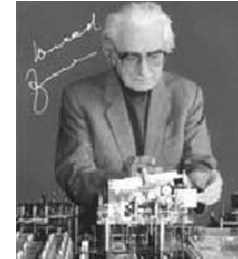


Modul 1: Grundkonzepte der IT



Konrad Zuse in GMD Bericht Nr 63 (1972): Der Plankalkül, S. 5 Bild aus [W1]

„... zur untersten Stufe gehört das Rechnen mit Zahlen; hier ist der Rechnungsgang bereits so schematisch und klar, dass mechanische Lösungen bereits in großem Umfang angewendet werden. Rechnen heißt also noch einmal kurz: aus gegebenen Angaben nach einer Vorschrift neue Angaben bilden.“

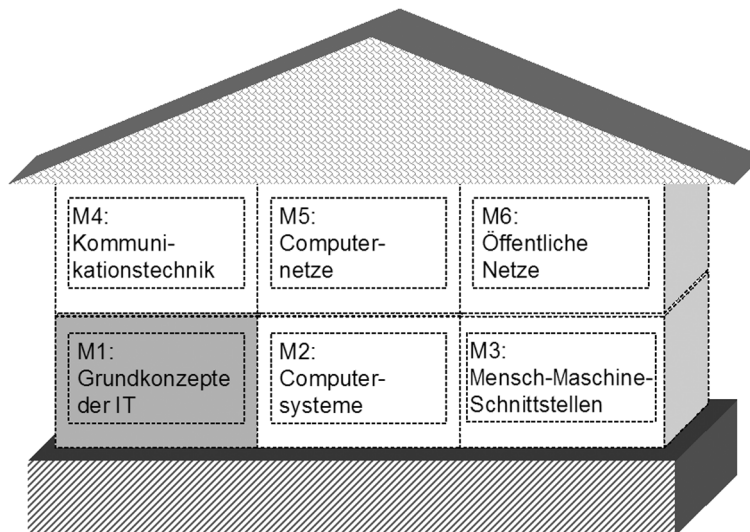
Lernziele

Für ein Verständnis sowohl der *Informationstechnik* als auch der *Informatik* ist zunächst ein Basiswissen über Begriffe rund um **Information** notwendig. Ein Einblick in Informationstheorie, Zahlensysteme und binäre Arithmetik stellt die Basis für das Verständnis der **Digitaltechnik** dar. Ausgehend von Prinzipien der **Computer-Organisation**, werden schließlich Schaltnetze und Schaltwerke behandelt. Damit wird ein Fundament für das Verständnis des darauf folgenden Moduls 2 „Computersysteme“ gelegt.

Informationsbegriff

Informationstheorie

Schaltnetze und Schaltwerke



1 Informationsbegriff 32

1.1 Signale 32

- 1.1.1 Grundlagen: Von Analog zu Digital 32
- 1.1.2 Abtastung und Quantisierung 36
- 1.1.3 Abtasttheorem 37
- 1.1.4 Aliasing 37
- 1.1.5 Quantisierungsrauschen 38

1.2 Daten 39

- 1.2.1 Datentyp 39
- 1.2.2 Elementare Datentypen 40

- 1.2.3 Strukturierte Datentypen 41
- 1.3 Nachricht 42
- 1.4 Information 43
 - 1.4.1 Informationsträger 44
 - 1.4.2 Informationstransport 44
 - 1.4.3 Digitale Information 44
 - 1.4.4 Informationsdarstellung 45
- 1.5 Wissen 46
- 1.6 Codierung 47
 - 1.6.1 Numerische Codes 48
 - 1.6.2 Alphanumerische Codes 49
 - 1.6.3 Strichcodes 51
 - 1.6.4 Flächencodes 51
- 2 Informationstheorie 52
 - 2.1 Informationsgehalt 52
 - 2.1.1 Informationsgehalt diskreter Zeichen mit gleicher Auftretenswahrscheinlichkeit 53
 - 2.1.2 Informationsgehalt diskreter Zeichen mit unterschiedlicher Auftretenswahrscheinlichkeit 54
 - 2.2 Entropie 56
 - 2.3 Redundanz 56
 - 2.4 Informationsfluss 57
 - 2.5 Kanalkapazität 58
 - 2.6 Informationsquader 60
- 3 Zahlensysteme und binäre Arithmetik 61
 - 3.1 Zahlensysteme: Grundlage für das Rechnen 61
 - 3.2 Boole'sche Algebra 64
- 4 Computer-Organisation 67
 - 4.1 Definitionen 67
 - 4.2 Schichtenmodell von Tanenbaum 68
- 5 Schaltnetze und Schaltwerke 69
 - 5.1 Schaltkreise 69
 - 5.1.1 Moore-Gesetz 69
 - 5.1.2 Halbleitertechnologie 70
 - 5.1.3 Die Basistechnologie: Transistor 71
 - 5.2 Schalter 72
 - 5.3 Schaltelemente 72
 - 5.3.1 Gatter 72
 - 5.3.2 Gattertechnologien 73
 - 5.4 Kombinatorische Logik: Schaltnetze 74
 - 5.5 Sequentielle Logik: Schaltwerke 75
 - 5.6 Rechenwerke 76
 - 5.6.1 Register 77
 - 5.6.2 Halbaddierer 77
 - 5.6.3 Volladdierer 78
 - 5.6.4 Akkumulator 78
- 6 Modulkurzzusammenfassung 79
- 7 Modulanhang 80
 - 7.1 Literatur 80
 - 7.1.1 Bücher 80
 - 7.1.2 Artikel 81
 - 7.1.3 Books in English 82
 - 7.1.4 Articles in English 82
 - 7.1.5 Journals 83
 - 7.2 Internet-Links 83
 - 7.3 Prüfungsfragen 83
 - 7.4 Lösungen 84
 - 7.5 Hands-On: Übungen 85
 - 7.6 Diskussionsfragen 85
 - 7.7 Timeline: Grundkonzepte der IT 85
 - 7.8 Glossar 86

Information (aus lat. „informatio“ = Vorstellung, Erläuterung, Belehrung) gehört zu den Schlüsselbegriffen in unserer Informationsgesellschaft. Unsere gesamte Lebens- und Arbeitswelt wird geprägt durch die rasant zunehmende Verwendung der **Informationstechnik (IT)**.

Es vollzieht sich auch ein fortlaufender Prozess einer *Annäherung* zwischen den klassischen Bereichen **Telekommunikation, Informationstechnik, Informatik** und (Multi-) **Medien**.

Aber was liegt allen diesen Bereichen zugrunde?

Die Grundlage dieser Entwicklung stellt die unvorhersehbar erfolgreiche Entwicklung der **Digitaltechnik** und der damit verbundenen Entwicklung **digitaler Computer** dar. Die **Digitalisierbarkeit** kann als ein Paradigma („Kennzeichen“) des 21. Jahrhunderts gesehen werden (Bild 1.1).

Ein Computer (aus lat. computare = „rechnen“) ist praktisch nichts anderes als eine (**digitale**) **Rechenmaschine**, die mit dem Dualsystem („0“ und „1“) rechnet.

Die **theoretische Grundlage** dieses Rechnersystems bildet die Boole'sche Algebra bzw. die Aussagenlogik.

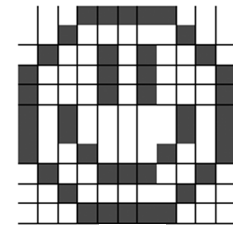
Die **technische Grundlage** bilden elektronische Schaltelemente (Transistoren, Integrierte Schaltkreise (ICs)).

Der Siegeszug der Digitaltechnik und dessen Monopol im Medienbereich liegt zweifelsohne begründet in *mindestens* drei Merkmalen:

- **Universalität** des digitalen Codes,
- **verlustfreie Reproduzierbarkeit** von Speicherinhalten und das
- **Von-Neumann-Prinzip**.

Die Entwicklung von Computern wird weltweit seit etwa 1930 betrieben. Zuerst wurden elektromechanische Bauelemente (Schalter, Relais) und später elektronische Bauelemente (Röhren) verwendet. In beiden Fällen nahmen diese Computer unglaubliche Ausmaße für eine sehr geringe Leistung ein. Erst durch die Entwicklung von Halbleiter-Bauelementen (Transistoren) seit etwa 1946 wurde es möglich, auf extrem kleinen Bauteilen sehr große Mengen von Schaltern bzw. Speichern unterzubringen. Diese „Integration“ war eine entscheidende Voraussetzung für den großflächigen Einsatz von Computern, wie wir sie aus unseren Tagen kennen. Mehr darüber in Modul 2.

Bevor wir uns mit der Funktion von Halbleiterbauteilen vertraut machen, ohne dabei auf physikalische Einzelheiten einzugehen, müssen wir Begriffe wie Signale, Daten, Nachricht, Information und Wissen klären.



```

0 0 0 1 1 1 1 1 0 0 0
0 0 1 0 0 0 0 0 1 0 0
0 1 0 0 1 0 1 0 0 1 0
1 0 0 0 1 0 1 0 0 0 1
1 0 0 0 1 0 1 0 0 0 1
1 0 1 0 0 0 0 0 1 0 1
1 0 1 0 0 0 0 0 1 0 1
1 0 0 1 0 0 0 1 0 0 1
0 1 0 0 1 1 1 0 0 1 0
0 0 1 0 0 0 0 0 1 0 0
0 0 0 1 1 1 1 1 0 0 0
    
```

Bild 1.1 Analog – diskret – digital: ein Beispiel für die „Digitalisierbarkeit“: Ein „analoges“ Bild (ganz oben), dessen Zerlegung in diskrete Elemente (mitte) und dessen digitale Darstellung; in Anlehnung an Rechenberg (2000), 28



1 Informationsbegriff

„Wissen“ wird eingehend in Band 2 von „Basiswissen Multimedia“ behandelt

Begriffe wie Signal, Daten, Nachricht, Information und Wissen sind für uns im alltäglichen Sprachgebrauch selbstverständlich.

Wir „setzen Signale“, wir „geben Daten weiter“, wir „erhalten Nachrichten“, wir „holen Information ein“, wir „eignen uns Wissen an“. In der Informationstechnik/Informatik benötigen wir allerdings klare Definitionen für diese Begriffe.

1.1 Signale

1.1.1 Grundlagen: Von Analog zu Digital

Signale als Träger von Informationen

Signale stellen nach DIN 44 300 die **physikalischen Träger** für Daten dar, die wiederum als Träger einer Nachricht und schließlich einer Information wirken. Prinzipiell werden zwei Typen von Signalen unterschieden:

- **analoge Signale**, bestehend aus **kontinuierlichen Funktionen**, und
- **digitale Signale**, bestehend aus **diskreten (einzelnen) Zeichen**.

Signale sind durch *akustische, optische* oder andere *physikalisch-technische* Mittel („Signalmittel“) gegebene Funktionen oder Zeichen zur Übermittlung von Daten.

Signale enthalten informationsabhängige und informationsunabhängige Merkmale.

Parameter = kennzeichnende Größe in Prozessen

Die informationsabhängigen Merkmale eines Signals nennen wir **Signalparameter**.

Alle Signale sind zeitabhängig. Die **Zeit** t ist eine wesentliche Größe in der Informationstechnik. Das der Information eindeutig zugeordnete Signal wird **Nutzsignal** genannt (desired signal) – in Abgrenzung zu den **Störsignalen** (drop-in, interference signal).

Signale können nach deren äußerer Form klassifiziert werden. Da Signale sowohl hinsichtlich des **Signalparameters (Wert)** als auch hinsichtlich der **Zeitabhängigkeit (Zeit)** eingeteilt werden können, ergeben sich insgesamt *vier* Möglichkeiten:

kontinuierlich = stetig durchlaufend;
diskret = in einzelne Punkte zerfallend, abzählbar

- wert- und zeitkontinuierlich („reines“ analoges Signal),
- wertdiskret und zeitkontinuierlich (quantisiertes Signal),
- wertkontinuierlich und zeitdiskret (abgetastetes Signal),
- wert- und zeitdiskret („reines“ digitales Signal).

Ein analoges Signal ist ein Signal, das einen **kontinuierlichen Vorgang kontinuierlich** abbildet. Ein digitales Signal ist ein Signal, das nur aus (codierten) **diskreten Zeichen** besteht.

Das Gebiet, das die Vorgänge bei der Verarbeitung und Übermittlung von Signalen beschreibt, wird **Systemtheorie** (signals and systems) genannt. Ein „System“ besteht dabei aus einer Menge von „Elementen“, die durch *Kopplungen* miteinander verbunden sind. Durch Abgrenzung können Ein- und Ausgangssignale von bzw. zur Systemumgebung unterschieden werden.

An dieser Stelle wird darauf hingewiesen, dass in der Informationstechnik hauptsächlich nicht determinierte (zufällige oder besser: nicht voraussagbare) Signale eine Rolle spielen. Die Grundlage zur Behandlung dieser Signale bildet die Wahrscheinlichkeitstheorie.

Wird ein kontinuierlicher Vorgang über einen kontinuierlichen Zeitraum abgebildet, sprechen wir von einem „reinen“ analogen Signal (Bild 1.2 A).

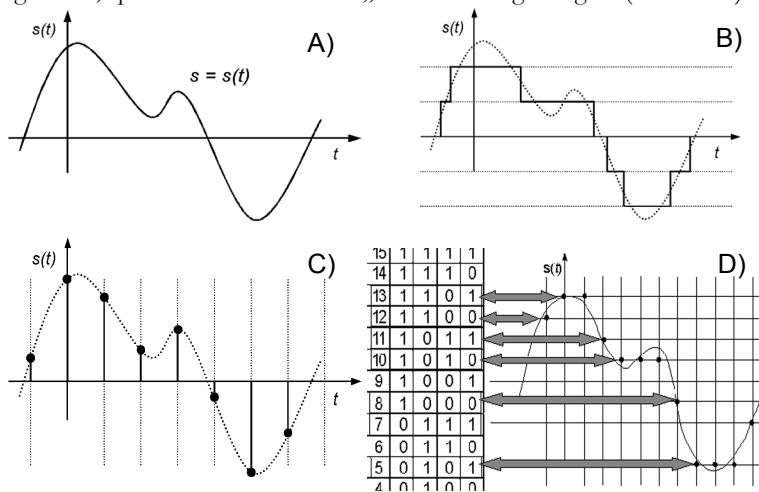


Bild 1.2 Vom Analogsignal zum Digitalsignal:
A) Ein „reines“ analoges Signal hat einen kontinuierlichen Verlauf s über die Zeit t ;
B) Ein amplitudenquantisiertes Signal ist zwar noch zeitkontinuierlich, aber wertdiskret;
C) Durch Abtastung zu bestimmten Zeitpunkten erhalten wir ein „Abtastsignal“;
D) Durch die Codierung der Abtastwerte in binäre Datenwörter entsteht ein „reines“ Digitalsignal

Der Begriff „analog“ kommt aus dem altgriechischen analogos = „verhältnismäßig, übereinstimmend“. Signal kommt aus dem lateinischen signalis = „dazu bestimmt, ein Zeichen zu geben“. Da analoge Signale wichtig in Multimedia-Systemen sind (Audio und Video), müssen wir uns noch etwas näher damit beschäftigen.

Allgemein werden analoge Signale mathematisch dargestellt durch:

$$s = s(t)$$

s (beliebiger) Signalparameter (kann z.B. eine Spannung $u(t)$ sein)
 t Zeit

IT 1 Grundkonzepte

Im Signal $s = s(t)$ ist die Zeit t die **unabhängige** Veränderliche und die Amplitude s die **abhängige** Veränderliche.

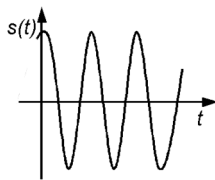
Das s steht für beliebiges Signal, $s(t)$ (sprich: „s von t“) und deutet auf die Abhängigkeit von der Zeit t hin.

In der Informationstechnik kann s beispielsweise eine elektrische Spannung $u(t)$, ein elektrischer Strom $i(t)$ oder ein mechanischer Schalldruck $p(t)$ sein.

s_{max} = Amplitude
 $s_{(t_1)}$ = Augenblickswert

Der Ausdruck s_{max} wird **Amplitude** genannt, der Ausdruck $s(t = t_1)$ oder kurz $s(t_1)$ **Augenblickswert** (sprich: s an der Stelle t_1).

Analoge Signale können weiter in unterschiedliche **Signalklassen** eingeteilt werden. Eine übergeordnete Unterscheidung erfolgt zwischen



- determinierbaren (vorhersehbaren) Signalen und
- nicht determinierbaren (nicht vorhersehbaren, „zufälligen“) Signalen.

Ein Signal ist dann **determinierbar**, wenn der Werteverlauf für jeden Zeitpunkt mathematisch **vorausberechenbar** ist.

Bild 1.3 Ein periodisches Signal in Form einer harmonischen Schwingung

Die determinierbaren Signale werden weiter eingeteilt in

- periodische Signale und
- nicht periodische Signale.

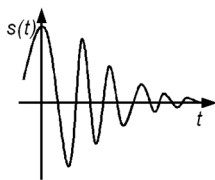


Bild 1.4 Ein nicht periodisches Signal in Form einer transienten Ausschwingfunktion (Übergangsvorgang)

Periodische Signale sind beispielsweise harmonische Schwingungen. Ein bestes Beispiel dafür ist die bekannte Sinusschwingung, definiert durch:

$$s(t) = s_{max} \cdot \sin((\omega t) + \varphi)$$

- ω Kreisfrequenz ($\omega = 2\pi f$)
- φ Phasenverschiebung (Abweichung vom Ursprung)

Nach JEAN BAPTISTE JOSEPH FOURIER (1768–1830) können periodische Signale als Summe harmonischer Schwingungen, d.h. als Summe von sinus- bzw. cosinusförmigen Teilschwingungen (in einer mathematischen Reihe, der sog. Fourierreihe), zusammengesetzt werden (Fourierzerlegung).

Nicht periodische Signale können sein:

EKG
Elektrokardiogramm
(Herzstromkurve)

- quasiperiodisch (z.B. EKG) oder
- transient (z.B. ausschwingendes Pendel, siehe Bild 1.4).

Die nicht determinierbaren Signale sind für uns (vom informationstheoretischen Standpunkt aus) die wichtigsten. Durch ihr „zufälliges Verhalten“ werden sie als **stochastisch** bezeichnet.



Bild 1.5 Stationäres Breitbandrauschen

Das zufälligste Signal, das wir kennen, nennen wir **Rauschen** (noise).

Experimentell kann Rauschen (Bild 1.5) am Fernsehgerät bei abgezogenem Antennenstecker hörbar (Rauschen) und sichtbar (Grieseln) gemacht werden (Bild 1.6).

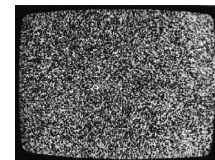


Bild 1.6 Das statistisch zufälligste Signal können wir auf einem TV-Gerät sichtbar machen

Werden nicht unendlich viele Funktionswerte zugelassen, sondern wird der Signalparameter in eine endliche Anzahl **diskreter Intervalle** unterteilt, erhalten wir aus einem analogen Signal ein **amplitudenquantisiertes** Signal (siehe Bild 1.2 B). Ein gutes Beispiel dafür ist die Aussteuerungsanzeige bei Magnetbandgeräten zur Tonaufzeichnung (Kassettenrecorder).

Wenn ein analoges Signal zu bestimmten, diskreten Zeitpunkten **abgetastet** wird, erhalten wir ein Abtastsignal (siehe Bild 1.2 C).

Ein wertkontinuierliches, zeitdiskretes **Abtastsignal** wird **Pulsamplitudenmodulation (PAM)** genannt. Dieses ist sehr empfindlich gegen Störungen und tritt praktisch nur in der „Zwischenstufe“ vor der eigentlichen Codierung auf.

Mathematisch erfolgt eine Zeitdiskretisierung, indem das analoge Signal

$$s = s(t)$$

durch Abtastung an diskreten Zeitpunkten $t_1 \dots t_n$ Amplitudenwerte annimmt, die kontinuierlich innerhalb des Signalwertebereichs sein dürfen.

Technisch kann die Erzeugung eines PAM-Signals über einen von einem Rechteckgenerator (RG) gesteuerten **Analogschalter** erfolgen (Bild 1.7): Der Schalter öffnet und schließt entsprechend der Abtastfrequenz f_a

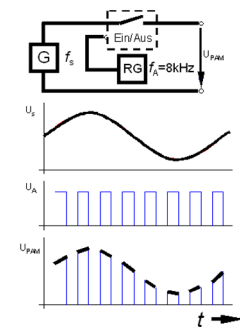


Bild 1.7 Eine Möglichkeit zur technischen Realisierung einer PAM (siehe Text)

Eigenschaften der Pulsamplitudenmodulation:

- NF-Signalquellen können gemultiplext werden,
- hohe Bandbreite für das Signal notwendig,
- stör anfällig (Information liegt in der Amplitude).

Die Pulsamplitudenmodulation (PAM) dient in der Praxis als Vorstufe für ein anderes Pulsmodulationsverfahren.

IT 1 Grundkonzepte

Bit 3 MSB				Bit 0 LSB
0	1	0	1	
2^3	2^2	2^1	2^0	

Bild 1.8 Ein 4-bit-Datenwort: die höchste Stelle ist das most und die niedrigste das least significant bit

Die Anzahl der darstellbaren Zeichen ist 2^w

Erst wenn ein analoges Signal *abgetastet*, *amplitudenquantisiert* und dann noch die Abtastwerte *binär verschlüsselt* (codiert) werden, erhalten wir ein „reines“ digitales Signal (siehe Bild 1.2 D).

Die binäre Digitaltechnik basiert auf **zweiwertiger Logik**: Werte werden durch Binärziffern „0“ und „1“ dargestellt.

In unserem Beispiel sind die Abtastwerte durch Datenwörter mit einer Wortlänge von 4 bit dargestellt. Bei Beschränkung auf ganzzahlig positive Zahlen ist der größte realisierbare Zahlenwert für eine Binärziffer (Dualzahl) der Wortlänge w bit durch folgenden Ausdruck gegeben:

$$Z_{max} = 2^w - 1$$

In unserem Beispiel (4-bit-Datenwort) ist die Anzahl der darstellbaren Zeichen $2^4 = 16$ und damit der größte realisierbare Zahlenwert $16 - 1 = 15$.

Unter **Digitalisierbarkeit** wird die Möglichkeit zur Überführung in wert- und zeitdiskrete Signale (Digitalsignale) verstanden. Das ist das Paradigma des 21. Jahrhunderts (digital age): Jedes akustische und optische Element kann in eine binäre (zweiwertige) „0/1-Sprache“ übersetzt (codiert, siehe Kapitel 1.6) und damit maschinell verarbeitbar gemacht werden.

1.1.2 Abtastung und Quantisierung

Das Umwandeln in zeitdiskrete Werte erfolgt durch Abtastung (sampling) des Eingangssignals $s(t)$ im Zeitbereich von $n \cdot T_s$, wobei T_s den Abstand zwischen zwei Abtastwerten angibt. Beispielsweise bei einer Abtastfrequenz von 16 kHz bedeutet dies einen jeweiligen Abstand der nächsten Abtastpunkte (Samples) von 62,5 μ s.

Abtastung: Es werden dem Signal in gleichmäßigen Zeitabständen „Signalproben“ (Abtastwerte) entnommen.

Das abgetastete Signal existiert nun nur noch an den Abtastzeitpunkten. Wichtig bei dem Abtastverfahren ist, dass das Abtasttheorem von SHANNON eingehalten wird.

Quantisierung: Es erfolgt eine Einteilung des Amplitudenbereiches in N Intervalle, und jeder Abtastwert wird durch die Nummer desjenigen Intervalls gekennzeichnet, in das er fällt.

1.1.3 Abtasttheorem

Das Abtasttheorem von SHANNON besagt, dass die Abtastfrequenz **mehr als doppelt so groß** sein muss als die maximal auftretende Frequenz des abzutastenden Signals.

Es muss also gelten:

$$f_a > 2 \cdot f_{max}$$

f_a Abtastfrequenz
 f_{max} höchste im Signal enthaltene Frequenz

Die Abtastzeitdauer T_s ergibt sich somit zu:

$$T_s = \frac{1}{2 \cdot f_{max}}$$

Falls das Abtasttheorem nicht eingehalten wird, kommt es im Signal zu ungewollten **Verzerrungen** während der Abtastung, somit auch zu einer Signalverfälschung (Verfremdung).

Diese ungewollten Verzerrungen stellen „falsche (alias) Frequenzen“ dar und sind die Ursache für Bildstörungen (edges) oder Tonstörungen (buzz).

**false (alias)
frequency**

Beispiel Musik-CD: Da die maximale Hörfrequenz (Modul 3) des Menschen bei etwa 20 kHz liegt, muss – um optimale Musikqualität zu gewährleisten – die verwendete Abtastrate doppelt so groß sein: 44 100 Hz bei 16 Bit (= 65 536 mögliche Abtastwerte).

1.1.4 Aliasing

Wird nun ein Signal unterabtastet (siehe Bild 1.9), so tritt der so genannte **Aliasing-Effekt** auf. Dies kann dazu führen, dass es zu Überlappungen des sich periodisch wiederholenden Spektrums kommt, nämlich dann, wenn:

$$f_s < 2 \cdot f_{max}$$

**Unterabtastung →
Aliasing
(Verfremdung)**

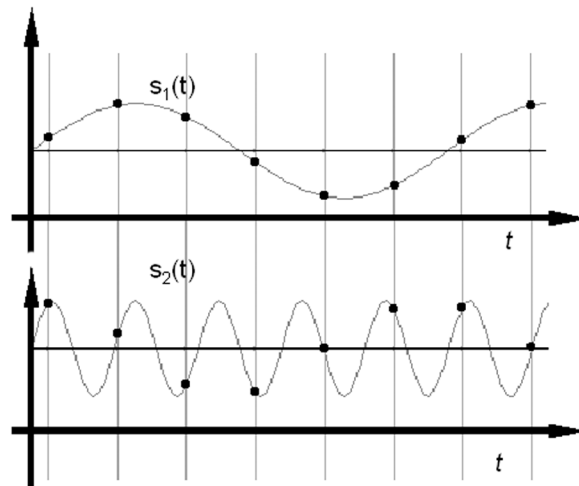
Die zeitdiskreten Werte von Abtastsignalen können nur in Datenwörter endlicher Wortlänge abgebildet werden. Dies bedeutet natürlich, dass jeder Amplitudenwert nur mit einer *endlichen Genauigkeit* aufgelöst werden kann, die unmittelbar mit der Datenwortlänge verbunden ist. Der zulässige Amplitudenbereich wird in eine endliche Anzahl diskreter Amplitudenintervalle aufgeteilt (Amplitudenquantisierung).

Tiefpass = lässt tiefe Frequenzen passieren, schwächt hohe Frequenzen ab

In der Praxis werden Signale mit Frequenzanteilen oberhalb der halben Abtastfrequenz deshalb durch einen **Tiefpass** vorgefiltert.

Ein solches Filter hat daher den Namen **Antialiasing-Filter**, manchmal auch **Presampling-Filter**.

Bild 1.9 Bei zu kleiner Abtastrate ergibt die gedachte Verbindungslinie durch die Abtastpunkte ein falsches Abbild des Signals (unten)



1.1.5 Quantisierungsrauschen

Da das wertkontinuierliche Analogsignal bei der Digitalisierung auf Werte einer endlichen Wertemenge abgebildet wird, kommt es zwangsläufig zu **Rundungsfehlern**. Da die Fehlerbeträge im Zeitverlauf statistisch gleich verteilt auftreten, können sie als Rauschen interpretiert werden (Quantisierungsrauschen).

Nochmal: Rauschen

Mit Rauschen (siehe Bild 1.5 und Bild 1.6) wurde ursprünglich der durch statistische Schwankungen des Luftdrucks hervorgerufene Schalleindruck bezeichnet. Allgemein wird mit Rauschen jede statistische Störung von Signalen bezeichnet.

Insbesondere bei Signalen, die in einem geradzahigen Verhältnis zur Abtastfrequenz stehen, ergibt sich ein unangenehmes „tonales“ Rauschen. Die Fehlerrate ist nicht mehr zufällig, sondern folgt dem Phasenverhalten des Signals.

Die für die Abtastung gewählte Frequenz (Bit-Anzahl der Digitalisierung) bestimmt die Größe des Quantisierungsrauschens. Je höher die Bit-Anzahl, desto höher der Signal-Rausch-Abstand und desto niedriger das Quantisierungsrauschen.

1.2 Daten

Daten sind Fakten. Nach DIN 44 300 sind Daten „*Zeichen oder kontinuierliche Funktionen, die zum Zweck der Weitergabe Information auf Grund bekannter oder unterstellter Abmachungen darstellen*“. Eine Gliederung kann zunächst folgendermaßen erfolgen

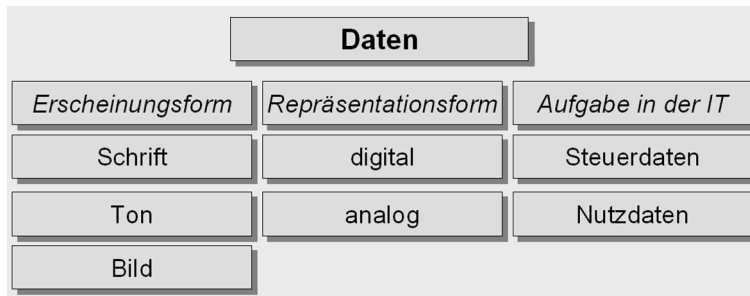


Bild 1.10 Daten können hinsichtlich der Erscheinungsform, der Art der Darstellung (Repräsentationsform) und der Aufgabe in der IT eingeteilt werden

In der IT/Informatik sind Daten codierte Informationen, die in Computersystemen gespeichert sind. Die Information kann dabei durch *Zeichenfolgen (digitale Daten)* oder *kontinuierliche mathematische Funktionen (analoge Daten)* dargestellt und maschinell verarbeitet werden.

Daten sind codierte Informationen

1.2.1 Datentyp

Hier werden zunächst nur die wichtigsten Datentypen vorgestellt, Datenstrukturen und Algorithmen der Informationsverarbeitung werden in Band 2, Modul 1 behandelt.

Datentypen stellen in der Informatik (siehe Band 2) „Zeichensorten“ dar, mit denen bestimmte Algorithmen arbeiten. Daten bzw. Datentypen haben nur eine maschineninterne Bedeutung (technische Basiszeichen).

Die Informatik beschäftigt sich mit der Formalisierbarkeit von Zeichenbedeutungen und der Möglichkeit, Information als Daten zu speichern und zu verarbeiten.

Ein **Datentyp** ist die Festlegung der Interpretation einer gespeicherten (physikalischen) Bitfolge. Datentypen können einfacher Art sein (ganze oder reelle Zahlen, logische Werte, Zeichen) oder zusammengesetzt (also komplex) bzw. strukturiert sein (Felder, Zeichenfolgen, Strukturen usw.).

Der Datentyp legt die Art der gespeicherten Information und deren Auswertungsmöglichkeit fest.

Die Festlegung des Datentyps für eine Variable wird Definition (auch Deklaration oder Vereinbarung) genannt und besitzt folgenden Aufbau:

<definitionszeichnung> <identifikation> <typmerkmal> <definitionsabschluss>

Beispiel (Vorgriff auf Band 2): In der Programmiersprache Pascal gibt es eine Definition: `var Zahl integer`. „Zahl“ definiert dabei eine ganzzahlige Größe.

Die Definitionskennzeichnung kann in manchen Programmiersprachen auch entfallen. Außerdem kann das Typmerkmal der Identifikation bzw. dem Variablennamen voranstehen.

Die Möglichkeit der digitalen Wertedarstellung der Informationen eines Anwendungsbereiches bestimmt wesentlich auch die Anwendbarkeit eines Computers.

Probleme bzw. Lösungsformen für Wertedarstellungen sind beispielsweise: die Anwendung zweidimensionaler Zeichendarstellung für chinesische, japanische oder so ähnliche Sprachen, die Digitalisierung von multimedialen Informationen, wie Bilder, Audio- und Geruchsinformationen, die Codierung und Decodierung von Symbolen in Fachsprachen, wie Noten, Icons usw. oder z.B. die Transformation der von Sensoren (z.B. Radsensor in Anti-Blockier-Systemen (ABS)) erhaltenen Signale.

1.2.2 Elementare Datentypen

Daten werden nach drei Gesichtspunkten eingeteilt:

- Struktur,
- Bedeutung und
- Operationsmöglichkeiten.

Zu den elementaren Datentypen werden folgende drei gezählt:

- **Bit, Byte:** Ein Bit ist überhaupt die *elementarste Einheit* der Informationsverarbeitung und nimmt die Werte „0“ oder „1“ an. Ein Byte ist eine geordnete Menge von 8 Bit (siehe dazu Kapitel 1.4).
- **Zahlen** (z.B. Integer = ganze Zahlen; Real = Gleitpunktzahlen; Boolean = boolesche Daten): diese können durch eine Menge von Bytes dargestellt werden.
- **Zeichen, String, Wort:** Neben den Zahlen gibt es viele alphabetische Zeichen und Sonderzeichen (`char` = Zeichen) und ihre Gruppierung in Form von Strings. Ein *String* ist eine Aneinanderreihung von Zeichen oder Sonderzeichen. Ein *Wort* ist eine geordnete Menge alphabetischer Zeichen (ohne Sonderzeichen).

1.2.3 Strukturierte Datentypen

In der Praxis kommt man mit den einfachen Datentypen nicht aus. Daher können in vielen Programmiersprachen beliebig kompliziert *zusammengesetzte* Datenstrukturen definiert und durch eigene Variablen bezeichnet werden:

- **Mengen, Arrays, Listen, Matrizen:** Die Anordnung von Zahlen und Strings in geordneten oder ungeordneten *Mengen* (ein-, zwei- oder mehrdimensional) ermöglicht hohe Flexibilität zur Manipulation von Daten. Ein *Array* ist eine geordnete Menge gleichartiger Datentypen. Eine *Liste* ist eine geordnete Menge verschiedener elementarer oder zusammengesetzter Datentypen. Eine *Matrix* ist eine mehrdimensional geordnete Menge von gleichartigen Datentypen.
- **Tabellen, Relationen:** Im Bereich der Datenbanken (siehe Band 2, Modul 3) haben relationale Datenbanken eine lange Tradition. Daten werden in *Tabellen* gespeichert und können durch *Relationen* verknüpft werden. Die Relationen selbst werden in Form von Tabellen abgelegt. Eine Tabelle besteht dabei aus n Zeilen und m Spalten, ist also eine zweidimensionale geordnete Struktur. Oft enthält eine Spalte nur Elemente eines vorgegebenen Datentyps. Die Anzahl der Spalten wird als konstant betrachtet und ist in der Regel bei Anlegen der Tabelle festgelegt. Die einzelnen Datensätze werden in neue Zeilen eingefügt bzw. gelöscht. Die Zeilenzahl einer Tabelle ist damit flexibel.
- **Bäume (Trees), Graphen:** Bäume und Graphen sind Datenstrukturen, die aus Knoten und Kanten bestehen: Eine *Kante* verbindet stets zwei *Knoten* und kann somit als ein Paar von Knoten verstanden werden.
- **Files:** ein Konzept, das in Zusammenhang mit Betriebssystemen (siehe Band 2, Modul 2) entwickelt wurde: Ein File ist eine geordnete Menge von Bytes. Es kann selbst z.B. Zahlen, Zeichen usw. enthalten.
- **Programme:** Ausführbare Programme sind spezielle geordnete Mengen von Daten (physikalisch von Bytes), die der Prozessor eines Computers ausführen kann. Es wird unterschieden zwischen Programmcode (auch „Quellcode“ genannt) und Maschinencode (auch „Bytecode“ genannt). Der *Programmcode* besteht aus Files und ist in einer künstlichen Sprache (Programmiersprache, Band 2, Modul 3) geschrieben. Der *Maschinencode* hingegen besteht auf der untersten Ebene nur aus Binärcode („0“ und „1“) und kann vom Prozessor (oder einer virtuellen Maschine) direkt ausgeführt werden.
- **Objekte, Klassen, Methoden:** Ab ca. 1990 wurden objektorientierte Datenstrukturen in viele Programmiersprachen eingeführt. *Objekte* sind dabei prinzipiell wieder verwendbare „Bausteine“. *Klassen* sind flexible Ansammlungen verschiedenster elementarer und zusammengesetzter Datentypen. Klassen können weitere Klassen und Programme – die so genannten *Methoden* – enthalten.

Relation = allg. jede Beziehung zwischen Datensätzen

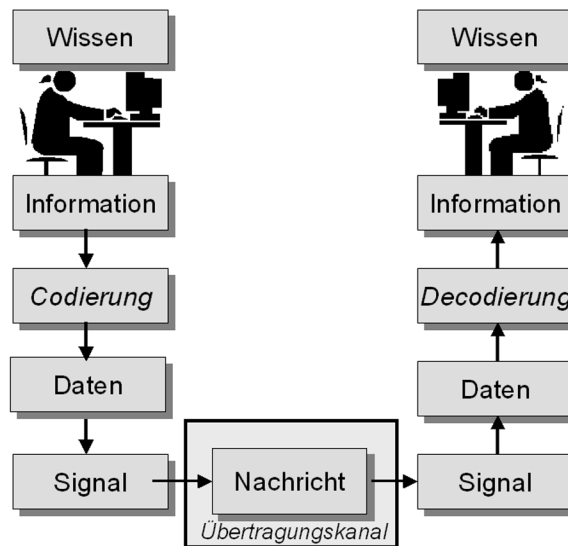
Objekt = ausführbare Programmeinheit, die Zustand besitzt, über Algorithmen (Methoden) verfügt und Nachrichten versenden und empfangen kann

Beispiel (Vorgriff auf Band 2): In der Programmiersprache Java werden vier Gruppen einfacher Datentypen unterschieden: logischer Typ (boolean), Ganzzahl-Typen (int, byte, short, long), Gleitpunkt-Typen (float, double) und der Zeichentyp (char). Zeichenketten oder komplexere Formen sind in Java durch Objekte auszudrücken und umzusetzen.

1.3 Nachricht

In Zusammenhang mit „klassischen“ Nachrichtenübertragungssystemen (z.B. Telefon) wird von Nachrichten anstelle von Daten gesprochen. Eine Nachricht ist eine Folge von Zeichen, oft aus sinnesphysiologischen oder technischen Gründen in Wörter zusammengefasst.

Bild 1.11 Im Kontext einer Übertragung wird oft von Nachricht anstelle von Daten gesprochen; auch wird (insbesondere in dieser Darstellung) Information mit Nachricht gleichgestellt; dieses Bild soll uns also nur ein Gefühl für mögliche Unterschiede geben



Information dagegen (siehe Kapitel 1.4) enthält bereits eine *Bedeutung*. Daher wird in Übertragungssystemen „Nachricht“ als Oberbegriff verwendet. Nachrichten können dabei gesprochene oder geschriebene Texte, Bilder, Töne etc. sein, die mit den menschlichen Sinnesorganen wahrnehmbar sind und/oder mit technischen Hilfsmitteln übertragen und gespeichert werden.

Welche *Information* einer Nachricht zugeschrieben wird, ist vom *Empfänger* dieser Nachricht *abhängig* und daher **subjektiv**.

So kann zum Beispiel die Nachricht „mayday“ für einen Empfänger bedeutungslos sein (was ist das überhaupt?); für einen anderen kann diese Nachricht aber eine lebenswichtige Information enthalten („mayday“ ist im internationalen Funkverkehr ein Notsignal).

1.4 Information

Der Begriff „Information“ kristallisiert sich als ähnlich fundamentaler Begriff heraus, wie das die Begriffe Energie und Materie schon seit langem als Basis der Naturwissenschaft sind (Bild 1.12). Allerdings wird der Begriff in Wissenschaft und Technik sehr unterschiedlich verwendet, z.B.:

- In jeder Art von Struktur von Materie oder Energie steckt Information;
- alles, was als Folge von „0“ und „1“ darstellbar ist, ist Information;
- Information ist beseitigte Ungewissheit;
- Information ist eine räumlich-zeitliche Folge physikalischer Signale, die mit bestimmten Wahrscheinlichkeiten oder Häufigkeiten auftreten;
- Information ist eine Nachricht, die beim Empfänger interpretiert wird und ein bestimmtes Verhalten auslöst.

Für die Natur wurde als Information „jeder Unterschied, der einen Unterschied macht“ definiert (BATESON) oder spezieller: „Information ist der Veränderungsprozess, welcher zu einem Zuwachs an Wissen führt“ (SCHUCAN) oder noch spezieller: „Information ist nutzbare Antwort auf eine konkrete Fragestellung“ (ZEHNDER).

In der Nachrichten- und Informationstechnik wird die Informationstheorie von SHANNON & WEAVER zur Definition herangezogen: Informationen sind nur Aussagen bzw. Nachrichten, die etwas **Neues** bringen. Informationen müssen also einen Informationsgehalt (eine Bedeutung) besitzen (siehe Kapitel 2.2). Häufige, gleich ablaufende (monotone) Ereignisse bieten in diesem Sinne „nichts Neues“.

Überraschende, seltene oder neue Vorgänge sind nachrichtentechnisch informativ. In der Änderung eines Signals liegt die Information.

NORBERT WIENER, Begründer der Kybernetik (*Lehre von Kommunikations- und Regelungssystemen in lebenden Organismen, gesellschaftlichen Organisationen und auch technischen Systemen*), führt Information als dritte **Grundkomponente** ein.

Information ist eine raum- und/oder zeitvariante (veränderliche) Folge physikalischer Signale, bestehend aus dem physikalisch-materiellen **Träger** und dem **Informationsinhalt**.

Eine interessante Definition gibt schließlich KUHLEN (1999): „Information ist eine Teilmenge von Wissen (siehe Kapitel 1.5), die aktuell in Handlungssituationen benötigt wird und vor der Informationsverarbeitung nicht vorhanden ist. Information ist demnach von den Kontextfaktoren der Benutzer und der Nutzungssituation abhängig“. Der Kontext ist das „Hintergrundwissen“, das notwendig ist, um einen Sachverhalt zu verstehen.

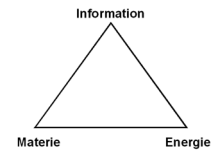


Bild 1.12 So könnte die „Dreieinigkeit“ der Naturwissenschaften aussehen

Die „Neuigkeit“, nicht die „Inhalts-schwere“ einer Nachricht wird definiert



Bild 1.13 Norbert Wiener (1894–1964) führte Information als dritte Grundkomponente ein – neben Materie und Energie! [W7]

Die Abhängigkeit vom Kontext wird in der Kommunikation augenscheinlich (siehe Basiswissen Multimedia, Band 1, Modul 1).

1.4.1 Informationsträger



Bild 1.14 Auf „Holz-Datenträgern“, so genannten Kerbhölzern, wurden bis ins 20. Jahrhundert Informationen gespeichert [W2]

RAM = Random Access Memory = beliebig beschreibbarer und löschrbarer Arbeitsspeicher, ROM = Read Only Memory = nur lesbarer Speicher, siehe Modul 2

Informationsträger sind Dinge, die zur Übertragung bzw. Speicherung von Information genutzt werden können, z.B. wurden seit der Urzeit Kerbhölzler verwendet (Bild 1.14).

Schallwellen, elektromagnetische Wellen (auch Licht ist elektromagnetische Energie), Schrift, Materiestruktur oder Stoffkonzentrationen sind nur einige Beispiele für Informationsträger. Ein Lehrsatz lautet: Information ohne Informationsträger gibt es nicht.

Einige Anforderungen an Informationsträger sind: Lesbarkeit, Alterungsbeständigkeit, dürfen nicht unabsichtlich überschreibbar, aber müssen absichtlich überschreibbar sein.

Beispiele für löschbare und überschreibbare Informationsträger: Schiefertafel, Musikkassette, Videoband, RAM-Speicher (siehe Modul 2).

Beispiele für nicht neu beschreibbare Informationsträger: in Stein gehauene Schrift, Schallplatte, CD, ROM-Speicher (siehe Modul 2).

1.4.2 Informationstransport

Information kann im luftleeren Raum (Vakuum) mit Lichtgeschwindigkeit c transportiert werden, auf einer elektrischen Leitung mit etwa $2/3$ von c .

Beispiel: Telefax. Ein Schriftstück am Ort A enthält eine Information. Diese wird mit Hilfe des Faxgerätes in elektrische Signale umgewandelt und über eine Telefonleitung über große Entfernungen transportiert. Am Ort B wird es durch das vorhandene Faxgerät wieder in ein (zweites) Schriftstück mit demselben Informationsinhalt zurückverwandelt.

Eine Informationsübertragung zwischen zwei Systemen ist immer sowohl mit Energie- als auch Entropieübertragung (siehe Kapitel 2.2) verbunden. Dabei kann mit wenig Energie viel Information übertragen werden (z.B. Telefonnetz; Schwachstromtechnik). Andererseits kann mit der Übertragung von viel Energie sehr wenig Informationstransport verknüpft sein (z.B. Überlandleitungen: Starkstromtechnik).

Information und Energie sind selten direkt proportional.

1.4.3 Digitale Information

Nach Einführung und Verbreitung von Computern wurde bald klar, dass Systeme, die nur binär (mit *zwei Zuständen*) arbeiten (0/1; Nein/Ja; Aus/Ein) zur Verarbeitung großer Datenmengen am besten geeignet sind. Die Basisaussage der Informationstheorie ist der Satz von der Digitalisierbarkeit:

$$c = 3 \cdot 10^8 \text{ m/s} \\ = 300\,000 \text{ km/s}$$

Digitale Information ist **alles**, was als Folge von „0“ und „1“ (binär) darstellbar ist. Digitalisierung ist damit die Überführung beliebiger Daten in eine solche binäre Darstellung.

```

0 0 0 1 1 1 1 1 0 0 0
0 0 1 0 0 0 0 0 1 0 0
0 1 0 0 1 0 1 0 0 1 0
1 0 0 0 1 0 1 0 0 0 1
1 0 0 0 1 0 1 0 0 0 1
1 0 1 0 0 0 0 0 1 0 1
1 0 1 0 0 0 0 0 1 0 1
1 0 0 1 0 0 0 0 1 0 0 1
0 1 0 0 1 1 1 0 0 1 0
0 0 1 0 0 0 0 0 1 0 0
0 0 0 1 1 1 1 1 0 0 0
    
```

In der Öffentlichkeit wurde erst mit der massenhaften Einführung der (digitalen) Musik-CD und der damit verbundenen Ablösung der (analogen) Schallplatte, die „Mächtigkeit“ der Digitaltechnik allgemein bekannt.

1.4.4 Informationsdarstellung

Ein **Bit (binary digit)** oder auch **Binärzeichen** ist die kleinstmögliche Informationseinheit des Computers. Ein Bit kann *nur zwei Werte* einnehmen: entweder den Wert „0“ (aus) oder „1“ (ein). Physikalisch wird ein Bit dadurch erzeugt, dass den **logischen Werten** „0“ und „1“ entsprechende elektrische Spannungswerte zugewiesen werden. Diese Codierung von „aus“ und „ein“ wird als **Binärcode** (siehe Kapitel 1.6) bezeichnet.

Bild 1.15 Digitale Information ist alles, was als Folge von 0 und 1 dargestellt werden kann (siehe auch Bild 1.1)

In der Praxis wird mit *Gruppen von Bit* operiert:

- **Nibbel:** Ein Nibbel (auch Halbbyte, Tetrade) besteht aus 4 Bit und wurde vor allem in der Anfangszeit der Computertechnik verwendet.
- **Byte:** Ein Byte ist eine *Folge von 8 Bit*, die gemeinsam verarbeitet werden. Ein Byte ist auch eine 8-Bit-Größe mit einer eindeutigen Speicheradresse. Es ist die kleinste adressierbare Speichereinheit („Speicherzelle“).
- **Wort:** Ein Wort ist eine 16-Bit-Größe, die aus zwei aufeinander folgenden Bytes besteht.
- **Doppelwort:** Ein Doppelwort ist eine 32-Bit-Größe, die aus vier aufeinander folgenden Bytes besteht.
- **Vierfachwort:** Ein Vierfachwort ist eine 64-Bit-Größe, die aus acht aufeinander folgenden Bytes besteht.



Bild 1.16 Einige praktische Beispiele, um ein Gefühl für die Größe „Bit“ zu erhalten

Bit und Byte sind also (digitale) Maßeinheiten in der Informationstechnik (Achtung: Unterschied in der Informationstheorie, siehe Kapitel 2.1). Gebräuchliche Abkürzungen sind:

- 1 KB (Kilobyte) = 1 024 Bit
- 1 MB (Megabyte) = 1 024 KB = 1 048 576 Bit
- 1 GB (Gigabyte) = 1 024 MB = 1 073 741 824 Bit
- 1 TB (Terrabyte) = 1 024 GB = 1 099 511 627 776 Bit

*Aus der Informationsmenge können wir zurückschließen auf die Zahl Z der zugrunde liegenden möglichen Ereignisse: 0 Bit → Z = 1 da $\lg_2 1 = 0$, da $2^0 = 1$; 1 Bit → Z = 2 da $\lg_2 2 = 1$, da $2^1 = 2$; 8 Bit = 1 Byte → Z = 256, da $\lg_2 256 = 8$, da $2^8 = 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2 = 256$; 1 Kilobyte = 1kB = 2hoch¹⁰ Byte = 210 Byte = 1024Byte = 8 * 1024 Bit = 8192 Bit = ca 1 000 Byte usw.*

1.5 Wissen

Wissen ist angewandte Information



Bild 1.17 Heinz Zemanek (geb. 1920), der Computerpionier, während seiner Ansprache zu seinem achtzigsten Geburtstag



Bild 1.18 Zusammen mit Warren McCulloch, Norbert Wiener, John von Neumann u.a. war Heinz von Foerster (geb. 1911 in Wien) der „Architekt“ der Kybernetik [W3]

Wissen (engl. knowledge; lat. scientia; grch. episteme) ist das Ergebnis – das Ziel – einer (menschlichen) Informationsverarbeitung.

Denken und Wissen sind untrennbar mit dem menschlichen Geist verbunden: **Wissen** ist (im menschlichen Gedächtnis) organisierte, verfügbare und anwendbare Information.

Nach HEINZ ZEMANEK gehört gespeicherte Information gewiss zum Wissen, aber Wissen wird diese *erst mit dem Kontext*, den der unvergleichliche Verbund unserer Sinnesorgane – diesem dient z.B. die Multimedia-Technik – liefert, wenn die Information vom menschlichen Verstand verarbeitet wird.

Weder Telefonbuch noch Datenbank sind Wissen – aber mit Wissen holt man aus beiden wesentlich Gescheiteres heraus (Heinz Zemanek).

HEINZ VON FOERSTER (1998) meint, ganz im Sinne des Konstruktivismus (siehe Basiswissen Multimedia Band 2), dass Wissen *gar nicht vermittelbar* ist: Es lässt sich nicht als Gegenstand, eine Sache oder ein Ding begreifen, das man – wie Zucker oder Kaffee – von A nach B transportieren kann, um in einem Organismus eine bestimmte Wirkung zu erzeugen.

Wissen ist vorausschauend und **probabilistisch**: Der Wert eines Wissens ist hoch, wenn die Voraussagen, bezogen auf den Übertragungskanal, korrekt sind. Das steht im Gegensatz zur Information, die **possibilistisch** ist, da die unwahrscheinlichsten Zeichen den höchsten Informationsgehalt tragen.

JOHN GUNDRY, Director von Knowledge Ability Ltd., erläutert den Unterschied zwischen Daten, Information und Wissen folgendermaßen:

Daten sind 0-dimensional: Fakt: Intercity IC 656 Graz–Wien, Abfahrt: 06:12, Gleis 5, täglich, Ankunft Wien: 09:01

Informationen sind 1-dimensional: Damit ich um 09:00 Uhr in Wien sein kann, muss ich den Zug um 06:12 Uhr erwischen.

Wissen ist 2-dimensional: Wenn ich den Zug um 06:12 verpasse, muss ich mir eine Alternative ausdenken, damit ich doch noch um 09:00 Uhr in Wien bin. Der nächste Zug geht erst um 07:26, daher wird es das Beste sein, nach Hause zu gehen, sich ins Auto zu setzen und über die Autobahn nach Wien zu fahren – das sollte normalerweise bis 09:00 Uhr zu schaffen sein.

1.6 Codierung

Information kann nicht nur als Folge von „0“ und „1“ dargestellt werden. Es können auch andere Darstellungsarten, andere **Codes** benutzt werden. Codierung wird umgangssprachlich häufig mit *Verschlüsselung* gleichgesetzt. Allgemein wird darunter jede durch **feste Regeln** bestimmte Ausdrucksweise (z.B. Morsecode, Bild 1.20) verstanden.

Codierung (aus lat. codicillus = „geheimer Brief“) als ein Begriff aus der **digitalen Signalverarbeitung** umfasst auch das Konzept **Kompression**. Eine Kompression ist insbesondere bei multimedialen Daten von großer Wichtigkeit (siehe Basiswissen Multimedia Band 1, Modul 4, Kapitel 4).

Achtung: In der Informatik ist der Code die *Form* eines Programms, in der seine Erstellung stattfindet; die *Formulierung* in einer „Programmiersprache“ (z.B. Fortran, Java, C++...). Es wird von Quellcode gesprochen und sollte nicht mit (informationstechnischer) Codierung verwechselt werden. In der Informationstechnik werden analoge Signale zu diskreten Zeitpunkten abgetastet, dann amplitudenquantisiert und schließlich *codiert*.

Formal wird unter Codierung eine **Abbildung** von einer Sprache auf eine andere verstanden (Bild 1.21): Zeichen eines Quellalphabetes werden auf Zeichen (Zeichenfolgen) eines Zielalphabetes abgebildet (z.B. 15 → 1111). In diesem Fall nennt man die so erhaltene Zeichenfolge **Wort**. Die Menge aller Wörter eines Zielalphabetes – die durch diese Abbildung verwendet werden – wird **Code** genannt.

Ist die Wortlänge des Codes für alle Wörter gleich, wird von fester Wortlänge gesprochen. Ist die Wortlänge variabel (veränderlich), wird von variabler Wortlänge gesprochen (z.B. Morsecode, Bild 1.20). Codes mit variabler Wortlänge haben den Vorteil, dass Zeichen, die öfter vorkommen, auf kürzere Wörter abgebildet werden können. Dadurch kann die übertragene Datenmenge reduziert werden.

Nach DIN 44 300 ist der Begriff Codierung folgendermaßen definiert:

Codierung ist eine Vorschrift für die *eindeutige Zuordnung* eines Zeichenvorrats zu dem eines anderen Zeichenvorrats (Bild 1.21).

Es gibt unzählige Codierungsmöglichkeiten. Prinzipiell wird unterschieden zwischen:

- numerischen Codes (**Zahlencodes**) und
- alphanumerischen Codes (**Zeichencodes**).

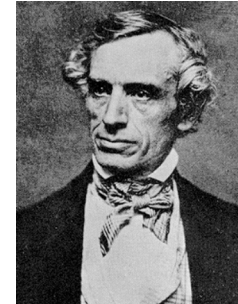


Bild 1.19 Samuel Morse (1791–1872), der Pionier der Codierungstechnik

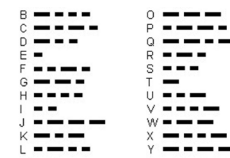


Bild 1.20 Der klassische Morsecode ist ein Code mit variabler Wortlänge, der die Buchstaben durch kurze und lange Stromimpulse codiert

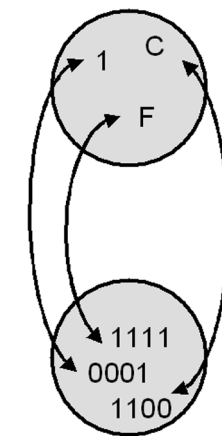
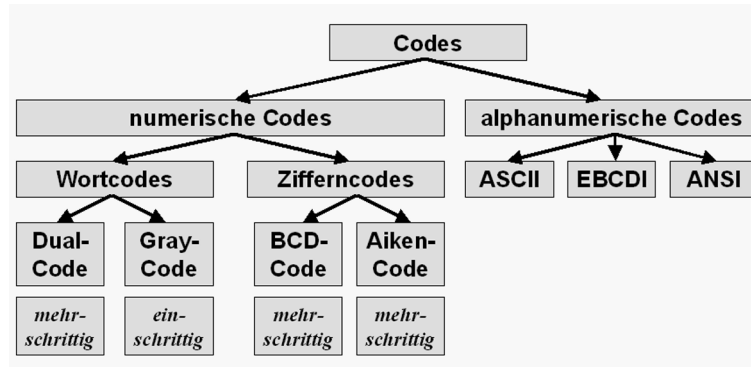


Bild 1.21 Die formale Definition von Codierung als Abbildung von einer Sprache auf eine andere

IT 1 Grundkonzepte

Bild 1.22 Einteilung der Codes; einschrittig = beim Übergang wechselt nur ein Bit; mehrschrittig = beim Übergang wechseln u.U. mehrere Bits gleichzeitig



Dezimalzahl	2 ⁿ⁻¹	Wertigkeiten			
		2 ³	2 ²	2 ¹	2 ⁰
0	0	0	0	0	0
1	0	0	0	0	1
2	0	0	0	1	0
3	0	0	0	1	1
4	0	0	1	0	0
5	0	0	1	0	1
6	0	0	1	1	0
7	0	0	1	1	1
8	0	1	0	0	0
9	0	1	0	0	1
10	1	0	0	0	0
11	1	0	0	0	1
12	1	0	0	1	0
13	1	0	0	1	1
14	1	0	1	0	0
15	1	0	1	0	1
2 ⁿ⁻¹	1	1	1	1	1

1.6.1 Numerische Codes

Die numerischen Codes werden weiter unterteilt in:

- Wortcodes (z.B. Dualcode, Graycode usw.) und
- Zifferncodes (z.B. BCD-Code, Aikencode usw.).

Die **Wortcodes** nehmen eine *wortweise Codierung* vor (Definition „Wort“, siehe Kapitel 1.4.4), d.h., eine Zahl wird „als Ganzes“ codiert, *nicht ziffernweise*. Der bekannteste Wortcode ist der **Dualcode**. Das ist ein reiner Binärcode, der einer Zahlendarstellung im Dualsystem entspricht (Bild 1.23 und siehe Kapitel 3). Die wortorganisierte binäre Codierung hat den Vorteil einer einfachen Arithmetik, allerdings ist die Umcodierung bei Ein- und Ausgabe oder Übertragung relativ schwierig.

Bild 1.23 Dualcode, aus Borucki (1977)

Zugeordnete Ziffer	Wertigkeit
	8 4 2 1
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
	1 0 1 0
	1 0 1 1
	1 1 0 0
	1 1 0 1
	1 1 1 0
	1 1 1 1

Bild 1.24 8-4-2-1-BCD-Code, aus Borucki (1977)

Bei den **Zifferncodes** wird im Gegensatz zu den Wortcodes die Dezimalzahl *ziffernweise* codiert, d.h., jede Ziffer einer Zahl wird getrennt codiert (untercodiert). Zifferncodes haben den Vorteil einer einfacheren Codierung, allerdings den Nachteil, dass *ungenutzte Bitkombinationen* existieren (Pseudotraden). Der bekannteste Zifferncode ist der **BCD-Code** (binär codierte Dezimalziffer). Zifferncodes codieren nur die 10 Dezimalziffern (0 bis 9) und lassen sich weiter unterteilen in:

- **bewertbare Zifferncodes** (8-4-2-1-BCD, Aiken, 1-aus-10-Code, usw.) und
- **Anordnungscodes** (Stibitz, Exzess-3, Craig-Code usw.).

Wird von BCD-Code gesprochen, wird meistens der 8-4-2-1-BCD-Code gemeint (Bild 1.24).

1.6.2 Alphanumerische Codes

Um alphanumerische Zeichen (Buchstaben, Ziffern, Satzzeichen, Sonderzeichen usw.) zu codieren, sind spezielle Codes entwickelt worden. Sollen die zehn Ziffern, die 26 Buchstaben des Alphabetes und 15 Sonderzeichen codiert werden, sind 51 Binärkombinationen, also *mindestens* 6 Binärstellen, notwendig.

Der 5-Spur-Lochstreifencode CCITT Nr. 2 (Baudot-Code) wurde bereits im Jahr 1932 genormt. Da nur 5 Spuren (Wortlänge n = 5) vorhanden sind, können eigentlich nur 2⁵ = 32 Binärkombinationen gebildet werden. Das ist aber für die Codierung von Zahlen und Buchstaben nicht ausreichend. Deshalb werden jeder Binärkombination zwei Zeichen zugeordnet (Doppelbelegung). Ein Umschaltbefehl legt fest, ob es sich bei den folgenden Kombinationen um Buchstaben (Bu = 11111) oder Ziffern (Zi = 11011) handelt.

ASCII (American Standard Code for Information Interchange)

Ursprünglich als Code für Fernschreiber entstanden, bildet der ASCII-Code seit etwa 1963 bis heute die Grundlage der Codierung von Standardzeichen in Personalcomputern (PCs), allerdings nicht in Großrechnern). Als 7-Bit-Code können aber nur 128 Zeichen codiert werden (Bild 1.25). Der ASCII-Code wird auch für das Senden von eMails im Internet verwendet. Allerdings stehen jedoch (bedingt durch die 7 Bit) keine Umlaute zur Verfügung; daher werden Texte mit Umlauten z.B. mit MIME (Multipurpose Internet Mail Extensions) codiert. Die meisten eMail-Programme sind in der Lage, eMails automatisch mit MIME zu codieren bzw. zu decodieren.

ASCII-Art (ASCII-Kunst), siehe z.B. [W4], [W5]

MIME, siehe z.B. [W6]

	HEX	0	1	2	3	4	5	6	7
HEX	BIN	0000	0001	0010	0011	0100	0101	0110	0111
0	0000	NUL	DLE	SP	0	@	P	`	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(8	H	X	h	x
9	1001	HT	EM)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	-	=	M]	m	}
E	1110	SO	RS	.	>	N	^	n	~
F	1111	SI	US	/	?	O	_	o	DEL

Bild 1.25 ASCII-Code nach DIN 66 003; Erklärung der HEX- (= hexadezimal) und BIN- (= binär) Zahlen siehe Kapitel 3, weiteres Beispiel siehe [W7]

IT 1 Grundkonzepte

ANSI-Code (American National Standards Institute)

Es gab einige Versuche von Drucker- und Computerherstellern, den 7-Bit-ASCII-Code auf 8 Bit zu erweitern, ein Beispiel ist der so genannte erweiterte IBM-Zeichensatz.

Der ANSI-Code wird für die Zeichencodierung unter dem Betriebssystem Windows verwendet: Die Steuerzeichen mit den Codenummern 0 bis 31 werden dabei nicht verwendet, die Codenummern 32 bis 127 entsprechen dem ASCII-Zeichensatz, und der Bereich 128 bis 255 ist aus folgendem Bild ersichtlich:

Bild 1.26 Beispiel für den ANSI-Code



EBCDIC-Code (Extended Binary Coded Decimal Interchange Code)

Im Bereich der Großrechner (mainframes) wird hauptsächlich der von IBM entwickelte EBCDIC benutzt. Das ist ein auf 16 Bit erweiterter BCD-Code, der in unterschiedlichen Varianten existiert.

Unicode

Beim ANSI-Code ist die Zuweisung zu den 128 weiteren Zeichen landesspezifisch. Einen anderen Ansatz wählt der seit 1992 verfügbare Unicode, der alle Zeichen mit 16 Bit codiert und daher $2^{16} = 65\,536$ mögliche Zeichen unterstützt.

Der Unicode ist die 16-Bit-Variante des 32-Bit-**UCS-Codes (Universal Character Set, ISO 10646)** und dient zur Darstellung von Schriftzeichen mit Integration des ASCII-Codes und verbreiteter Zeichensätze. Dazu gehören z.B. die Zeichen für Arabisch und Hebräisch, Griechisch, Kyrillisch und Armenisch, Indisch, Einheitszeichen aus dem Chinesischen, Koreanischen und Japanischen (Kanji) sowie mathematische, technische und grafische Symbole.

1.6.3 Strichcodes

Jede Zahl lässt sich (ähnlich wie beim Morsecode) aus einer Kombination von Strich bzw. kein Strich darstellen, so genannte Strichcodes bzw. Balkencodes (Barcodes).

Lineare Barcodes setzen sich aus *Balken* und *Zwischenräumen* zusammen, die nacheinander von einer Seite (z.B. von links nach rechts) eingelesen werden. In diesem „Muster“ sind alphanumerische und/oder numerische Zeichen codiert. Zu den linearen Barcodes zählen z.B. der Code 39, Interleaved 2/5, UPC, Codabar, Code 128 und der Code 93. Der bekannteste ist aber sicher die 13-stellige EAN (europäinheitliche Artikelnummer, siehe Bild 1.27):

Beim Einlesen ergibt sich aus dem Strichmuster eine Folge von Lichtzeichen: Licht, kein Licht, Licht usw., wie beim Morsecode. Wenn die Einlesegeschwindigkeit bekannt ist, ist die Decodierung kein Problem. Deshalb existieren etwas längere Kontrollstriche (zweimal schmal) am Anfang, in der Mitte und am Ende (Bild 1.27).

1.6.4 Flächencodes

2D-Codes sind zweidimensionale Symbologien und setzen sich aus Datenreihen zusammen, die in einem rechteckigen oder quadratischen Muster angeordnet sind.

Die Datenreihen sind so geschichtet, dass sie eine Datenkette bilden, die *gleichzeitig* in zwei Dimensionen gelesen werden muss. In zweidimensionalen Symbolen können etwa fünfzigmal mehr Daten gespeichert werden als in linearen Barcodes. Zu den gängigsten 2D-Codes zählen Maxi-Code, Data Matrix, Aztec-Code (Bild 1.28), QR Code, Code 49 und der kombinierte Code Aztec-Mesa. Zum Einlesen zweidimensionaler Barcodes ist ein Bildscanner erforderlich (siehe Modul 3).

PDF417 ist eine kombinierte lineare Symbologie, die sowohl Charakteristiken linearer Barcodes als auch von 2-D-Barcodes aufweist. Wie eine lineare Symbologie werden PDF417 und MicroPDF417 ebenfalls nacheinander von rechts nach links oder von links nach rechts eingelesen.

PDF417 setzt sich aus mehreren geschichteten Codes zusammen und wird deshalb auch noch vertikal gelesen. Dadurch kann der Benutzer bedeutend mehr Daten auf kleiner Fläche codieren.

Kombinierte, auf lineare Symbologien gestützte Codes wie UCC oder RSS fallen ebenfalls unter diese Gruppe. UCC, RSS, PDF417 und MicroPDF417 können mit Laserscannern, bestimmten CCD-Scannern und Geräten für lineare Bilderfassung gelesen werden.



Bild 1.27 Die EAN, die europäinheitliche Artikelnummer; oben: Beispiel aus dem Alltag; unten: Schema [W8]



Bild 1.28 Der Aztec-Code [W9]

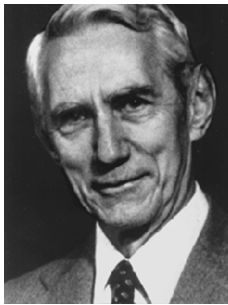


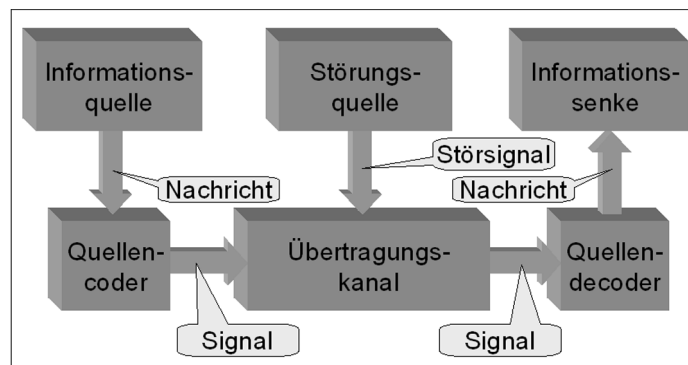
Bild 1.29 Claude E. Shannon (geb. 1918 in Michigan) begründete 1948 die Informationstheorie. Er starb am 24. Februar 2001 [W2]

Bild 1.30 Das nachrichtentechnische Kommunikationsmodell von Shannon & Weaver beschreibt den Informationsfluss zwischen Sender und Empfänger

2 Informationstheorie

Die Informationstheorie entstand aus den Arbeiten von CLAUDE ELWOOD SHANNON und stellt eine grundlegende **Basis für die IT/Informatik** dar. Sie dient der *mathematischen Beschreibung* von Informationssystemen und ist eng an die Nachrichtentechnik angelehnt. Mit Hilfe einer *einheitlichen Theorie* ist es möglich, verschiedene Nachrichtenübertragungsverfahren zu *vergleichen* und die einzelnen Komponenten eines Übertragungssystems zu optimieren.

Ein Informationssystem unterliegt den Gesetzen der Nachrichtentechnik und kann durch das Kommunikationsmodell von SHANNON & WEAVER gemäß Bild 1.30 dargestellt werden.



Die Informationstheorie stellt auch die Grundlage für die Psychophysik (siehe Basiswissen Multimedia Band 1, Modul 3) und die psychologische Analyse von Kommunikation dar.

2.1 Informationsgehalt

Der Informationsgehalt (information content) einer Nachricht wird in der Informationstheorie nach SHANNON & WEAVER aus rein mathematischer Sicht definiert. Die **Bedeutung** der Nachricht wird dabei völlig **ignoriert!** Unter Bezug auf unsere (nachrichtentechnische) Definition des Begriffes Information (Kapitel 1.1.3) muss der Informationsgehalt einer Nachricht folgende drei Eigenschaften erfüllen:

- Der Informationsgehalt I_x einer Nachricht x muss umso größer sein, je kleiner die Wahrscheinlichkeit p_x ihres Auftretens ist (je größer der „Überraschungswert“ ist).
- Eine Nachricht mit der Wahrscheinlichkeit $p_x = 1$ muss den Informationsgehalt $I_x = 0$ haben.
- Der Informationsgehalt von voneinander unabhängigen Nachrichten soll sich addieren.

Allerdings muss, wenn wir von Informationsgehalt sprechen, zunächst unterschieden werden, ob diskrete Zeichen mit gleicher Auftretenswahrscheinlichkeit oder diskrete Zeichen mit unterschiedlicher Auftretenswahrscheinlichkeit verwendet werden.

2.1.1 Informationsgehalt diskreter Zeichen mit gleicher Auftretenswahrscheinlichkeit

Der Informationsgehalt I ergibt sich aus dem Zeichenvorrat s . Nehmen wir z.B. die ersten ganzen Zahlen von 0 bis 15, so ergibt sich ein Zeichenvorrat von $s = 16$. Stellen wir diese Zahlen im Dualcode dar, so erhalten wir:

$$15 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 1111$$

$$14 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 1110$$

... usw. bis

$$0 = 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 0000$$

Um den Zeichenvorrat von $s = 16$ digital darzustellen, brauchen wir eine vierstellige Dualzahl in der Größe von 4 bit, da $4 = \text{lb}(16)$.

Der Informationsgehalt ergibt sich somit folgendermaßen:

$$I = \text{lb}(s)$$

I Informationsgehalt in bit

s Anzahl der verschiedenen Zeichen (Zeichenvorrat)

Die Verwendung des Logarithmus zur Basis 2 ist nur eine Frage des Maßstabes. Die Basis 2 wurde deshalb gewählt, weil für den einfachsten Fall einer symmetrischen, binären Nachricht mit $p_x = 0,5$ der Informationsgehalt eines Zeichens $I_x = 1$ bit wird.

Die Einheit für eine Informationsmenge ist das **bit**.

Achtung: Der Begriff **bit** hat ursprünglich zwei verschiedene Bedeutungen (wird in der Praxis allerdings kaum konsequent unterschieden):

- *Kleingeschrieben* handelt es sich um die **Einheit** für den **Informationsgehalt** einer Nachricht (siehe Erklärung nächste Seite).
- *Großgeschrieben* handelt es sich um den **Namen** für eine „binäre Stelle oder binäre Ziffer“.

Prinzipiell enthält irgendetwas, das zwei Möglichkeiten beinhaltet, den Informationsgehalt von einem bit (Versuch: Lichtschalter ein- und ausschalten oder Münzwurf). Die Binärelemente „0“ und „1“ dienen dazu als Grundbausteine.

lb = binärer Logarithmus zur Basis 2, auch als ld (logarithmus dualis) bzw. log₂ bezeichnet

binary digit, die Einheit der Informationsmenge

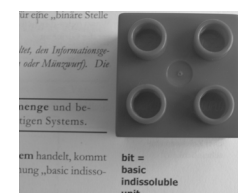


Bild 1.31 Ein Binärelement ist sozusagen der „kleinste Baustein“

Ein bit ist die **kleinste binäre Informationsmenge** und beschreibt den **logischen Zustand** eines zweiwertigen Systems.

**bit =
basic
indissoluble
unit**

Die Tatsache, dass es sich dabei um ein **minimales System** handelt, kommt in der ursprünglich von SHANNON eingeführten Bezeichnung „basic indissoluble unit“ zum Ausdruck.

**1 Shannon (Sh) =
1 bit**

Achtung: Für die Einheit des Informationsgehaltes (die durchaus eine Bruchzahl sein kann) wird daher in der Informationstheorie (zumindest im englischsprachigen Raum) die Bezeichnung **Shannon (Sh)** verwendet.

Im Englischen kommen auch häufig die Einheiten Hartley (Hart) und Natural Unit (nat) vor. 1 Hart = $\log_2 10 = 3\,321\,928$ Sh oder $\log_e 10 = 2\,302\,585$ nats.

Beispiel aus dem „Informationsalltag“:

Wie groß ist der Informationsgehalt eines 7-stelligen Tresorcodes (1234567)?

$s = 10$ (für die Ziffern von 0 ... 9), daraus ergibt sich:
 $I = 7 \text{ lb}(10) \text{ bit} = 7 \cdot 3,32 \text{ bit} = 23,26 \text{ bit} = \underline{\underline{23,26 \text{ sh}}}$

Zur Erklärung: $\text{lb } 10 = {}^2\log 10$

$$2^x = 10 \Rightarrow x \log(2) = \log(10)$$

$$x = \frac{\log(10)}{\log(2)} = \frac{1}{0,30103} = 3,3219281$$

2.1.2 Informationsgehalt diskreter Zeichen mit unterschiedlicher Auftretenswahrscheinlichkeit

In unseren bereits genannten Beispielen wird vorausgesetzt, dass die Wahrscheinlichkeit p_x für das Auftreten aller Zeichen gleich ist.

Die p_x eines Zeichens bei s möglichen Zeichen ist dann

$$p_x = \frac{1}{s}$$

**Laplace-
Wahrscheinlichkeit**

wobei dieser Ausdruck – wie aus der Wahrscheinlichkeitsrechnung bekannt – die Anzahl der günstigen durch die Anzahl der möglichen Fälle darstellt.

Damit gilt für den Informationsgehalt:

$$I_x = lb(s) = lb\left(\frac{1}{p_x}\right)$$

Wenn jedes Zeichen eine individuelle Wahrscheinlichkeit p_{xi} besitzt, so lautet die Gleichung:

$$I_{xi} = lb\left(\frac{1}{p_{xi}}\right)$$

Stellen wir diesen Sachverhalt grafisch dar, erhalten wir folgendes Bild:

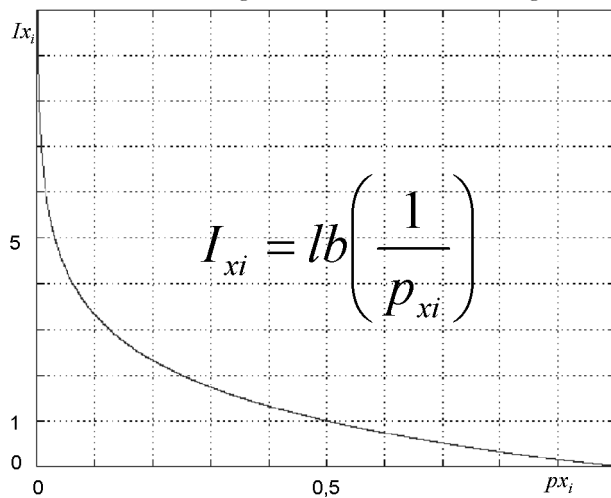


Bild 1.32 Informationsgehalt (bei unterschiedlicher Zeichenwahrscheinlichkeit)

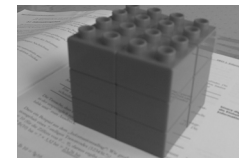


Bild 1.33 Völlig gleichförmige Ordnung von 12 Legosteinen

Der Informationsgehalt soll *unabhängig* von der Form der Codierung sein und ausschließlich von der Wahrscheinlichkeit abhängen, mit der der Empfänger diese Nachricht erwartet.

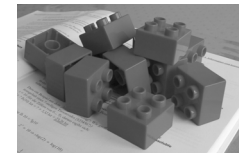


Bild 1.34 Völlige Unordnung

Nachrichten, die mit *hoher Wahrscheinlichkeit* erwartet werden, haben einen *niedrigen* Informationsgehalt, solche, die mit *geringer Wahrscheinlichkeit* erwartet werden, hingegen einen *hohen Informationsgehalt*. Einige Beispiele mit Lego-Bausteinen (Bilder 1.33 bis 1.35) sollen diesen Sachverhalt verdeutlichen und folgende Fragen illustrieren: In welcher Anordnung steckt der größte Informationsgehalt?



Bild 1.35 Dieser Ordnung liegt ein definierter Bauplan zugrunde

Bild 1.33 zeigt eine *völlig gleichförmige Ordnung* aus 12 Lego-Bausteinen: Der *Informationsgehalt ist gering*. In Bild 1.34 sind diese 12 Lego-Bausteine *in völliger Unordnung*: Das Auftreten einer bestimmten Anordnung ist zufällig, daher sehr unwahrscheinlich und hat damit *höchsten Informationsgehalt*. In der Regel ist es ein „Mittel“: eine definierte Ordnung, z.B. das „Haus“ in Bild 1.35.

2.2 Entropie

Entropie ist der Grad der Unordnung in einem System

Der Begriff Entropie (entropy) wurde aus der Wärmelehre (Thermodynamik) entlehnt. Dort ist die Entropie ein Maß für die statistische Unordnung (Unsicherheit) in einem thermodynamischen System.

Nun können wir für den **mittleren Informationsgehalt** einer Nachricht den arithmetischen Mittelwert als Summe aller mit p_{xi} multiplizierten Einzelinformationsgehalte I_i nehmen:

$$H = \bar{I}_x = \sum_{i=1}^s p_{xi} \text{lb} \left(\frac{1}{p_{xi}} \right)$$

Ein höherer Entropiewert H bedeutet eine größere Unsicherheit darüber, was die Nachricht besagt

H bezeichnet dabei die **nachrichtentechnische Entropie**, den mittleren Informationsgehalt einer Nachricht.

2.3 Redundanz

Die **maximale** nachrichtentechnische Entropie wird auch als **Entscheidungsgehalt** H_0 bezeichnet. Diese ist dann gegeben, wenn die einzelnen Nachrichtenelemente gleich verteilt und damit statistisch unabhängig sind.

Eine Abweichung von der Gleichverteilung bewirkt eine Verringerung des mittleren Informationsgehaltes einer Nachricht. Diese Verringerung wird als **Redundanz** R bezeichnet.

Es gelten:

$$R = H_0 - H$$

und

$$r = \frac{H_0 - H}{H_0}$$

Redundanz ist nicht nur keine Information in unserem Sinn, sie ist vielmehr Verlust an Information

R Redundanz
 H_0 Entscheidungsgehalt (mittlere Entropie)
 r relative Redundanz

Beispiel: Würde in der deutschen Sprache die Auftretenswahrscheinlichkeit aller Buchstaben ($s = 26$) gleich sein ($p_x = 1/26$) und wären die Buchstaben auch noch unabhängig voneinander, wäre der Informationsgehalt eines Buchstabens (und ebenso der Mittelwert) $H_0 = I_x = \text{lb}(26) = 4,7$ bit.

In der Praxis ist es aber so, dass der Buchstabe „e“ weit häufiger auftritt als der Buchstabe „x“. Unter Berücksichtigung dieser Buchstabenhäufigkeit ist die Entropie $H = 4,1$ bit/Symbol.

Es muss aber auch berücksichtigt werden, dass beispielsweise mit höherer Wahrscheinlichkeit auf ein „a“ ein „n“ als ein „o“ folgt. Unter Berücksichtigung dieser bedingten Wahrscheinlichkeiten wird $H = 3$ bit/Symbol.

Werden zusätzlich noch die Abhängigkeiten der bedingten Wahrscheinlichkeit von allen längeren Buchstabenfolgen und auch die Unmöglichkeit des Auftretens bestimmter Buchstabenfolgen berücksichtigt, ergibt sich ein Wert von $H = 1,3$ bit/Symbol.

Damit können folgende Werte für die deutsche Sprache angesetzt werden:

$$\begin{aligned} H &= 1,3 \text{ bit/Symbol} \\ R &= 3,4 \text{ bit/Symbol} \\ r &= 0,73 \end{aligned}$$

**Informations-
theoretische
Werte der
deutschen Sprache**

73 % der deutschen Sprache sind redundant („überflüssig“),
nur 27 % sind informationstragend.

Diese Redundanz der Sprache ist **nicht** unerwünscht. Obwohl diese zwar die eigentliche Nachricht verlängert und den Informationsfluss verlangsamt, wird dadurch eine Fehlererkennung und Fehlerkorrektur bei gestörten Nachrichten ermöglicht.

**Information ohne
Redundanz ist völlig
wertlos, da sie keine
Verlässlichkeit mit-
bringt**

2.4 Informationsfluss

Die Übertragungsgeschwindigkeit einer Information wird eine immer wichtigere Größe. Diese Geschwindigkeit wird Informationsfluss genannt und folgendermaßen definiert:

$$F = \frac{H}{t_m}$$

wobei:

$$t_m = \sum p_x t_x$$

F Informationsfluss in bit/s

t_m mittlere Zeitdauer für die Übertragung eines Symbols in s/Symbol

2.5 Kanalkapazität

Für verteilte Informationssysteme (z.B. netzbasierende Multimedia-Systeme) ist es besonders wichtig, die obere Grenze der Leistungsfähigkeit der verwendeten Übertragungskanäle zu kennen.

Unter „Kanal“ verstehen wir einen beliebigen Weg, den eine Nachricht vom Sender zum Empfänger nehmen kann.

Die Kanalkapazität (channel capacity) C ist der **maximale Informationsfluss**, der über den jeweiligen Nachrichtenkanal (channel) fehlerfrei übertragen werden kann.

Der Informationsfluss ist abhängig von der Informationsquelle und vom Übertragungskanal.

Die Kanalkapazität ist ein wichtiges Gütekriterium für Informationssysteme

Die Kanalkapazität ist hingegen unabhängig von den Eigenschaften der Informationsquelle und hängt nur vom Übertragungskanal selbst ab.

Es handelt sich um eine absolute obere Leistungsgrenze; sie kann folgendermaßen berechnet werden:

$$C = F_{max} = \left(\frac{H}{t} \right)_{max}$$

C Kanalkapazität in bit/s

Für einen durch Rauschen (auch Rauschen ist – allerdings in diesem Fall unerwünschte – Information) gestörten Übertragungskanal mit einer vorgegebenen Bandbreite B gilt:

$$C = B \log_2 \left(1 + \frac{P_S}{P_N} \right)$$

B Bandbreite in Hz (1/s)

P_S Signalleistung in W (S = Signal)

P_N Rauschleistung in W (N = Noise)

In der Praxis bleibt der Informationsfluss in einem Übertragungskanal weit unter der Kanalkapazität. Gehen wir unter Berücksichtigung von

$$\log_2(x) = \frac{\lg(x)}{\lg(2)}$$

auf den Logarithmus zur Basis 10 über, erhalten wir mit $\lg(2) \sim 0,3$:

$$C \approx \frac{B}{3} 10 \lg \left(1 + \frac{P_S}{P_N} \right)$$

wobei

$$10 \lg \left(1 + \frac{P_S}{P_N} \right) = \rho$$

Dieses ρ stellt das **Signal-Rausch-Verhältnis** (in dB) dar, und wird in der Praxis als **Störabstand** bezeichnet.

dB = Dezibel, siehe Modul 3, Kapitel 2.3.2

In den meisten Fällen ist das Signal sehr viel größer als das Rauschen, dann wird mit

$$P_S \gg P_N$$

$$\rho \approx 10 \lg \left(\frac{P_S}{P_N} \right)$$

Die Kanalkapazitäten von Übertragungskanälen sind sehr unterschiedlich:

Übertragungskanal	B	r	C
Telegrafie	25 Hz	15 dB	75 bit/s
Analoges Telefon	3,1 kHz	40 dB	50 kbit/s
UKW-Radio	15 kHz	60 dB	300 kbit/s
Fernsehen	5 MHz	45 dB	75 Mbit/s
ISDN	64 kHz	60 dB	64 kbit/s
Breitband (ATM)	10 MHz	45 dB	155 Mbit/s
Vergleich: Auge	285 THz	-	10 Mbit/s

Bild 1.36 Kanalkapazitäten einiger typischer Übertragungskanäle
 B = Bandbreite
 r = Störabstand
 C = Kanalkapazität

Die Bandbreitenproblematik wird im Modul 6 „Multimedia via Internet“ in Basiswissen Multimedia Band 1 behandelt

Durch die Zunahme der Internet-Nutzer und das wachsende Angebot im Multimedia-Bereich ist ein steigender Bedarf an Kanalkapazität bei stetig steigender Datenrate an den Schnittstellen zu den Nutzern entstanden. Datenraten im Gbit/s- und Tbit/s-Bereich werden heute nur mehr über optische Übertragungskanäle (Glasfaser) über weite Strecken transportiert. Die Glasfaser kommt immer näher zum Nutzer (FTTH).

FTTH = „fiber to the home“

2.6 Informationsquader

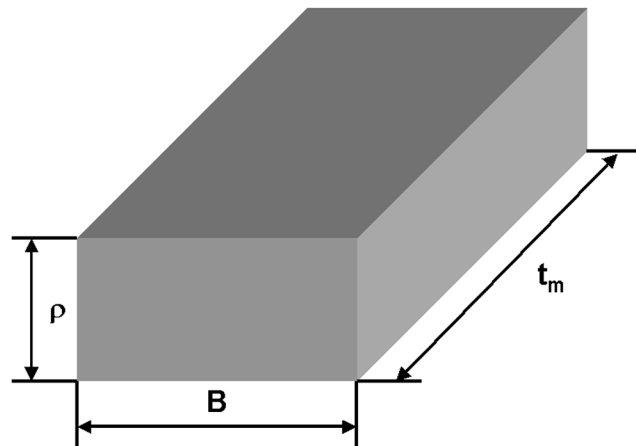
Die übertragbare Informationsmenge (das „Informationsvolumen“) kann anschaulich als Quader – manchmal auch „Nachrichtenquader“ genannt – dargestellt werden (Bild 1.37).

Die Kantenlängen dieses Quaders (die Parameter) hängen ab

- von der verfügbaren Bandbreite B ,
- vom Störabstand ρ und
- von der zur Verfügung stehenden Übertragungszeit t_m .

Ein Informationsquader kann grafisch dargestellt werden:

Bild 1.37 Informationsquader, mit Bandbreite B , Störabstand ρ und Übertragungszeitfenster t_m



Die richtige Dimensionierung dieser drei Parameter ist notwendig für eine optimale Übertragung der Information.

Beispiel: Für einen Internet-Service-Provider (ISP) sind besonders die Bandbreite und die Übertragungszeit wichtige Größen zur Bemessung der Anbindungen (Kanalkapazitäten) seiner Kunden an das externe Netzwerk (Internet).

Nach SHANNON ist das Volumen des Informationsquaders gleich der theoretischen Grenze für die maximal übertragbare Informationsmenge.

3 Zahlensysteme und binäre Arithmetik

Eine Vorstufe des Rechnens ist das Zählen. Im Altertum wurden als Hilfsmittel zunächst die Finger (lat. *digitus*), Kerbhölzer – *das germanische Wort „Zahl“ bedeutet Kerbe* – und aufgefädelte Perlen als Rechenbretter (Abacus, Bild 1.38) verwendet.



Bild 1.38 Abacus ... ein Beispiel für ein frühes Rechengerät

3.1 Zahlensysteme: Grundlage für das Rechnen

Eine wesentliche Grundlage für das „echte“ Rechnen sind *Zahlensysteme*, mit deren Hilfe Zahlen durch Zeichen bzw. Zeichenreihen dargestellt werden.

Die ersten Zahlensysteme entstanden vor etwa 5000 Jahren und die bekanntesten sind das römische Zahlensystem und das zwischen dem 5. und dem 9. Jahrhundert entstandene hindu-arabische Zahlensystem, das als Dezimalsystem heute bei uns in Gebrauch ist. Es zeichnet sich durch die verwendeten zehn Ziffern 0, 1, 2, 3, ..., 9 und das Stellenwertprinzip aus. Notwendige Voraussetzung für ein Stellenwert- oder Positionssystem ist das Vorhandensein der Null. Sie wurde erst um 500 n. Chr. eingeführt.

Ein Zahlensystem besteht aus der Gesamtheit der zur Darstellung einer Zahl verwendeten Zahlzeichen (Ziffern) und den Regeln für deren Zusammensetzung.

Man unterscheidet grundsätzlich zwei Gruppen von Zahlensystemen:

- **Polyadische Zahlensysteme** (Stellenwertsysteme, Positionssysteme): Die Bedeutung einer Ziffer hängt von ihrer Stellung innerhalb der Zahl ab; Beispiel: Dezimalsystem.
- **Nichtpolyadische Zahlensysteme** (z.B. additive Zahlensysteme): Die Zahlzeichen werden aneinander gereiht, der Zahlenwert ergibt sich durch einfaches Zusammenzählen; Beispiel: römisches Zahlensystem)

I = 1	
V = 5	IV = 4
X = 10	IX = 9
L = 50	XL = 40
C = 100	XC = 90
D = 500	CD = 400
M = 1000	CM = 900

Zum Rechnen sind nur die *polyadischen Zahlensysteme* geeignet.

Für polyadische Zahlensysteme gelten folgende Vereinbarungen:

- Die **Basis** B ist eine natürliche Zahl > 1 (z.B. Dualsystem $B=2$),
- eine **Ziffer** ist ein Zeichen aus dem Zeichenvorrat von B Zeichen,
- der **Nennwert** der Ziffer ist eine ganze Zahl aus $\{0, 1, \dots, B-1\}$,
- der **Stellenwert** entspricht der Stelle innerhalb der Ziffernfolge,
- der **Ziffernwert** = Nennwert \cdot Stellenwert,
- eine **Zahl** ist die Folge von Ziffern des verwendeten Zahlensystems,
- der **Zahlenwert** ist die Summe der einzelnen Ziffernwerte.

Bild 1.39 Römisches Zahlensystem. Links: Ziffern und ihre Wertigkeiten; rechts: die Ausnahmen

Ein polyadisches Zahlensystem mit der Basis B , auch B -adisches Zahlensystem genannt, ist ein Zahlensystem, in dem eine Zahl nach **Potenzen** von B zerlegt werden kann.

Eine Zahl setzt sich also zusammen aus:

$$\sum_{k=0}^n a_k B^k = z$$

- z Wert der Zahl
 a Nennwert der Ziffer
 B Basis
 n Anzahl der Stellen

Am geläufigsten ist uns im täglichen Leben das **Dezimalsystem** (aus lat. decem = „zehn“; Basis $B=10$; zulässige Ziffern sind $\{0\dots9\}$).

Für die Informationstechnik/Informatik sind aber insbesondere folgende Zahlensysteme wichtig:

- **Hexadezimalsystem** (aus lat. hexa und lat. decem = „sechzehn“; Basis $B=16$; zulässige Ziffern sind $\{0\dots9, A, B, C, D, E, F\}$),
- **Oktalsystem** (aus lat. octo = „acht“; Basis $B=8$; zulässige Ziffern sind $\{0\dots7\}$) und insbesondere das
- **Dualsystem** (aus lat. duo = „zwei“; engl. „binary system“, Basis $B=2$; zulässige Ziffern sind $\{0, 1\}$).

In der Anfangszeit der Computer hat sich gezeigt, dass das im Alltagsleben verwendete Dezimalsystem für die Verwendung in den Maschinen nicht geeignet ist. Das liegt daran, dass die verfügbaren eindeutigen Zustände von elektromechanischen Schaltern (Relais) und den elektronischen Schaltern (Röhren, Transistoren) sich in der Regel auf 2 beschränken, also z.B. auf „aus/ein“ bzw. „nichtleitend/leitend“, was „0“ oder „1“ entspricht.

Daraus ergibt sich die Notwendigkeit, ein Zahlensystem zu verwenden, das nur **2 Zustände** kennt. Das ist ein „binäres“ oder „duales“ Zahlensystem, das als grundlegende Ziffern eben nur die „0“ und die „1“ kennt.

Man kann mit einem auf 2 statt auf 10 Ziffern beruhenden Zahlensystem ebenso gut und so genau wie mit dem dezimalen System rechnen. Allerdings wäre dieses für den Alltagseinsatz allein wegen der Länge der entstehenden Zahlen unhandlich.

Für interne Darstellungen in Computern werden neben dem Binärsystem auch solche auf Basis der Zahl 8 („oktal“) und der Zahl 16 „hexadezimal“ verwendet:

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

Bild 1.40 Gegenüberstellung der wichtigsten Zahlensysteme in der IT

Die Umwandlung einer Dualzahl in eine Dezimalzahl ist ein sehr einfacher Vorgang:

Beispiel:

$$\begin{aligned}
 (1010.011)_{\text{dual}} &= 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} \\
 &= 8 + 0 + 2 + 0 + 0 + 1/4 + 1/8 \\
 &= 10 + 0,25 + 0,125 = 10,375
 \end{aligned}$$

Bild 1.41 Beispiel für eine Umwandlungsoperation: Dual → Dezimal

Ungewohnter ist die Umwandlung einer Oktalzahl in eine Dezimalzahl:

Beispiel:

$$\begin{aligned}
 (630.4)_{\text{oktal}} &= 6 \cdot 8^2 + 3 \cdot 8^1 + 0 \cdot 8^0 + 4 \cdot 8^{-1} \\
 &= 6 \cdot 64 + 3 \cdot 8 + 4 \cdot 1/8 \\
 &= 384 + 24 + 4 \cdot 0,125 \\
 &= 408,5
 \end{aligned}$$

Bild 1.42 Beispiel für eine Umwandlungsoperation: Oktal → Dezimal

Die wichtigsten Operationen mit Dualzahlen zusammengefasst:

Operation	Ergebnis	Übertrag auf nächsthöhere Stelle
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1 = Übertragbit
0 - 0	0	0
0 - 1	1	1 = „Borgbit“
1 - 0	1	0
1 - 1	0	0
0 × 0	0	0
0 × 1	0	0
1 × 0	0	0
1 × 1	1	0

1. Addition

$$\begin{array}{r}
 1010_2 = 10_{10} \\
 + 1011_2 = +11_{10} \\
 \hline
 \text{(Übertrag)} \quad 1 \\
 \hline
 10101_2 = 21_{10}
 \end{array}$$

2. Subtraktion

$$\begin{array}{r}
 1101_2 = 13_{10} \\
 - 1010_2 = -10_{10} \\
 \hline
 \text{(Borgen)} \quad 1 \\
 \hline
 0011_2 = 3_{10}
 \end{array}$$

Bild 1.43 Die wichtigsten Operationen zusammengefasst

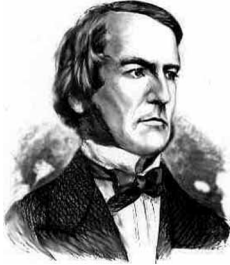


Bild 1.44 George Boole (1815–1864), der Entwickler der Boole'schen Algebra [W5]

3.2 Boole'sche Algebra

Zur Lösung mathematischer Probleme (z.B. zur Lösung einer Gleichung) existieren Hilfsmittel, die unter der Bezeichnung **Algebra** (aus arab. al-gabr = „Wiederherstellung gebrochener Teile“) zusammengefasst werden.

Auch in der Digitaltechnik müssen komplexe Probleme mit geeigneten mathematischen Methoden behandelt werden. Allerdings ist ja die Anzahl der möglichen Informationszustände beschränkt: auf genau zwei.

GEORGE BOOLE entwickelte bereits 1854 eine algebraische Struktur, die für ein zweiwertiges Informationssystem optimal geeignet ist.

Diese Algebra wird inzwischen als „Boole'sche Algebra“ bezeichnet. Ihre große Bedeutung beruht auf ihren vielfältigen Anwendungsmöglichkeiten: Sie eignet sich zur Behandlung von Problemen der Mengenalgebra sowie zur algebraischen Darstellung der Aussagenlogik und zur algebraischen Beschreibung und Manipulation von *digitalen Schaltkreisen*.

Die Arbeiten von BOOLE wurden von anderen Mathematikern ergänzt, u.a. von WILLIAM HAMILTON (1788–1856), AUGUSTUS DE MORGAN (1806–1871), ERNST SCHRÖDER (1841–1902), JOHN VENN (1834–1923) und CLAUDE E. SHANNON (1918–2001).

CLAUDE E. SHANNON erkannte insbesondere die Anwendungsmöglichkeiten der Boole'schen Algebra bei der Behandlung von Schaltnetzen. Er zeigte, dass die prinzipiellen Eigenschaften von Serien- und Parallelschaltungen mechanischer **Schalter** mit Hilfe der zweiwertigen Logik besonders gut beschrieben werden können.

Schalter

Daher wird dieses Spezialgebiet der Boole'schen Algebra heute allgemein als **Schaltalgebra** bezeichnet.

Die Boole'sche Algebra wird hier in vereinfachter Form als System von Regeln (Postulaten und Theoremen) eingeführt, indem die Verknüpfungen zwischen den binär dargestellten Informationen beschrieben werden. Die zugrunde liegenden Operationen werden algebraisch durch die Funktionen vermittelt, die als Boole'sche Funktionen bezeichnet werden.

Die **Boole'schen Funktionen** zeichnen sich durch zwei Eigenschaften aus:

- Jedes **Funktionsargument** (jeder Operand) kann nur den Wert „0“ oder „1“ annehmen,
- Der **Funktionswert** (also das Ergebnis, das nach Anwendung der Funktion auf die Operanden erhalten wird) kann ebenfalls nur den Wert „0“ oder „1“ annehmen.

Anmerkung: Der Begriff der Boole'schen Funktion muss ausdrücklich unterschieden werden von dem Begriff der Binäroperation bzw. der Binärfunktion: Eine uneingeschränkt definierte Funktion kann eine beliebig große Anzahl von Argumenten (Operanden) n haben.

Hat die Funktion $n = 1$ Operanden, so sprechen wir von einer so genannten „unären Operation“. Das bekannteste Beispiel einer unären Operation ist die **Vorzeichenoperation** („+“, „-“).

**unäre Operation
(z.B. Vorzeichen-
operation)**

Hat die betrachtete Funktion $n = 2$ Operanden, so sprechen wir von einer „binären Operation“ (Binäroperation). Beispiele sind die Operationen „Addition“, „Subtraktion“ usw.

**binäre Operation
(z.B. Addition,
Subtraktion)**

Achtung: Der gleiche Begriff „Binäroperation“ wird häufig verwendet, wenn die Argumente nur binäre Werte (0/1) annehmen dürfen. Da an dieser Stelle auch Operationen behandelt werden sollen, die eine beliebige Anzahl binärer (dualer) Operanden besitzen können, entsteht ein Begriffskonflikt.

Um diese *Mehrdeutigkeit* zu vermeiden, wird für alle n -stelligen Funktionen ($n = 1, 2, \dots$), deren Argumente und Resultate *binäre Größen* sind, der Begriff „**Boolesche Funktion**“ verwendet.

Mengendarstellungen in der Boole'schen Algebra

Um die Regeln der Boole'schen Algebra anschaulicher einzuführen, können aus der Mengenlehre bekannte Anschauungsformen herangezogen werden. Die grafische Darstellung durch **Mengendiagramme** (Venn-Diagramme) erweist sich in diesem Zusammenhang als besonders geeignet (Bild 1.45).

Eine **Menge** ist die Zusammenfassung bestimmter, unterscheidbarer **Objekte** zu einem Ganzen. Die unterscheidbaren Objekte werden als **Elemente der Menge** bezeichnet.

Eine Menge kann definiert werden durch die Aufzählung ihrer Elemente.

Beispiele für Mengen:

- Menge der Dezimalziffern: $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- Menge der Binärwerte: $B = \{0, 1\}$
- Menge der binären 2-Tupel: $C = \{[0,0], [0,1], [1,0], [1,1]\}$
- Menge der Wahrheitswerte: $D = \{\text{true}, \text{false}\}$

IT 1 Grundkonzepte

Zusammenfassend gibt die folgende Tabelle einen Überblick über die Boole'schen Funktionen (siehe die technologische Umsetzung in Kapitel 5) und deren Darstellung durch Venn-Diagramme (Bild 1.45):

Bild 1.45 Übersicht über die Boole'schen Verknüpfungen (technologische Umsetzung in Form von Gattern in Kapitel 5)

Wertetabelle	Boole'sche Funktion		Bezeichnung in der Aussagenlogik	verbale Beschreibung	alternative Darstellung	Venn-Diagramm a b
0 0 0 0	$f_0 = 0$		Kontradiktion	Nullfunktion (Nie)		
0 0 0 1	$f_1 = a \wedge b$		Konjunktion	UND	$a \cdot b, a \& b$	
0 0 1 0	$f_2 = \bar{a} \wedge b$	$b \rightarrow a$	Inhibition	wenn b dann \bar{a}		
0 0 1 1	$f_3 = b$		Identität			
0 1 0 0	$f_4 = a \wedge \bar{b}$	$a \rightarrow b$	Inhibition	wenn a dann \bar{b}		
0 1 0 1	$f_5 = a$		Identität			
0 1 1 0	$f_6 = (a \wedge \bar{b}) \vee (\bar{a} \wedge b)$	$a \neq b$	Antivalenz	Exclusiv ODER	$a \oplus b$	
0 1 1 1	$f_7 = a \vee b$		Disjunktion	ODER	$a + b$	
1 0 0 0	$f_8 = \overline{a \vee b}$		Peircefunktion	NOR	$a \downarrow b, a * b$	
1 0 0 1	$f_9 = (a \vee \bar{b}) \wedge (\bar{a} \vee b)$	$a \equiv b$	Äquivalenz	gleichwertig	$a \leftrightarrow b$	
1 0 1 0	$f_{10} = \bar{a}$		Negation		$\neg a$	
1 0 1 1	$f_{11} = \bar{a} \vee b$	$a \rightarrow b$	Implikation	Wenn a dann b	$a \supset b$	
1 1 0 0	$f_{12} = \bar{b}$		Negation		$\neg b$	
1 1 0 1	$f_{13} = a \vee \bar{b}$	$b \rightarrow a$	Implikation	Wenn b dann a	$b \supset a$	
1 1 1 0	$f_{14} = \overline{a \wedge b}$		Shefferfunktion	NAND	$a b$	
1 1 1 1	$f_{15} = 1$		Einsfunktion	Tautologie (immer)		

Venn-Diagramme wurden bereits ab 1880 von JOHN VENN (1834–1923) eingeführt, um logische Zusammenhänge zu visualisieren (darzustellen).

4 Computer-Organisation

Hier sprechen wir nur über grundlegende Prinzipien, um einen Überblick zu erhalten. Im nachfolgenden Modul 2 „Computersysteme“ wird der Aufbau (Architektur) besprochen.

Wenn wir über „Computer“ sprechen, meinen wir damit heute eine Vielzahl verschiedener Aspekte aus dem Hardware- und Software-Bereich. Unter den beiden oft äquivalent gebrauchten Begriffen „Pervasive Computing“ und „Ubiquitous Computing“ wird die heutige Allgegenwärtigkeit der Informationstechnik und der Zugriff auf Informationen zu jeder Zeit, an jedem Ort verstanden.

Das Bild des Computers als (digitale) „Rechenmaschine“ hat sich endgültig verändert. Internetfähige Handys, Personal Digital Assistants (PDAs), die drahtlos („wireless“) mit anderen Geräten kommunizieren, das sind nur die Vorboten eines aufkommenden „Post-PC-Zeitalters“. Informationstechnik und Informatik wachsen dabei immer weiter zusammen (Bild 1.46).

pervasive = „durchdringend, verbreitet“; ubiquitous = „allgegenwärtig“



Bild 1.46 ... auch das ist ein Computer!

4.1 Definitionen

Computer sind *Maschinen* zur Ausführung von Instruktionen. Im Gegensatz zu *Rechenmaschinen*, die schon vor Jahrhunderten gebaut wurden, handelt es sich aber beim Computer um eine *transklassische Maschine* – eine universelle Maschine, deren Verwendungszweck nicht durch die gegebene Hardware vorbestimmt wird (Bild 1.47). Ein und derselbe Computer kann, *mit dem entsprechenden Programm*, als Schreibmaschine, als Softwareentwicklungstool, als Kriegswaffe, als Spiel ... oder als Lernhilfe verwendet werden.

Instruktionen sind Anweisungen – sie beschreiben, wie eine bestimmte Aufgabe auszuführen ist

Computer = transklassische (universelle) Maschine

Bei einer *klassischen Maschine* – beispielsweise einer *Handbohrmaschine* – kann der jeweilige *Verwendungszweck* bereits aus dem *mechanischen Aufbau* erschlossen werden.

Der Computer ist eine **transklassische Maschine**, deren Verwendungszweck erst durch die Software bestimmt wird.

Hardware (wörtl. „harte Ware“, Eisen- bzw. Metallwaren, im Sinne von „nicht veränderbar“) ist die Bezeichnung für die Gesamtheit der *physikalischen* Komponenten (Teile) eines Computersystems. Zur Nutzung dieser Hardware ist Software erforderlich.

Software (wörtl. „weiche Ware“, im Sinne von „veränderbar“) ist die Bezeichnung für die Programme (Instruktionen) und die zugehörigen Daten. Es wird unterschieden zwischen Systemsoftware (Betriebssystem) zur Funktion des Computersystems und Anwendersoftware (Programm, Anwendung) zur Lösung von Problemen bei den Anwendern.



Bild 1.47 Gegenüber einer klassischen Maschine wird beim Computer der Verwendungszweck erst durch Software bestimmt

4.2 Schichtenmodell von Tanenbaum

TANENBAUM (1996) verwendet ein Schichtenmodell, das anschaulich das Zusammenwirken von Informationstechnik (Hardware) und Informatik (Software) zeigt. Prinzipiell besteht sein Modell aus n Ebenen (Level bzw. Schichten). Die meisten heute verwendeten Computersysteme basieren auf 6 Ebenen (0–5), wir beginnen mit der untersten (der „hardware-nächsten“):

siehe nächstes
Kapitel „Schaltnetze
und Schaltwerke“

siehe Modul 2
„Computersysteme“

siehe Modul 2
„Computersysteme“

siehe
Band 2, Modul 2
„Datentechnik“

siehe
Band 2, Modul 2
„Datentechnik“

siehe
Band 2, Modul 2
„Datentechnik“

- **Level 0:** Die Ebene der **digitalen Logik** (Digital Logic Level) ist die Ebene der Gatter und Register. Gatter sind **logische Elemente** (Digitalbausteine, Digitalschaltungen, „Chips“), die selbst wiederum aus Halbleiterbauelementen (Transistoren) aufgebaut sind. Sie verarbeiten und verknüpfen digitale *Eingangssignale* jeweils zu bestimmten digitalen *Ausgangssignalen* (siehe nächstes Kapitel: Schaltnetze und Schaltwerke).
- **Level 1:** Die **Mikro-Architektur-Ebene** (Microarchitecture Level) ist eine Zusammenfassung von Gattern und Registern in der Form von **logischen Einheiten** (z.B. arithmetisch-logische Einheit, **arithmetic logical unit**, kurz: ALU). Damit können bereits *Grundoperationen* ausgeführt werden. Register dienen der Zwischenspeicherung. Sie sind über Datenbusse (siehe Modul 2) miteinander verbunden.
- **Level 2:** Die **ISA-Ebene** (Instruction Set Architecture Level) wird z.B. in den Handbüchern der Computer- bzw. Prozessorhersteller behandelt. Hier sind die *Instruktionen* verzeichnet, die vom Mikroprogramm des Computers interpretiert oder zunehmend, weil das schneller geht, direkt von der Hardware ausgeführt werden.
- **Level 3:** Die **Betriebssystem-Ebene** (Operating System Machine Level) enthält zusätzlich zu grundlegenden Instruktionen auch *komplexere Instruktionen* und eine weiter entwickelte Speicherorganisation. Hier können auch mehrere Programme *gleichzeitig* ablaufen („Multitasking“). Manche Anweisungen werden vom Betriebssystem direkt ausgeführt, andere vom Mikroprogramm (Level 2) bzw. durch die Hardware selbst (Level 1).
- **Level 4:** Die **Assembler-Ebene** beginnt erstmals die Kommunikation mit der Maschine für den Menschen „verständlicher“ zu machen: Assemblersprachen bestehen aus *verbalen Abkürzungen*. Ein Programm, der Assembler, übersetzt diese „Abkürzungen“ in Maschinsprache. Diese wird wieder auf Level 3, 2 oder 1 physikalisch ausgeführt oder von einer so genannten „virtuellen Maschine“ interpretiert.
- **Level 5:** Die **Hochsprachen-Ebene** (Level of High Level Languages) enthält schließlich die Sprachen, (z.B. C, C++, Java usw.) die speziell für die Anwendungsprogrammierung konzipiert wurden. Früher wurden Programme dieser Ebene meist vor Ablauf mit einem Compiler übersetzt (compiliert), heute wieder zunehmend direkt interpretiert (wenn der zusätzliche Zeitaufwand vertretbar ist).

5 Schaltnetze und Schaltwerke

5.1 Schaltkreise

Nahezu alle Teile in Computern sind heute durch **integrierte Schaltkreise** (integrated circuits, ICs) realisiert. Solche ICs enthalten Schaltnetze und Schaltwerke, die selbst wiederum aus aktiven Bauelementen (Transistoren) und passiven Bauelementen (Kondensatoren, Widerstände) bestehen.

5.1.1 Moore-Gesetz

Ein wichtiger Faktor bei der Entwicklung leistungsfähiger Computer ist die Anzahl der Transistoren, die auf einem Chip untergebracht werden können: *Je mehr Transistoren pro Chip integriert werden, desto größere Speicherleistungen und schnellere Prozessoren können realisiert werden.*

MOORE (1965) beobachtete, dass es ungefähr alle drei Jahre zu einer neuen Generation von Speicherchips kommt und dass jede neue Generation gegenüber deren Vorgänger etwa viermal so viel Kapazität hat. Er fasste diese Beobachtung in seinem so genannten Moore-Gesetz zusammen:

Die Anzahl der Transistoren auf einem Chip verdoppelt sich etwa alle 18 Monate.

Das entspricht einem jährlichen Wachstum von 60%. Diese Prognose hat sich bis jetzt bestätigt. Branchenkenner vermuten, dass das Moore-Gesetz noch bis etwa 2020 seine Gültigkeit behält. Dann wird man an physikalische Grenzen stoßen.

Neben dem Moore-Gesetz gibt es noch drei andere Gesetze, die zusammengefasst als „Digital Power“ bezeichnet werden:

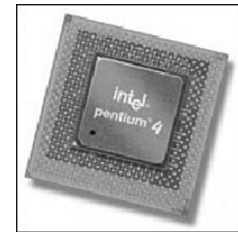
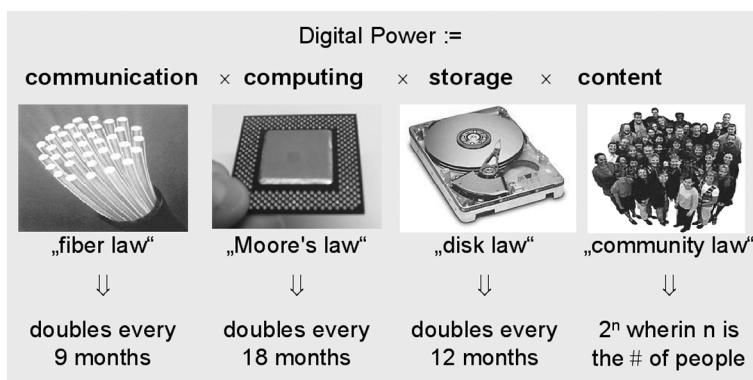


Bild 1.48 Prozessoren werden in hochintegrierter Halbleitertechnologie gefertigt



Bild 1.49 Gordon Moore, Mitbegründer und Vorsitzender von Intel, veröffentlichte 1965 das Moore-Gesetz [W14]

Bild 1.50 Die „Digital Power“ wird durch vier Gesetze bestimmt



Bild 1.51 Jack Kilby baute bei Texas Instruments den ersten integrierten Schaltkreis

5.1.2 Halbleitertechnologie

Die Schwedische Akademie der Wissenschaften hat den Physik-Nobelpreis des Jahres 2000 drei Forschern zuerkannt, die maßgeblich die Grundlagen der Mikroelektronik formten: ZHORES ALFEROV (geb. 1930) und HERBERT KROEMER (geb. 1928) haben optische Komponenten auf Basis geschichteter Halbleiter entwickelt. Diese extrem schnell schaltenden Bauteile finden sich heute sowohl in Kommunikationssatelliten als auch in Mobilfunkstationen. JACK KILBY (geb. 1923) baute 1958 die erste integrierte Schaltung (Bild 1.51) und leitete damit das Zeitalter der Mikroelektronik ein.

Halbleiter sind *kristalline Stoffe*, z.B. **Silizium (Si)** oder Germanium (Ge), deren elektrische Leitfähigkeit zwischen der von *Nichtleitern* (Isolatoren) und jener von den metallischen *Leitern* liegt.

Halbleitereigenschaften beruhen auf der Eigenschaft, dass Ladungsträger erst durch z.B. Wärme, elektromagnetische Strahlung (Licht) und durch gezielten Verunreinigungen (Dotierung mit Fremdatomen) aktiviert werden müssen, bevor diese zur Leitfähigkeit beitragen und als Bauelement entsprechend verwendet werden können.

Eigenleitung. Die Temperatur eines Kristalls entspricht mechanischen Schwingungen im Kristallgitter. Dadurch lösen sich ständig Elektronen aus dem Gitter. Sie hinterlassen ein positiv geladenes „Loch“ im Kristallgitter. Bei Annäherung eines freien Elektrons an ein Loch kann es aufgrund der elektrischen Anziehung zu einer *Rekombination* kommen. Bei Anlegen einer elektrischen Spannung fungieren die freien (thermischen) Elektronen als negative, die Löcher als positive Ladungsträger. Die Eigenleitung wird bei Bauelementen wie z.B. Heißleiter (NTC-Widerstand), Kaltleiter (PTC-Widerstand), Fotowiderständen (LDR) usw. ausgenutzt.

Dotierung. In Si und Ge können leicht Fremdatome der Elemente der III. Hauptgruppe (z.B. Bor) als Elektronenempfänger (Akzeptoren) und Atome der Elemente der V. Hauptgruppe (z.B. Phosphor) als Elektronenspender (Donatoren) eingebaut werden (durch die so genannte Dotierung).

Durch *gezielte „Verunreinigung“ mit Fremdatomen* (Dotierung) werden die elektrischen Eigenschaften gezielt verändert und die Halbleiter technisch nutzbar gemacht.

Fremdleitung: N-Halbleiter und P-Halbleiter

Im *reinen Halbleiter* ist die Zahl der Elektronen und Löcher gleich groß; im dotierten Kristall können negative Elektronen bzw. positive Löcher überwiegen: Es entsteht ein N-Halbleiter bzw. ein P-Halbleiter.

N-Leiter
P-Leiter

Grundphänomen PN-Übergang: Zwischen aneinander stoßenden N-Leitern und P-Leitern bildet sich durch *Rekombination* von Elektronen und Löchern eine nicht leitende **Sperrschicht** aus. Die Dicke dieser Sperrschicht kann nun durch eine angelegte Spannung **gesteuert** werden. Auf solchen PN-Übergängen (Bild 1.52) beruht die Funktionsweise von Dioden und auf NPN- bzw. PNP-Übergängen die Funktionsweise von Transistoren.

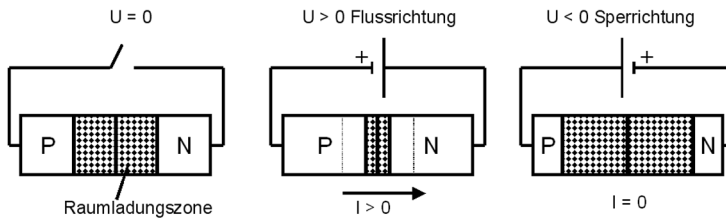


Bild 1.52 Aus der Funktion eines PN-Übergangs ergibt sich die Anwendung „Diode“

5.1.3 Die Basistechnologie: Transistor

Wichtigstes aktives Halbleiter-Bauelement und Bestandteil von integrierten Schaltungen ist der Transistor (aus engl. **transfer resistor** = „steuerbarer Widerstand“). Es wird unterschieden zwischen

- **bipolarem Transistor** (das ist „der“ klassische Transistor) und
- **unipolarem Transistor** (Feldeffekttransistor, FET).

Bipolarer Transistor. Dieser hat *drei Schichten* mit zwei Übergangsflächen: p-n-p oder n-p-n. Jede Schicht ist mit einer Leiterbahn kontaktiert, durch die Verbindungen zu benachbarten Bauelementen hergestellt werden. In Form der **Transistor-Transistor-Logik** (TTL, siehe Kapitel 5.3.2, Gattertechnologien) hat diese Art von Transistor weite Verbreitung gefunden.

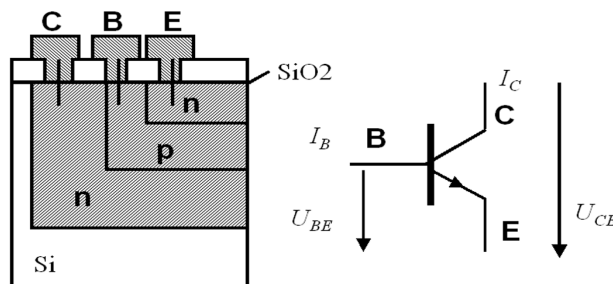


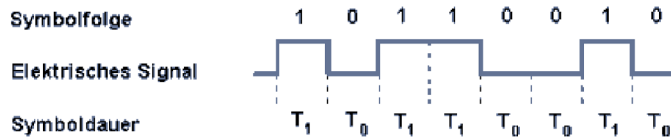
Bild 1.53 Prinzipieller vereinfachter Aufbau und Schaltzeichen eines Transistors, nur mit den wichtigsten Elementen; B = Basis, C = Kollektor, E = Emitter; U = Bezeichnung für Spannung, I = Bezeichnung für Strom, Si = Silizium, SiO₂ ist Siliziumdioxid

Unipolarer Transistor. In diesem wird der Strom in einem Kanal durch ein elektrisches Querfeld gesteuert, deshalb wird er als **Feldeffekttransistor** (FET) bezeichnet. Die **Metall-Oxid-Schicht-Technologie** (MOS) hat vor allem in CMOS-Bauelementen (**Complementary-MOS**, siehe Kapitel 5.3.2, Gattertechnologien) weite Verbreitung erfahren.

5.2 Schalter

In der Digitaltechnik werden die Gesetze der Schaltalgebra mit Hilfe von *elektronischen Verknüpfungsgliedern* realisiert. In (digitalen) Computern wird mit Digitalsignalen gearbeitet. Diese Digitalsignale haben rechteckförmigen Verlauf und werden von Taktimpulsen gesteuert:

Bild 1.54 Signalverlauf mit Taktsteuerung



Das Basiselement zur Erzeugung rechteckförmiger (digitaler) Signale ist der **Schalter**. Unabhängig vom physikalischen Aufbau (also ob der Schalter mechanisch oder elektronisch aufgebaut ist) ist ein Schalter ein Bauelement, das nur zwei Zustände annehmen kann: Schalter offen = „0“, Schalter geschlossen = „1“. Mit Hilfe von Transistoren lassen sich sehr gute Schalter realisieren.

Gatter haben zwei definierte (festgelegte) Schaltzustände: „Low“ („0“) und „High“ („1“)

5.3 Schaltelemente

Die Grundbestandteile von Schaltnetzen sind einzelne Logikelemente (Logikgatter, kurz: Gatter), die **logische Grundfunktionen** (UND-, ODER-, NICHT usw.) mit elementaren Transistorschaltungen realisieren.

5.3.1 Gatter

Ein Gatter (gate) ist ein elektronisches Bauelement mit n Eingängen (input) und m Ausgängen (output). Sowohl die Eingänge als auch die Ausgänge haben zwei definierte Schaltzustände: logisch „0“ bzw. „1“. Diese Werte sind je nach verwendeter Gattertechnologie (Kapitel 5.4.2) entsprechend definiert.

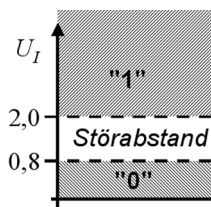


Bild 1.55 Beispiel für Logikpegel, U_I = Spannungspegel am Eingang (Input) in Volt

Beispielsweise ist in der TTL-Technologie (Transistor-Transistor-Logik) ein Signal am Gattereingang, dessen Wert sich zwischen 0 Volt und 0,8 Volt befindet, definiert als „0“ und ein Signal, das sich zwischen 2 Volt und der Betriebsspannung U_B von 5 Volt befindet, definiert als „1“ (Bild 1.55). Dazwischen befindet sich ein „undefinierter“ Bereich, der unbedingt vermieden werden muss (Störabstand, Bild 1.55).

Der logische Zustand der Gatter-Ausgänge bestimmt sich nach Regeln aus den Zuständen der Eingänge. Diese Regeln können mit Hilfe einer Wahrheitstabelle beschrieben werden.

Zu den Grundfunktionen zählen Inverter, AND, OR, EXOR, NAND und NOR. Aus den folgenden Bildern wird die Funktion der Grundgatter durch die Wahrheitstabelle ersichtlich:

x_1	x_2	NOT	AND	OR	NAND	NOR	XOR	ÄQUIV	NAME
0	0	1	0	0	1	1	0	1	Wertetabelle
0	1	1	0	1	1	0	1	0	
1	0	0	0	1	1	0	1	0	
1	1	0	1	1	0	0	0	1	
		\bar{x}_1	$x_1 \cdot x_2$	$x_1 + x_2$	$\overline{x_1 \cdot x_2}$	$\overline{x_1 + x_2}$	$x_1 \oplus x_2$	$x_1 \equiv x_2$	Funktion
									DIN (alt)
									DIN (neu)
									ASA

Bild 1.56 Die Grundgatter im Überblick; DIN = Deutsche Industrie Norm, ASA = Amerikanischer Standard

Gatter in Form einzelner ICs (Bild 1.57) werden in der Praxis nur noch bei einfachen oder speziellen Problemlösungen eingesetzt. Heute werden vorwiegend hoch integrierte Standardschaltkreise verwendet, wie zum Beispiel:

- PLD (Programmable Logic Device)
- EPLD (Erasable PLD)
- PEEL (Programmable Electrical Erasable Logic)
- FPGA (Free Programmable Gate Array)
- ASICS (Application Specific Integrated Circuit)

5.3.2 Gattertechnologien

Jedes Logikgatter kann als Baustein in unterschiedlichster Technologie (mit Vor- und Nachteilen) hergestellt werden (Bild 1.58):

PRIMARY CONCERN	SECONDARY CONCERN	5 V	3 V	2.5 V	1.8 V
HIGH SPEED	HIGH DRIVE	ABT, 74F	ALVT, LVT, ALVC	AVC, ALVC, ALVT	AUC
	LOW NOISE	ABT, 74F	ALVC, LVT, LVC	AVC	AUC
	LOW POWER	ABT, AC/ACT	ALVC, LVT, LVC	AVC	AUC
HIGH DRIVE	HIGH SPEED	ABT, 74F	ALVT, LVT, ALVC	AVC, ALVC, ALVT	AUC
	LOW NOISE	ABT, 74F	LVT	AVC	AUC
	LOW POWER	ABT	LVT	AVC	AUC
LOW NOISE	HIGH SPEED	ABT, AHC	ALVC, LVT, LVC, LV	AVC	AUC
	HIGH DRIVE	ABT, 74F	LVT	AVC	AUC
	LOW POWER	AHC, ABT	ALVC, LVT, LVC, LV, AHC	AVC	AUC
LOW POWER	HIGH SPEED	ABT, AHC	LVT, ALVC	AVC	AUC
	HIGH DRIVE	ABT	ALVC, ALVT, LVT, LVC	AVC	AUC
	LOW NOISE	AHC, ABT	ALVC, LVT, LVC, LV	AVC	AUC

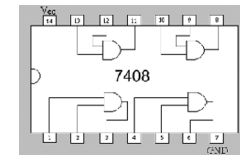


Bild 1.57 Beispiel für ein Logikgatter: ein vierfach UND-Baustein; Achtung: Die Anschlüsse von ICs werden zum Unterschied von diskreten Bauelementen (z.B. Einzeltransistoren) von oben (Draufsicht) bestimmt [W15]

Bild 1.58 Die zentrale Frage eines IT-Entwicklers lautet: Welche Gattertechnologie setze ich ein? Dieses Bild zeigt eine Auswahl von „Logikfamilien“, ausgehend von ihren typischen Eigenschaften, Erklärung der Abkürzungen siehe nächste Seite [W16]

IT 1 Grundkonzepte

- ABT**
- Advanced BiCMOS**
- AC/ACT**
- Advanced CMOS**
- AHC/AHCT**
- Advanced High-Speed CMOS**
- ALVC**
- Advanced Low-Voltage CMOS**
- ALVT**
- Advanced Low Voltage BiCMOS**
- AUC**
- Advanced Ultra-Low Voltage CMOS**
- AVC**
- Advanced Very Low Voltage CMOS**
- LVT**
- Low Voltage BiCMOS**
- 74F**
- Fast Logic**

Die verbreitetsten Technologien sind:

- Bipolar,
- CMOS und
- BiCMOS.

Bipolar umfasst als bekannte Vertreter die TTL-Technologie (Transistor-Transistor-Logik), STTL (Schottky-TTL), AS (Advanced Schottky-TTL) und LSTTL (Low-Power-Schottky-TTL). Diese Technologie hat folgende Vorteile: hohe Geschwindigkeit, hohe Treiberleistung (daher geeignet für Bustreiber) und gute Spannungsfestigkeit gegenüber statischer Aufladung. Als Nachteil wäre die relativ hohe Leistungsaufnahme zu erwähnen.

CMOS (Complementary Metal Oxide Semiconductor) hat u.a. folgende Vorteile: niedrige Leistungsaufnahme, sehr gute Skalierbarkeit, hohe Geschwindigkeit und hohe Packungsdichte. Als Nachteil wäre die Empfindlichkeit gegenüber statischen Aufladungen zu erwähnen.

BiCMOS ist eine Kombination der Bipolar- und CMOS-Technologie.

Es existieren unzählige spezielle Technologien, doch sei hier auf die Literatur verwiesen.

5.4 Kombinatorische Logik: Schaltnetze

Ein **Schaltnetz** ist eine Anordnung von Logikgattern (Digitalbausteinen), die Boole'sche Variablen (Signale) derart verknüpft, dass die Signale an den Ausgängen *zu jedem Zeitpunkt ausschließlich* von den logischen Werten der Schaltvariablen an den Eingängen abhängen. Schaltnetze sind rückführungsfrei, d.h., Rückkopplungen von Ausgängen einer Verknüpfungsstufe auf Eingänge der gleichen oder einer vorhergehenden Stufe kommen nicht vor.



Bild 1.59 Blockschaltbild eines Schaltnetzes: die outputs ($o_1 \dots o_m$) sind nur von den inputs ($i_1 \dots i_n$) abhängig

Schaltnetze sind nur zur Realisierung von *kombinatorischen* Verknüpfungen geeignet: Es können logische Operationen und Schaltungen gebaut werden.

Beispiele: Codierer, Multiplexer usw.

Schaltnetze sind *nicht* in der Lage, Informationen zu speichern, und können daher *keine sequentiellen Algorithmen* abarbeiten.

5.5 Sequentielle Logik: Schaltwerke

Schaltwerke sind Logikschaltungen, die Datenwörter speichern können und deren Ausgangsvariablen nicht mehr nur kombinatorisch aus den aktuellen Eingangsvariablen entstehen. Die Ausgangsvariablen hängen auch von Eingangsvariablen früherer Zeitpunkte ab und sind das Ergebnis eines algorithmischen Ablaufes. Schaltwerke können Schaltzustände **speichern** und **sequentielle Abläufe steuern** (Bild 1.60):

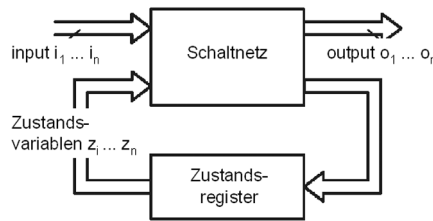


Bild 1.60 Blockschaltbild eines Schaltwerks: die outputs ($o_1 \dots o_m$) sind vom Zustand des Schaltwerks ($z_1 \dots z_n$) und von den inputs ($i_1 \dots i_n$) abhängig!

Ein Schaltwerk zeigt Speicherverhalten, es besitzt einen „Zustand“. Durch Eingangssignale ändert sich dieser Zustand.

Der Ausgang eines Schaltwerks hängt im Unterschied zum Schaltnetz nicht nur vom Eingang, sondern zusätzlich von der „Vorgeschichte“ (z.B. frühere Eingangssignale) ab.

Technisch werden Schaltwerke als „**Flipflops (FF)**“ ausgeführt (Bild 1.61):

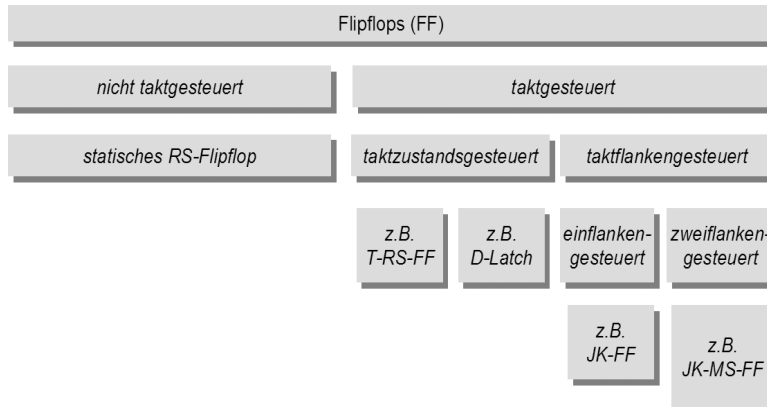


Bild 1.61 Arten von Flipflops

Direkt gesteuertes (nicht taktgesteuertes) Flipflop: *Nicht taktgesteuert* bedeutet, dass eine anliegende Eingangsinformation sofort gespeichert wird, d.h., sie wirkt sich unmittelbar aus (z.B. statisches RS-Flipflop).

IT 1 Grundkonzepte

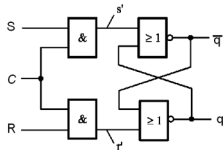


Bild 1.62 So wird ein taktgesteuertes Flipflop aus Grundgattern (siehe Bild 1.56) aufgebaut ...

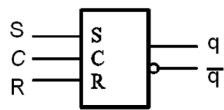


Bild 1.63 ... und so erkennt man ein solches Flipflop im Schaltplan

Taktgesteuerte Flipflops: Manche Anwendungen verlangen, dass ein Flipflop nur zu einem bestimmten Zeitpunkt auf den Eingangszustand reagiert. Diese Forderung hat zur Entwicklung der taktgesteuerten Flipflops geführt. Dabei bestimmt ein zusätzlicher Eingang C (Clock, Bild 1.62) den Zeitpunkt der Datenübernahme.

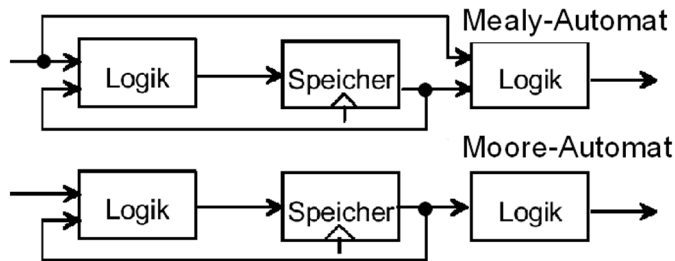
Die taktgesteuerten Flipflops werden in „taktzustandsgesteuert“ und in „taktflankengesteuert“ eingeteilt. Taktzustandsgesteuert bedeutet, dass erst der Pegel am C-Eingang die Datenübernahme ermöglicht, z.B. bei L-Pegel passiert nichts, bei H-Pegel erfolgt die Datenübernahme. Bei Taktflankensteuerung passiert nur etwas *während* ansteigender oder abfallender Flanke.

Bei einem nach MOORE (1956) benannten **Moore-Automaten** ist das Ergebnis Z nur vom Zustandsspeicher abhängig, der zu bestimmten Zeiten den Inhalt wechselt. Das Ergebnis ist daher auch nur zu bestimmten Zeiten veränderbar. Der Automat ist streng getaktet, denn das Ergebnis wechselt nur zu den Taktflanken.

Das Ergebnis Z eines nach MEALY (1955) benannten **Mealy-Automaten** ist vom Zustandsspeicher und der Eingabe X abhängig. Da die Eingabe zu jedem beliebigen Zeitpunkt wechseln kann (sie ist nicht getaktet), trifft dies auch auf das Ergebnis zu!

Der prinzipielle Unterschied zwischen den Ideen von MOORE und MEALY wird aus folgendem Bild ersichtlich:

Bild 1.64 Moore-Automaten und Mealy-Automaten sind getaktete synchrone Zustandsmaschinen und unterscheiden sich durch ihre Logik am Eingang



5.6 Rechenwerke

Rechenwerke sind die Vorstufe zum Aufbau von Computersystemen. Es werden die folgenden vier Schaltungen unterschieden:

- Register,
- Halbaddierer (HA),
- Volladdierer (VA) und
- Akkumulator.

5.6.1 Register

Register dienen zum Speichern von logisch zusammengehörenden Bits (meistens 8 Bit = 1 Byte).

Register sind aus einzelnen Flipflops aufgebaut, die untereinander nicht verbunden sind.

Bild 1.65 zeigt ein **Byte-Register**. Das Einspeichern erfolgt durch Anlegen der Daten an die Leitungen S_0 bis S_7 . Das Löschen erfolgt jeweils durch R_0 bis R_7 . Die einzelnen Teile von Computersystemen (wie z.B. Rechenwerk, Steuerwerk, Speicherwerk (siehe Modul 2) enthalten alle mehrere Register zur **kurzzeitigen Speicherung** von Daten (Bild 1.65).

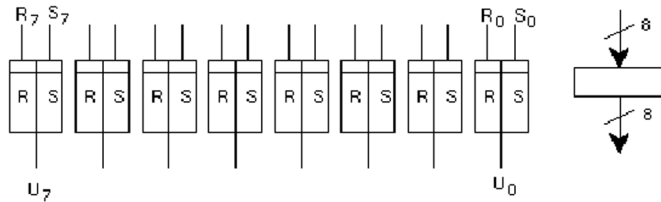


Bild 1.65 Ein Byte-Register, aus einzelnen Flipflops aufgebaut

5.6.2 Halbaddierer

Addierer werden als Rechenglieder zur Addition von Dualzahlen verwendet. Die Subtraktion von Dualzahlen wird auf die Addition zurückgeführt. Ein **Halbaddierer (HA)** addiert zwei 1-Bit-Zahlen **ohne Eingangsübertrag** (carry in).

Aus Bild 1.66 geht die Funktion hervor. Die Addition $x+y$ mit dem Ergebnis z und dem Übertrag \bar{u} wird durch die Wertetabelle beschrieben (Bild 1.66):

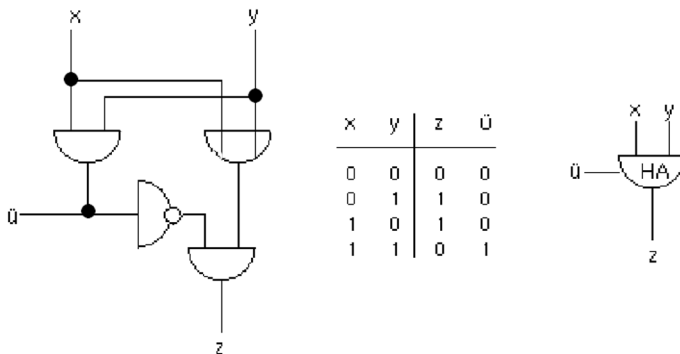
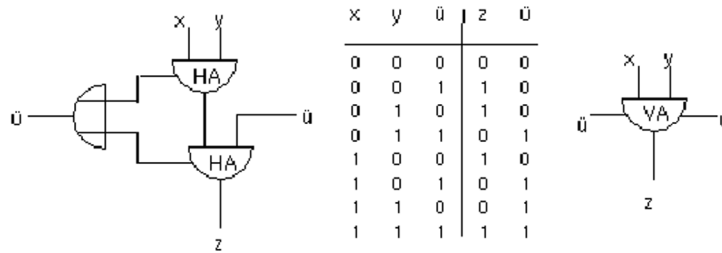


Bild 1.66 Schaltung, Wahrheitstabelle und Schaltzeichen eines Halbaddierers

5.6.3 Volladdierer

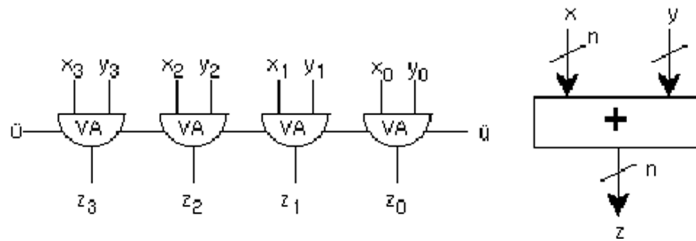
Zur (vollständigen) Addition von zwei einstelligen Dualzahlen muss der **Übertrag** \ddot{u} aus der *vorhergehenden Stelle* hinzugefügt werden. Dazu werden zwei Halbaddierer (HA) zu einem Volladdierer (VA) vereinigt (Bild 1.67):

Bild 1.67 Schaltung, Wahrheitstabelle und Schaltzeichen eines Volladdierers



Sollen zwei n -stellige Dualzahlen addiert werden, so werden n Volladdierer benötigt, die in einer Kette geschaltet werden (Bild 1.68):

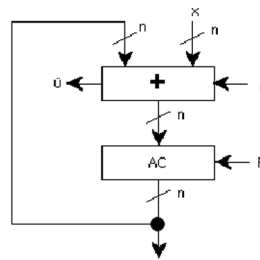
Bild 1.68 Schaltung einer n -stelligen Addierkette



5.6.4 Akkumulator

Ein Akkumulator ist eine Kombination aus Addierkette und Register. Diese „sammelt“ (lat. accumulare = „anhäufen“) die Ergebnisse fortlaufender Additionen und stellt damit den Kern eines Rechenwerks dar (Bild 1.69):

Bild 1.69 Das Prinzipschaltbild eines Akkumulators



Alle Operationen können auf eine Folge von Additionen zurückgeführt werden. Die Zeit, die zur Ausführung eines Additionsschrittes benötigt wird, heißt Additionszeit, wird durch die Taktfrequenz bestimmt und stellt eine wichtige Kenngröße für die Rechengeschwindigkeit eines Computers dar: Bei modernen Mikroprozessoren liegt sie unter 10 ns.

6 Modulkurzzusammenfassung

Signale sind physikalische Träger für Daten, die als Träger einer Nachricht und Information wirken. Ein **analoges Signal** bildet einen kontinuierlichen Vorgang kontinuierlich ab. Ein **digitales Signal** besteht nur aus (codierten) diskreten Zeichen. Daten sind codierte Informationen, die in Computersystemen gespeichert sind. Ein **Datentyp** ist die Festlegung der Interpretation einer gespeicherten (physikalischen) Bitfolge. Datentypen können einfacher Art sein (ganze oder reelle Zahlen, logische Werte, Zeichen) oder zusammengesetzt bzw. strukturiert sein (Felder, Zeichenfolgen, Strukturen usw.). Zu den elementaren Datentypen gehören: Bit (Byte), Zahlen (Integer, Real, Boolean), Zeichen, String, Wort. Zu den strukturierten Datentypen zählen: Mengen, Arrays, Listen, Matrizen, Tabellen, Relationen, Bäume, Graphen, Files, Programme, Objekte, Klassen, Methoden. Welche Information einer Nachricht zugeschrieben wird, ist vom Empfänger dieser Nachricht abhängig und daher subjektiv. Information ist eine raum- und/oder zeitvariante (veränderliche) Folge physikalischer Signale, bestehend aus dem physikalisch-materiellen Träger und dem Informationsinhalt. Digitale Information ist *alles, was als Folge von „0“ und „1“ (binär) darstellbar ist*. **Digitalisierung** ist damit die Überführung beliebiger Daten in eine solche binäre Darstellung. Denken und Wissen sind untrennbar mit dem menschlichen Geist verbunden: **Wissen** ist (im menschlichen Gedächtnis) organisierte, verfügbare und anwendbare Information. **Codierung** ist eine Vorschrift für die eindeutige Zuordnung eines Zeichenvorrats zu dem eines anderen Zeichenvorrats. Ein Informationssystem unterliegt den Gesetzen der Nachrichtentechnik und kann durch das *Kommunikationsmodell* von SHANNON & WEAVER dargestellt werden. Ein bit ist die kleinste binäre Informationsmenge und beschreibt den logischen Zustand eines zweiwertigen Systems. Die Kanalkapazität (channel capacity) ist der maximale Informationsfluss, der über einen Nachrichtenkanal fehlerfrei übertragen werden kann. Ein **Zahlensystem** ist die Gesamtheit der zur Darstellung einer Zahl verwendeten *Zahlzeichen (Ziffern)* und enthält die *Regeln* für deren Zusammensetzung. Für die IT/Informatik sind aber insbesondere folgende Zahlensysteme wichtig: Hexadezimal-, Oktal- und Dualsystem. Diese Algebra wird inzwischen als „Boole’sche Algebra“ bezeichnet. Ihre große Bedeutung beruht auf ihren vielfältigen Anwendungsmöglichkeiten: Sie eignet sich zur Behandlung von Problemen der Mengenalgebra sowie zur algebraischen Darstellung der Aussagenlogik und zur algebraischen Beschreibung und Manipulation von digitalen Schaltkreisen. **Computer** sind Maschinen zur Ausführung von Instruktionen. Im Gegensatz zu Rechenmaschinen handelt es sich aber um eine transklassische Maschine – eine *universelle Maschine*, deren Verwendungszweck nicht durch die gegebene Hardware vorbestimmt wird. Computerteile sind durch *integrierte Schaltkreise* (ICs) realisiert. Diese enthalten **Schaltnetze** (Logikgatter) und **Schaltwerke** (Flipflops). Die Basistechnologie ist der **Transistor**.

7 Modulanhang

7.1 Literatur

7.1.1 Bücher

- BEUTH, KLAUS (1998): *Elektronik 4. Digitaltechnik*. 10. Auflage. Würzburg: Vogel.
- BEUTH, KLAUS; BEUTH, OLAF (2000): *Elektronik 1. Elementare Elektronik. Mit Grundlagen der Elektrotechnik*. 6. Auflage. Würzburg: Vogel.
- BORUCKI, LORENZ (1977): *Digitaltechnik*. Stuttgart: Teubner.
- FRIEDRICH, BERND (1995): *Kanalcodierung. Grundlagen und Anwendungen in modernen Kommunikationssystemen (Information und Kommunikation)*. Berlin u.a.: Springer.
- GOLDSCHLAGER, LES; LISTER, ANDREW (1990): *Informatik: Eine moderne Einführung*. 3. Auflage. München, Wien u.a.: Hanser Prentice Hall International.
- HAKEN, HERMANN; HAKEN- KRELL, MARIA (1989): *Entstehung von biologischer Information und Ordnung (Dimensionen der modernen Biologie Bd. 3)*. Darmstadt: Wissenschaftliche Buchgesellschaft.
- KLIMANT, H.; PIOTRASCHKA, D.; SCHÖNFELD, D. (1996): *Informations- und Codierungstheorie*. Stuttgart: Teubner.
- LIPP, HANS MARTIN (2002): *Grundlagen der Digitaltechnik*. 3. Auflage. München: Oldenbourg.
- MILDENBERGER, OTTO (1999): *Informationstechnik kompakt. Theoretische Grundlagen*. Wiesbaden: Vieweg.
- MÜLLER, RUDOLF (1995): *Grundlagen der Halbleiter-Elektronik (Halbleiter-Elektronik Bd. 1)*. 7. Auflage. Berlin u.a.: Springer.
- MÜLLER, RUDOLF (1991): *Grundlagen der Halbleiter-Elektronik (Halbleiter-Elektronik Bd. 2)*. 4. Auflage. Berlin u.a.: Springer.
- NEGROPONTE, NICHOLAS (1997): *Total digital. Die Welt zwischen 0 und 1 oder Die Zukunft der Kommunikation*. Goldmann.
- PAUL, REINHOLD (1993): *Elektronische Halbleiterbauelemente (Teubner-Studienskripten; 112)*. 3. Auflage. Stuttgart: Teubner.
- POLANYI, M. (1985): *Implizites Wissen*. Frankfurt/Main: Suhrkamp.
- ROHLING, HERMANN (1995): *Einführung in die Informations- und Codierungstheorie*. Stuttgart: Teubner.

SCARBATA, GERD (1996): *Synthese und Analyse digitaler Schaltungen*. München: Oldenbourg.

SCHIFFMANN, WOLFRAM; SCHMITZ, ROBERT (2001): *Technische Informatik 1: Grundlagen der digitalen Elektronik (Springer-Lehrbuch)*. 4. Auflage. Berlin u.a. Springer.

SIEFKES, DIRK; EULENHÖFER, PETER; STACH, HEIKE HRSG. (1998): *Sozialgeschichte der Informatik*. Wiesbaden: Deutscher Universitätsverlag.

SIEFKES, DIRK; BRAUN, ANETTE; EULENHÖFER, PETER; STACH, HEIKE; STÄDTLER, KLAUS (1999): *Pioniere der Informatik. Ihre Lebensgeschichte im Interview. Interviews mit Bauer, Floyd, Weizenbaum, Wirth und Zemanek*. Berlin u.a.: Springer.

SCHÖNING, UWE (1992): *Logik für Informatiker*. Mannheim: B.I. Wissenschaftsverlag.

TIETZE, ULRICH; SCHENK, CHRISTOPH (2002): *Halbleiter-Schaltungstechnik*. 12. Auflage. Berlin u.a.: Springer.

URBANSKI, K.; WOITOWITZ, R. (2000): *Digitaltechnik. Ein Lehr- und Übungsbuch (Springer-Lehrbuch)*. 3. Auflage. Berlin u.a.: Springer.

VÖLKLEIN, FRIEDEMANN; ZETTERER, THOMAS (2001): *Einführung in die Mikrosystemtechnik. Grundlagen und Praxisbeispiele*. Wiesbaden: Vieweg.

WEIZENBAUM, JOSEPH (2001): *Die Macht der Computer und die Ohnmacht der Vernunft*. 11. Auflage. Frankfurt/Main: Suhrkamp.

WERNER, M. (2000): *Signale und Systeme: Ein Lehr- und Arbeitsbuch*. Wiesbaden: Vieweg.

ZEMANEK, HEINZ (1992): *Das geistige Umfeld der Informationstechnik*. Berlin u.a.: Edition SEL-Stiftung, Springer.

ZEMANEK, HEINZ (2001): *Vom Mailüfterl zum Internet. Geschichte, Perspektiven und Kritik der Informationstechnik*. Wien: Picus-Verlag.

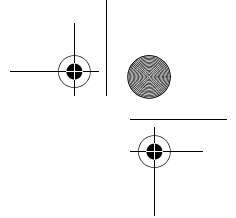
7.1.2 Artikel

COY, WOLFGANG (1995): Automat – Werkzeug – Medium. *Informatik Spektrum*, 18, 1, 31–38.

COY, WOLFGANG (1994): Computerkultur – Schriftkultur. Computer und Buch im Wettbewerb. *Forschung & Lehre*, 9/94, 376 (1994).

COY, WOLFGANG (1994): Gutenberg und Turing. Fünf Thesen zur Geburt der Hypermedien. *Zeitschrift für Semiotik*, 16, 69–74 (1994).

MAURER, HERMANN (2000): Prognosen und Thesen nicht nur zum Schmunzeln. *Informatik Spektrum*, 23, Februar 2000, 51–59.



7.1.3 Books in English

CLEMENTS, ALAN (2000): *The Principles of Computer Hardware*. Oxford: Oxford University Press.

COVER, THOMAS M.; THOMAS, JOY A. (1991): *Elements in Information Theory*. New York: Wiley.

HAKEN, HERMANN (1988): *Information and Self-Organization*. Berlin u.a.: Springer.

HOROWITZ, PAUL; HILL, WINFIELD (1989): *The Art of Electronics*. 2nd Edition. Cambridge: Cambridge University Press.

KHINCHIN, ALEXANDER I. (1957): *Mathematical Foundations of Information Theory*. Dover Publications.

MENDELSON, ELLIOTT (1970): *Schaum's Outline of Boolean Algebra and Switching Circuits*. McGraw Hill.

SHANNON, CLAUDE E.; WEAVER, WARREN (1963): *The Mathematical Theory of Communication*. 2nd Ed. Urbana, Chicago, London: University of Illinois Press.

SKLAR, BERNHARD (1988): *Digital Communications: Fundamentals and Applications*. London: Prentice Hall.

STONIER, TOM (1997): *Information and Meaning. An Evolutionary Perspective*. Berlin et.al.: Springer.

TANENBAUM, ANDREW S. (1998): *Structured Computer Organization*. 4th Edition. London: Prentice Hall.

WAKERLY, J. (1990): *Digital Design: Principles and Practices*. London: Prentice-Hall.

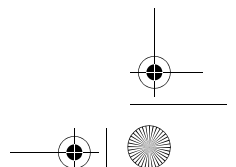
7.1.4 Articles in English

HOFKIRCHNER, WOLFGANG (1999): Cognitive Sciences In the Perspective of a Unified Theory of Information. In: ALLEN, J. K., HALL, M. L. W., WILBY, J. (Eds.), *Proceedings of the 43rd Conference of the Int. Society for the Systems Sciences: www.iss.org/1999meet*

MOORE, GORDON (1965): Cramming more components onto integrated circuits. *Electronics*, Vol. 38, No. 8, 19th April 1965.

MOORE EDWARD F. (1956): Gedanken-Experiments on Sequential Machines. *Automata Studies*, Princeton University Press, 129-153.

MEALY, GEORGE H. (1955): A Method for Synthesizing Sequential Circuits. *Bell System Technical Journal*, Vol. 34, pp. 1045-1079.



MEALY, GEORGE H. (1962): Programming systems, whither? *Communications of the ACM*, July 1962, Volume 5 Issue 7, 374–375.

SHANNON, CLAUDE ELWOOD (1948): A mathematical theory of communication. *Bell System Technical Journal*, vol. 27, July and October, 1948, 379–423 and 623–656.

WANT, R., BORRIELLO, G. (2000): Special Issue on Information Appliances. *IEEE Computer Graphics and Applications*, May/June 2000.

WEISER, M. (1991): The Computer for the 21st Century. *Scientific American*, September 1991, 66–75.

7.1.5 Journals

Computers & Electrical Engineering (ISSN: 0045-7906) | Elsevier

Information and Software Technology (ISSN: 0950-5849) | Elsevier

Telematics and Informatics (ISSN: 0736-5853) | Elsevier

Signal Processing (ISSN: 0165-1684) | Elsevier

7.2 Internet-Links

Aktualisierte Internet-Links zu diesem Modul sind auf der Buchhomepage www.basiswissen-it.at unter IT – Modul 1: Grundkonzepte verfügbar!

7.3 Prüfungsfragen

Fragentyp 1: Dichotome Ja/Nein Entscheidungen:

01	Ein Digitalsignal wird durch Abtastung erhalten und ist zeitkontinuierlich und wertdiskret.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
02	Der Datentyp „String“ ist eine zweidimensionale geordnete Struktur und besteht aus n Zeilen und m Spalten.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
03	Ein Objekt ist eine ausführbare Einheit, hat einen Zustand, verfügt über Algorithmen und kann Nachrichten versenden bzw. empfangen.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
04	Der ASCII-Code bildet die Grundlage der Codierung von Standardzeichen in Personalcomputern (PCs).	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
05	Die Kanalkapazität hängt nur vom Übertragungskanal ab und ist unabhängig von den Eigenschaften der Informationsquelle.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
06	Das Dualsystem ist ein polyadisches Zahlensystem mit der Basis 2 mit nur zwei zulässigen Ziffern: „0“ und „1“.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
07	Die Eigenschaften von Serien- und Parallelschaltungen mechanischer Schalter können durch die zweiwertige Logik gut dargestellt werden.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
08	Die Mikro-Architektur-Ebene ist die Ebene der digitalen Logik, die Gatter sind die kleinsten logischen Elemente (Digitalbausteine).	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
09	Die CMOS-Technologie ist relativ unempfindlich gegenüber statischer Aufladung und zeichnet sich durch hohe Geschwindigkeit aus.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein
10	Taktzustandsgesteuert bedeutet, dass erst ein entsprechender Pegel am C-Eingang (Clock) die Datenübernahme ermöglicht.	<input type="checkbox"/> Ja <input type="checkbox"/> Nein

IT 1 Grundkonzepte

Fragen-Typ 2: Mehrfachauswahlantworten (Multiple Choice):

01	Ein „reines“ Digitalsignal ist ... <input type="checkbox"/> a) ... wertdiskret und zeitkontinuierlich. <input type="checkbox"/> b) ... wertdiskret und zeitdiskret. <input type="checkbox"/> c) ... ein binär codiertes Abtastsignal. <input type="checkbox"/> d) ... ein Signal, das durch Abtastung, Quantisierung und Codierung entsteht.
02	Zu den elementaren Datentypen zählen ... <input type="checkbox"/> a) ... Graphen. <input type="checkbox"/> b) ... Byte. <input type="checkbox"/> c) ... Zeichen. <input type="checkbox"/> d) ... Mengen.
03	Im informationstheoretischen Sinn ist eine Nachricht ... <input type="checkbox"/> a) ... eine Folge von Zeichen auf einem Übertragungskanal und stets subjektiv. <input type="checkbox"/> b) ... prinzipiell das Gleiche wie eine Information. <input type="checkbox"/> c) ... eine Teilmenge von Wissen, die in Handlungssituationen benötigt wird. <input type="checkbox"/> d) ... ein Oberbegriff von Information. Erst die Information hat Bedeutung.
04	Zu den numerischen Codes zählen ... <input type="checkbox"/> a) ... ASCII-Code. <input type="checkbox"/> b) ... ANSI-Code. <input type="checkbox"/> c) ... BCD-Code. <input type="checkbox"/> d) ... EBCDI-Code.
05	Der Begriff „bit“ ... <input type="checkbox"/> a) ... ist die kleinste binäre Informationsmenge. <input type="checkbox"/> b) ... beschreibt den logischen Zustand eines Systems. <input type="checkbox"/> c) ... ist groß geschrieben „Bit“ der Name für eine binäre Stelle oder Ziffer. <input type="checkbox"/> d) ... ist die Einheit für den Informationsgehalt und kann Werte < 0 annehmen.
06	Boole'sche Funktionen ... <input type="checkbox"/> a) ... sind das gleiche wie Binäroperationen. <input type="checkbox"/> b) ... können durch entsprechende Formelzeichen dargestellt werden. <input type="checkbox"/> c) ... können durch Venn-Diagramme visualisiert werden. <input type="checkbox"/> d) ... sind gleich wie Binärfunktionen.
07	Ein Schaltnetz ... <input type="checkbox"/> a) ... ist definiert als Logikschaltung, die Datenwörter speichern kann. <input type="checkbox"/> b) ... ist zur Realisierung von kombinatorischen Verknüpfungen geeignet. <input type="checkbox"/> c) ... kann auch aus Flip-Flops bestehen. <input type="checkbox"/> d) ... ist taktgesteuert.
08	Ein Register ... <input type="checkbox"/> a) ... wird zur Addition von Dualzahlen verwendet. <input type="checkbox"/> b) ... ist aus Flip-Flops aufgebaut, die untereinander nicht verbunden sind. <input type="checkbox"/> c) ... ist eine Kombination aus Volladdierern. <input type="checkbox"/> d) ... dient nur zur kurzzeitigen Speicherung von Informationen.

7.4 Lösungen

Lösungen zu Fragen-Typ 1: 01 Nein; 02 Nein; 03 Ja; 04 Ja; 05 Ja; 06 Ja; 07 Ja; 08 Nein; 09 Nein; 10 Ja;

Lösungen zu Fragen-Typ 2: Richtig sind: 01 b) c) d); 02 b) c); 03 a) d); 04 c); 05 a) b) c) d); 06 b) c); 07 b); 08 b) d)

7.5 Hands-On: Übungen

- Schauen Sie sich um! Finden Sie heraus, welche Geräte „analog“ und welche „digital“ arbeiten. Sehen Sie in den Geräteunterlagen nach. Überlegen Sie, wo es um Nachricht, Information und/oder Daten geht!
- Zeigen Sie, dass Sie die Umwandlungsverfahren für Zahlen beliebiger Stellenwertsysteme beherrschen (z.B. 15 DEC in BIN). Kontrollieren Sie die Ergebnisse mit Hilfe des Windows-Rechners (Bild 1.70).
- Bauen Sie mit Hilfe eines 7408-Gatters (Kapitel 5.3.1) eine logische Verknüpfung auf. Bedenken Sie stets, dass die elementare Digitaltechnik die Basis der heutigen IT darstellt!



Bild 1.70 Unter Datei – Programme – Zubehör befindet sich ein Rechner; in der Ansicht: wissenschaftlich können Umwandlungen durchgeführt werden

7.6 Diskussionsfragen

- Diskutieren Sie über die Zukunft der Informationstechnik. Wird sich die Digitaltechnik weiter durchsetzen? Welche Aufgaben werden noch „analog“ gelöst? Welche können noch nicht „digital“ gelöst werden?
- Diskutieren Sie über die Bandbreitenproblematik. Warum wird überhaupt Bandbreite (Kanalkapazität) benötigt? Schlagen Sie Ideen zur Erhöhung der Bandbreite verschiedener Systeme vor: alles ist erlaubt.
- Jemand übernimmt die Rolle von GEORGE BOOLE und stellt „seine“ Algebra vor. Die Gruppe soll kritische Fragen stellen. Jemand in der Rolle von GORDON MOORE soll anknüpfen und die Idee von Chips erklären.

7.7 Timeline: Grundkonzepte der IT

30000 v. Chr. Verwendung von primitiven Zahlzeichen (Kerben, Knoten, Steinchen, Striche usw.) zum Abzählen.

3100 v. Chr. Die Sumerer verwenden einfache gegenständliche Abbildungen auf Tontafeln, um Geschäfte zu dokumentieren: Entstehung der Schrift.

500 Begründung des dezimalen Zahlensystems in Arabien.

1450 GUTENBERG: Beginn des Informationszeitalters.

1520 ADAM RIES veröffentlicht Rechenbücher.

1600 JOHN NEPER entwickelt Rechenstäbe, mit deren Hilfe die Multiplikation auf die Addition von Teilprodukten zurückgeführt wurde. Dieses Prinzip gelangt später in SCHICKARDS Rechenmaschine zur Anwendung (siehe Modul 2).

1620 Rechenschieber mit logarithmischer Skala.

1844 SAMUEL MORSE entwickelt den Morsecode zur Telegrafenkommunikation.

1854 BOOLE formalisiert die Aussagenlogik als Algebra.

1874 FERDINAND BRAUN entdeckt die Gleichrichterwirkung eines Punktkontaktes von Metall auf Halbleiter (Diode).

1926 Patent eines Feldeffekttransistors (FET) von LILIENFELD; allerdings wird der



erste FET erst gegen Anfang 1960 industriell gefertigt.

1945 Die Visionen von VANNEVAR BUSH in seinem MEMEX (memory expander) sind bahnbrechend für die IT: Er skizziert eine Maschine, mit der Informationen „auf Knopfdruck“ zur Verfügung stehen.

1947 J. BARDEEN, W. H. BRATTAIN und W. SHOCKLEY entdecken in den Bell Telephone Laboratories den Transistoreffekt. Er wirkt wie Relais und Röhre auch als Schalter, ist aber kleiner, schneller und haltbarer. Gründung der ersten Vereinigung von Computerfachleuten: Association for Computing Machinery (ACM) in USA.

1958 JACK KILBY baut die erste integrierte Schaltung (IC).

1963 Der 7 Bit USASCII (US Standard Code for Information Interchange) wird als Norm verabschiedet und löst damit das internationale Telegraphenalphabet Nr. 2 ab, das nur 5 Bit hat (mit Umschaltzeichen 64 Zeichen).

1993 Multipurpose Internet Mail Extensions (MIME) wird als Standard festgelegt. Er ist für die Übertragung aller nicht-ASCII-Dateien definiert, zum Beispiel Bilder, Töne und Binärdateien.

1996 TANENBAUM verwendet ein Schichtenmodell, das das Zusammenwirken von IT (Hardware) und Informatik (Software) anschaulich zeigt.

7.8 Glossar

Analoges Signal ist ein Signal, dessen Informationsparameter innerhalb festgelegter Grenzen jeden beliebigen Wert annehmen kann.

Binäres Signal ist ein diskretes Signal, dessen Informationsparameter genau zwei Werte annehmen kann. Entsprechend dem binären Zahlensystem (Dualsystem) werden die beiden Werte als Signalwert 0 und Signalwert 1 bezeichnet (Binärzeichen).

Binärzeichen (binary character, Bit) ist jedes Zeichen aus einem Zeichenvorrat von genau zwei Zeichen.

Diskretes Signal ist ein Signal, dessen Informationsparameter nur bestimmte (diskrete) Werte einer endlichen Menge annehmen kann.

Digitales Signal ist ein diskretes Signal, bei dem den Werten der Informationsparameter Zeichen (Worte) entsprechen. Die Wortzuordnung wird als Codierung bezeichnet.

Signal ist die Darstellung von Informationen durch physikalische Größen.

Signalparameter ist die Kenngröße eines Signals, dessen Wert die Information beinhaltet. Der Informationsparameter einer Wechselspannung als Signalträger kann z.B. die Amplitude, die Frequenz oder der Phasenwinkel sein.

Zeichen (Character) ist ein Element aus einer zur Darstellung von Informationen vereinbarten endlichen Menge von Objekten.

