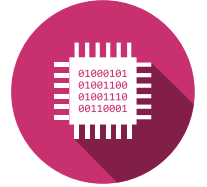


- Wiederholen Sie die Division der angegebenen Dezimalzahl durch 16, bis der Quotient 0 ist.
- Die hexadezimale Zahl besteht dann aus den aufeinanderfolgenden Resten der einzelnen Teilungen, wobei der erste gefundene Rest der Ziffer mit niedrigem Gewicht und der letzte Rest der Ziffer mit hohem Gewicht entspricht.

$$\begin{array}{lcl} 245 & / & 16 = 15 + \text{remainder } 5 \\ 15 & / & 16 = 0 + \text{remainder } 15 \end{array}$$

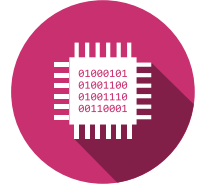
$245_{10} = F5_{16}$

# Inhalt



- Zahlensysteme
- Umwandlung von Zahlensystemen
- **Operationen auf Logikzahlen**
  - Addition
  - Subtraktion
  - Multiplikation
- Codes
- Darstellung von Arithmetischen Zahlen (Vorzeichenbehaftet)

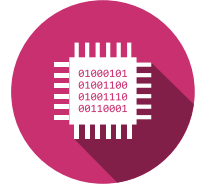
# Addition Binärer Zahlen



- Wie im Dezimalsystem: von LSb nach MSb mit Fortschreibung des Übertrages in der nachfolgenden Spalte
- Max. 1 zusätzliches Bit

$$\begin{array}{r} 2 \\ + 6 \\ \hline 8 \end{array} \qquad \begin{array}{r} 11 \\ 0010 \\ + 0110 \\ \hline 1000 \end{array}$$

# Subtraktion Binärer Zahlen

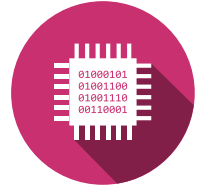


- Wie im Dezimalsystem: von LSb nach MSb mit Propagierung der Anfrage an die folgende Spalte

$$\begin{array}{r} 11 \\ - 4 \\ \hline 7 \end{array} \quad \begin{array}{r} 1011 \\ - 0100 \\ - 1 \\ \hline 0111 \end{array}$$

- Subtraktion auch mit Addition möglich  $11-4 = 11+(-4)$

# Aufgabe



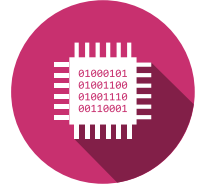
- 4.1.b) Führen Sie die folgende Operation im Binärsystem durch :

□  $00001111_2 + 01011010_2$

- 4.2.c) Führen Sie die folgende Operation im Binärsystem durch :

□  $00110100_2 - 00101000_2$

# Multiplikation Binärer Zahlen

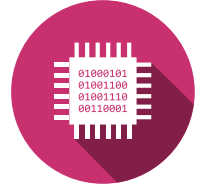


- Wie im Dezimalsystem: durch partielle Multiplikationen und Additionen. Partielle Multiplikationen sind im Binärsystem auf Verschiebungen nach links vom ersten Multiplikator beschränkt.

11	1011	1 <sup>st</sup> multiplier
x 13	x 1101	2 <sup>nd</sup> multiplier
143	00000000	Initialisation
	+ 1011	
	00001011	1 <sup>st</sup> product
	+ 0000	Multiplier shift
	00001011	2 <sup>nd</sup> product
	+ 1011	Multiplier shift
	00110111	3 <sup>rd</sup> product
	+ 1011	Multiplier shift
	10001111	Result

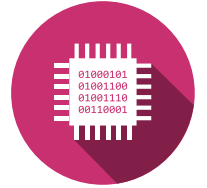


# Inhalt

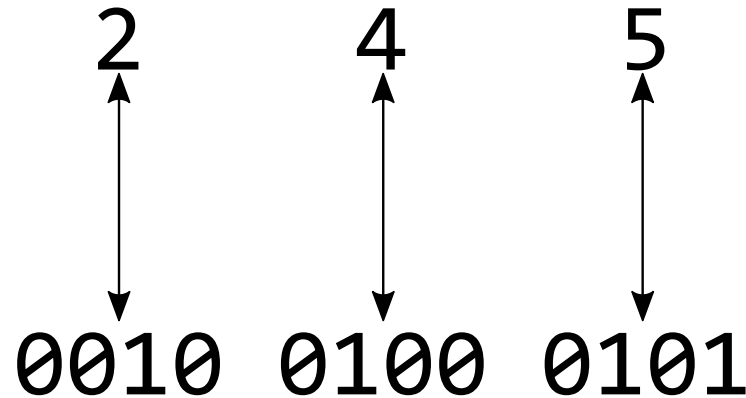


- Zahlensysteme
- Umwandlung von Zahlensystemen
- Operationen auf Logikzahlen
- **Codes**
  - Binär Codierte Dezimalzahlen (BCD)
  - Gray Code
- Darstellung von Arithmetischen Zahlen (Vorzeichenbehaftet)

# Binär Codierte Dezimalzahlen (BCD)

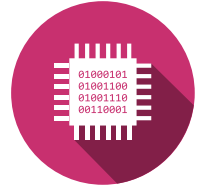


- Jede Stelle einer Dezimalzahl wird durch ihr binäres Äquivalent dargestellt
- Es braucht 4 Bits, um jede Dezimalstelle von 0 bis 9 zu kodieren.
- Einsatz für Displaysysteme



- Komplexere Arithmetik!

# Gray code



- Ein besonderer linearer Code, der so gestaltet ist, dass sich beim Wechsel von einem Wort zum nächsten nur eine Ziffer der Zahl ändert.
- Reflektierter Graybinärkode
- Einsatz bei Positionsgebern

$$b_3 = g_3$$

$$b_2 = b_3 \oplus g_2 = g_3 \oplus g_2$$

$$b_1 = b_2 \oplus g_1 = g_3 \oplus g_2 \oplus g_1$$

$$b_0 = b_1 \oplus g_0 = g_3 \oplus g_2 \oplus g_1 \oplus g_0$$

$$g_3 = b_3$$

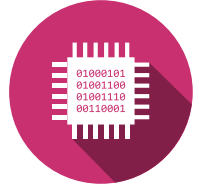
$$g_2 = b_3 \oplus b_2$$

$$g_1 = b_2 \oplus b_1$$

$$g_0 = b_1 \oplus b_0$$

Decimal	Binary	Gray	
0	0000	0000	
1	0001	0001	
2	0010	0011	Mirror
3	0011	0010	
4	0100	0110	Mirror
5	0101	0111	
6	0110	0101	
7	0111	0100	Mirror
8	1000	1100	
9	1001	1101	
10	1010	1111	
11	1011	1110	
12	1100	1010	
13	1101	1011	
14	1110	1001	
15	1111	1000	

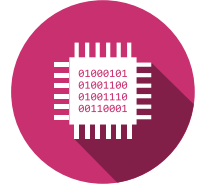
# Gray code



0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

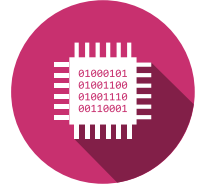
0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

# Inhalt



- Zahlensysteme
- Umwandlung von Zahlensystemen
- Operationen auf Logikzahlen
- Codes
- **Darstellung von Arithmetischen Zahlen (Vorzeichenbehaftet)**
  - Vorzeichen-Grösse
  - Biased Notation
  - Einer-Komplement (1<sup>st</sup>-Complement)
  - Zweier-Komplement (2<sup>nd</sup>-Complement)

# Vorzeichen-Grösse



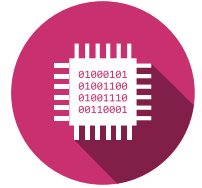
- Positive Zahlen = logische Zahlen
- Positive oder negative Zahlen = arithmetische Zahlen
- der Zahl ein Bit voranstellen, dessen Wert z.B. 0 ist, wenn das Vorzeichen positiv ist und 1, wenn das Vorzeichen negativ ist.

	S							
+1	0	0	0	0	0	0	0	1
-1	1	0	0	0	0	0	0	1

$$-(2^{n-1} - 1) \leq A \leq 2^{n-1} - 1$$
$$-7 \leq A \leq 7 \quad n=4$$

- Aber ... doppelte Darstellung der 0

# Biased-Notation



- Kodierung einer positiven oder negativen ganzen Zahl  $A$  als eine Zahl  $N$ , so dass  $N = A + R$ , wobei  $R = 2^{n-1} - 1$  eine positive Vorspannung ist, die so gewählt wird, dass  $N$  immer positiv ist.

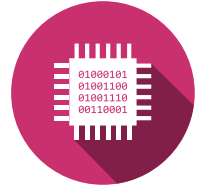
Representation	Min		...		Zero		...		Max
Shift decimal	0	1		126	127	128		254	255
Decimal	-127	-126		-1	0	1		127	128
Shift binary	00000000	00000001		01111110	01111111	10000000		11111110	11111111

$$-(2^{n-1} - 1) \leq A \leq 2^{n-1}$$

$$-7 \leq A \leq 8 \quad n=4$$

- Eine einzige Darstellung der 0
- Aber ... nicht-ideale Darstellung von positiven Zahlen ( $\neq$  binäre Darstellung der logischen Zahl), in der Praxis nicht verwendet

# 1<sup>st</sup>-Komplement



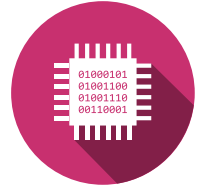
- voreingenommene Darstellung, aber nur auf negative Zahlen angewandt
- positive Zahlen bleiben unverändert
- $R = 2^n - 1$
- das Komplement zu 1 wird sehr einfach durch logische Negation aller Bits der Zahl berechnet

$$\begin{array}{rcl} 125 & & = 01111101 \\ \downarrow & & \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \\ -125 & R - 125 = 255 - 125 = 130 = & 10000010 \end{array}$$
$$-(2^{n-1} - 1) \leq A \leq 2^{n-1} - 1$$
$$-7 \leq A \leq 7 \quad n=4$$

- Aber ... doppelte Darstellung der 0
- In der Praxis wenig genutzt

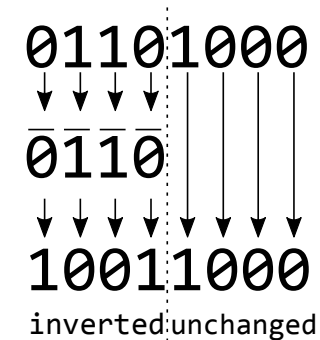


# 2<sup>nd</sup>-Komplement

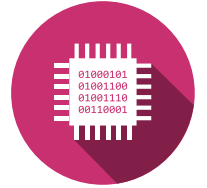


- Wie Einer-Komplement aber mit  $R = 2^n$ , also Einer-Komplement +1.
- das Zweier-Komplement wird sehr einfach berechnet, indem man die Zahl von rechts nach links durchsucht, indem man folgende Regel anwendet: alle Bits, die bis und mit der ersten 1 gefunden werden, werden behalten, alle weitere werden invertiert.

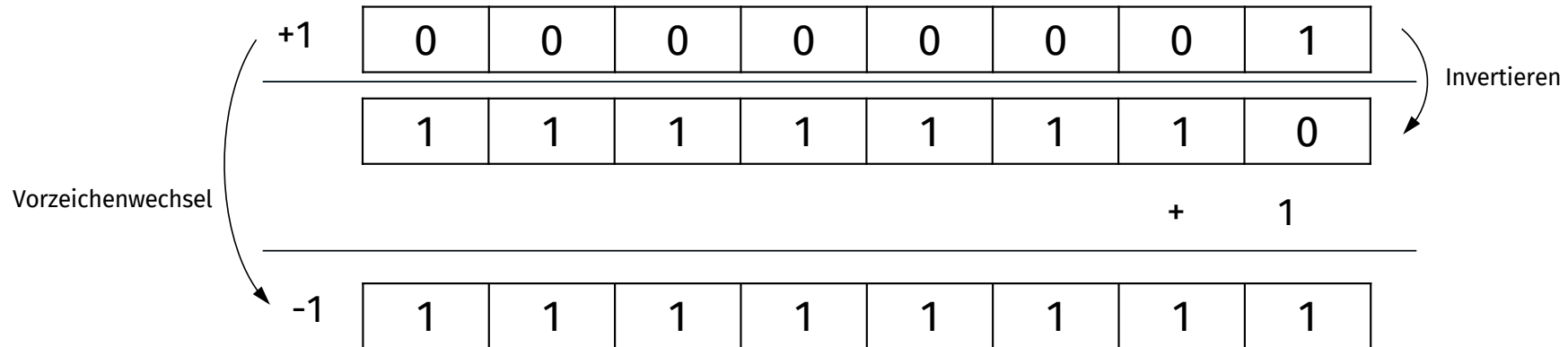
$$-(2^{n-1} - 1) \leq A \leq 2^{n-1}$$
$$-8 \leq A \leq 7 \quad n=4$$



# 2<sup>nd</sup>-Komplement

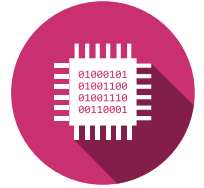


- Extrem einfach für Addition und Subtraktion
- Vorzeichenwechsel

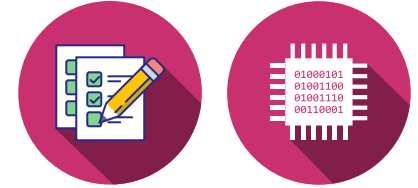


# 2<sup>nd</sup>-Komplement Bereich

	0	0	0	
7	1	1	1	
6	1	1	0	
5	1	0	1	
4	1	0	0	
3	0	1	1	+3
2	0	1	0	+2
1	0	0	1	+1
0	0	0	0	+0
	1	1	1	-1
	1	1	0	-2
	1	0	1	-3
	1	0	0	-4
	0	1	1	
	DiD	NUM		



# Aufgabe

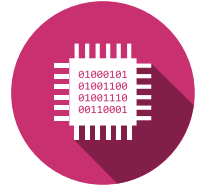


- 6.1.d) Geben Sie Darstellung in Vorzeichen-Amplitude, 1-Komplement und 2-Komplement über acht Bits der folgenden dezimalen und reinen Binärzahlen zu geben :

□  $00011010_2$

□  $-104_{10}$

# Referenzen



- [Max95] (englisch) Numerierungssysteme
- [Alm89] (englisch) Beispiele Operationen
- [Wak00] (englisch) Beispiele Operationen, Multiplikation
- [Beu01] (deutsch) Beispiele Operationen
- [Die88] (englisch) Beispiele Operationen

WHY ARE THERE MIRRORS ABOVE BEDS

WHY DO I SAY UH

WHY IS SEA SALT BETTER

WHY ARE THERE TREES IN THE MIDDLE OF FIELDS

WHY IS THERE NOT A POKEMON MMO

WHY IS THERE LAUGHING IN TV SHOWS

WHY ARE THERE DOORS ON THE FREEWAY

WHY ARE THERE SO MANY SUCHOST-EXE RUNNING

WHY AREN'T ANY COUNTRIES IN ANTARCTICA

WHY ARE THERE SCARY SOUNDS IN MINECRAFT

WHY IS THERE KICKING IN MY STOMACH

WHY ARE THERE TWO SLASHES AFTER HTTP

WHY ARE THERE CELEBRITIES

WHY DO SNAKES EXIST

WHY DO OYSTERS HAVE PEARLS

WHY ARE DUCKS CALLED DUCKS

WHY DO THEY CALL IT THE CLAP

WHY ARE KYLE AND CARTMAN FRIENDS

WHY IS THERE AN ARROW ON AANG'S HEAD

WHY ARE TEXT MESSAGES BLUE

WHY ARE THERE MUSTACHES ON CLOTHES

WHY WUBA LUBBA DUB DUB MEANING

WHY IS THERE A WHALE AND A POT FALLING

WHY ARE THERE SO MANY BIRDS IN SWISS

WHY IS THERE SO LITTLE RAIN IN WALLIS

WHY IS WALLIS WEATHER FORECAST ALWAYS WRONG

WHY ARE THERE MALE AND FEMALE BIKES

WHY ARE THERE BRIDESMAIDS

WHY DO DYING PEOPLE REACH UP

HOW FAST IS LIGHTSPEED

WHY ARE OLD KLINGONS DIFFERENT

WHY ARE THERE SQUIRRELS

WHY ARE THERE TINY SPIDERS IN MY HOUSE

WHY DO SPIDERS COME INSIDE

WHY ARE THERE HUGE SPIDERS IN MY HOUSE

WHY ARE THERE LOTS OF SPIDERS IN MY HOUSE

WHY ARE THERE SPIDERS IN MY ROOM

WHY ARE THERE SO MANY SPIDERS IN MY ROOM

WHY DO SPYDER BITES ITCH

WHY IS DYING SO SCARY

WHY IS THERE NO GPS IN LAPTOPS

WHY DO KNEES CLICK

WHY IS THERE CAFFEINE IN MY SHAMPOO

WHY HAVE DINOSAURS NO FUR

WHY DO IGUANAS DIE

WHY AREN'T ECONOMISTS RICH

WHY DO AMERICANS CALL IT SOCCER

WHY ARE MY EARS RINGING

WHY IS 42 THE ANSWER TO EVERYTHING

WHY CAN'T NOBODY ELSE LIFT THORS HAMMER

WHY IS MARVIN ALWAYS SO SAD

WHY ARE THERE ANTS IN MY LAPTOP

WHY IS EARTH TILTED

WHY IS SPACE BLACK

WHY IS OUTER SPACE SO COLD

WHY ARE THERE PYRAMIDS ON THE MOON

WHY IS NASA SHUTTING DOWN

WHY ARE THERE GHOSTS

WHY IS THERE AN OWL IN MY BACKYARD

WHY IS THERE AN OWL OUTSIDE MY WINDOW

WHY IS THERE AN OWL ON THE DOLLAR BILL

WHY DO OWLS ATTACK PEOPLE

WHY ARE FPGA's EVERYWHERE

WHY ARE THERE HELICOPTERS CIRCLING MY HOUSE

WHY ARE THERE GODS

WHY ARE THERE TWO SPOCKS

WHY ARE MY BOOBS ITCHY

WHY ARE CIGARETTES LEGAL

WHY ARE THERE DUCKS IN MY POOL

WHY IS JESUS WHITE

WHY IS THERE LIQUID IN MY EAR

WHY DO Q TIPS FEEL GOOD

WHY DO PEOPLE DIE

WHY AREN'T THERE GUNS IN

WHY ARE THERE DOGS AFRAID OF FIRE

WHY IS THERE NO KING IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

# QUESTIONS

CAN BE ASKED BY ANYONE ANYTIME

WHY AREN'T MY  
ARMS GROWING



WHY ARE THERE  
GHOSTS



WHY IS THERE AN OWL IN MY BACKYARD

WHY IS THERE AN OWL OUTSIDE MY WINDOW

WHY IS THERE AN OWL ON THE DOLLAR BILL

WHY DO OWLS ATTACK PEOPLE

WHY ARE FPGA's EVERYWHERE

WHY ARE THERE HELICOPTERS CIRCLING MY HOUSE

WHY ARE THERE GODS

WHY ARE THERE TWO SPOCKS

WHY ARE MY BOOBS ITCHY

WHY ARE CIGARETTES LEGAL

WHY ARE THERE DUCKS IN MY POOL

WHY IS JESUS WHITE

WHY IS THERE LIQUID IN MY EAR

WHY DO Q TIPS FEEL GOOD

WHY DO PEOPLE DIE

WHY AREN'T  
THERE GUNS IN

WHAT IS <https://xkcd.com/1256/>

WHY DO THEY SAY T-MINUS

WHY ARE THERE OBELISKS

WHY ARE WRESTLERS ALWAYS WET

WHY ARE THERE NO KINGS IN E

WHY ARE THERE NO KINGS IN E

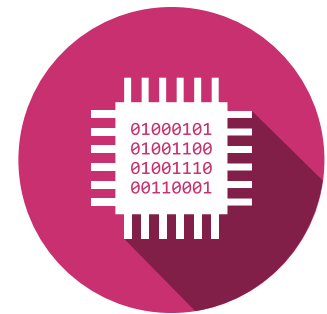
WHY ARE THERE NO KINGS IN E



**Hes·so**  **VALAIS  
WALLIS**



**Haute Ecole d'Ingénierie**  
**Hochschule für Ingenieurwissenschaften**



Silvan Zahno [silvan.zahno@hevs.ch](mailto:silvan.zahno@hevs.ch)  
Christophe Bianchi [christophe.bianchi@hevs.ch](mailto:christophe.bianchi@hevs.ch)  
François Corthay [francois.corthay@hevs.ch](mailto:francois.corthay@hevs.ch)