

Grundlagen der EDV

Vorlesung mit Übungen

Sommersemester 2018

www.solarflyer.de

Dipl. Ing. Martin Ney
E-Mail: m.ney@luxea.de



Grundlagen der EDV

Überblick über den Vorlesungsinhalt

- Geschichte der EDV, Entwicklung bis heute
- Zahlensysteme, Zeichensysteme, Datenaustausch
- Hardware, Speicher, Schnittstellen, Spezifikationen
- Betriebssysteme und ihre Aufgaben
- Grafikformate, Drucker, Bildschirm
- Internetstandards, Dateiformate, HTML, Sicherheit
- Datenbanken, SQL, Applikationen wie CMS, CRM
- Übungen: Word, Excel, Access, Tools, HTML...



Grundlagen der EDV

Ziele der Vorlesung

- Überblick über die Thematik
- Grundlegende Konzepte verstehen
- Wichtige Begriffe und Abkürzungen kennenlernen
- Zunehmende Relevanz im Alltagsleben
- Sicherheit von Systemen und im Internet
- Unterstützende Fertigkeiten beim Arbeiten mit Computern und Standardprogrammen z.B. Office
- Kenntnisse beim Einkauf von EDV Produkten



Grundlagen der EDV

Beispiele

- **Alexa bestellt aus versehen viele Puppenhäuser**
<https://www.heise.de/newsticker/meldung/Amazon-Echo-Nachrichtensprecher-loest-Massenbestellung-aus-3591039.html>
- **Apple 1997 kurz vor der Pleite, Investition von 4000 Dollar wäre heute rund eine Million wert**
<https://www.heise.de/newsticker/meldung/Vor-20-Jahren-Apple-fast-pleite-3653863.html>
<https://www.zdf.de/dokumentation/zdf-history/jobs-gegen-gates---das-duell-der-computerkoenige-100.html>
- **Apple wertvollstes Unternehmen der Welt**
<https://www.heise.de/newsticker/meldung/Apple-Aktie-eilt-von-Rekord-zu-Rekord-3657961.html>
- **Artikel: KI-Libratus gewinnt im Poker gegen Profis**
<http://www.spiegel.de/netzwelt/web/poker-mensch-gegen-maschine-libratus-der-gangster-a-1133714.html>
- **Kauf von Fernseher, Notebook oder Smartphone**



Informationsquelle Heise.de



Grundlagen der EDV 1

- Geschichte der EDV
- Mikroprozessoren
- Der IBM PC
- Zahlensysteme
- Grundlagen der digitalen Arithmetik
- Alphanumerische Zeichen
- Unicode



Erste Rechenmaschine

Abakus 1100 v.Chr, aus dem indo-chinesischen Kulturraum, findet heute noch Verwendung



Geschichte von Rechenmaschinen

- Abakus ca. 1000 v.Chr.
- Rechenschieber 1617
- Rechenmaschine (**Schickard**) 1623
- Addiermaschine (**Pascal**) 1641
- Rechenmaschine (**Leibniz**) 1673
- Lochkarte (**Jacquard**) 1805
- Analytische Maschine (**Babbage**) 1833
- Lochkartenmaschine (**Hollerith**) 1882



Geschichte der EDV

- In der Zeit um den 2. Weltkrieg erfolgte ein enormer Leistungsschub. Vor allem durch den Einsatz von Relais wurden mechanische Rechenmaschinen weit übertroffen.
- Ein Pionier war der englische Mathematiker und Kryptograph **Alan Turing**. Er arbeitete an der Entschlüsselung der deutschen **ENIGMA** und entwickelte den **Turing-Test**. *Film The Imitation Game*
- **Konrad Zuse** entwickelte 1941 mit der **Z3** den ersten funktionstüchtigen Computer der Welt.



Geschichte der EDV

Technikevolution

- Erste elektromechanische Rechner auf Basis von Relais (klackern...)
- Ab ca. 1945 gab es Rechner aus Röhren (wurden sehr warm, mussten regelmäßig ausgetauscht werden...)
- Ab 1959 ermöglichten Halbleiter (Silizium, Germanium) Rechner auf Basis diskreter Transistoren (Erfinder: William Shockley)
- Heute basieren Rechner auf hochintegrierten Silizium-Chips



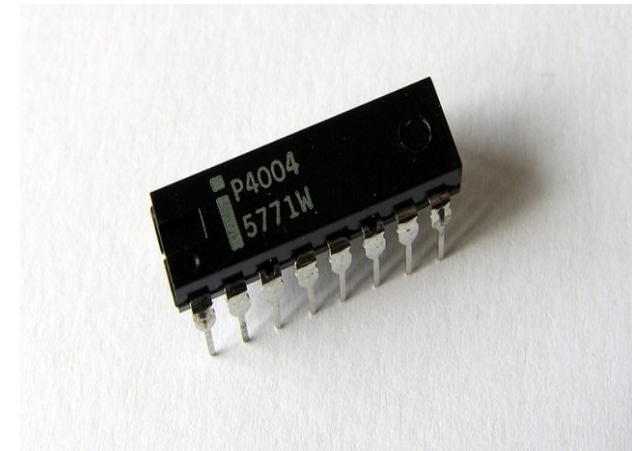
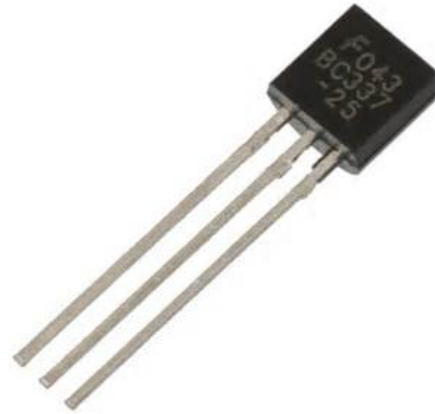
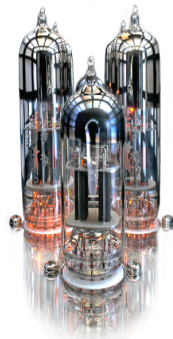
Technikevolution

Relais

Röhren

Transistoren

IC's (Chips)

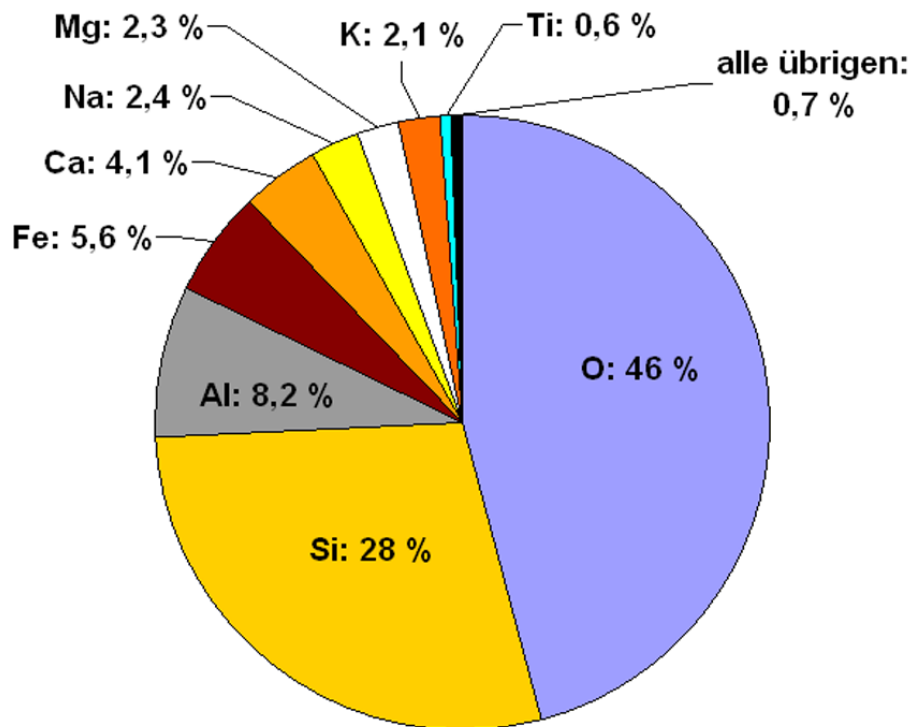


Halbleiter

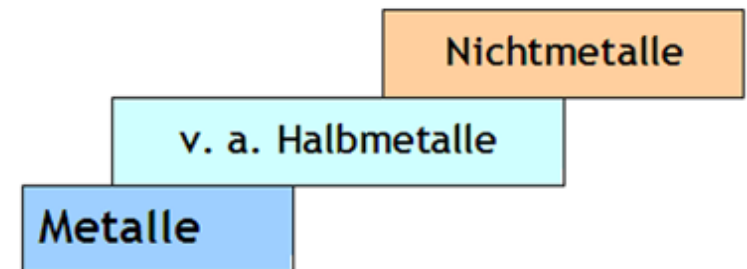
Schaltelemente

Silizium

- Monokristallin,
extreme Reinheit
weniger als 1 Teil pro Mrd.



	I	II	III	IV	V	VI	VII	VIII
1	H							He
2	Li	Be	B	C	N	O	F	Ne
3	Na	Mg	Al	Si	P	S	Cl	Ar
4	K	Ca	Ga	Ge	As	Se	Br	Kr
5	Rb	Sr	In	Sn	Sb	Te	I	Xe
6	Cs	Ba	Tl	Pb	Bi	Po	At	Rn
7	Fr	Ra						



Geschichte der EDV

Halbleitertechnik

- 1947 erfindet William Shockley den ersten Transistor auf Basis des Halbleiters Germanium
- Robert Noyce wurde während einem kurzen Telefongespräch mit W. Shockley vorgestellt ("Es war als würde ich mit Gott sprechen")
- 8 Mitarbeiter um R. Noyce gründen Fairchild Semiconductor und entwickeln 1959 den ersten integrierten Siliziumchip ([Wikipedia](#))
- 1968 gründet R. Noyce zusammen mit Gordon Moore die Intel Corp. und 1969 Teilhaber AMD



Geschichte der EDV

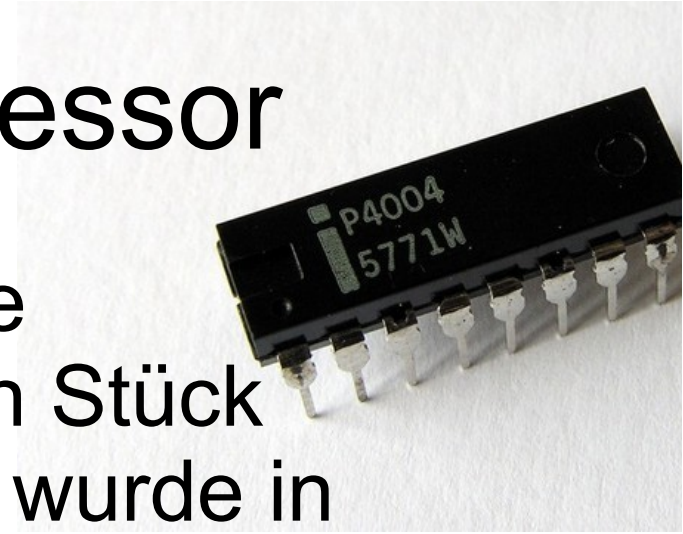
Halbleitertechnik

- 1958 entwickelte Jack Kilby bei der Firma Texas Instruments den ersten integrierten Schaltkreis auf Basis von Germanium. 10 Jahre später einigte er sich mit R. Noyce über die Patentrechte
- 1967 stellte er den darauf basierenden ersten Taschenrechner der Welt vor. Beide Erfindungen wurden von seinem Arbeitgeber kaum gewürdigt obwohl das wirtschaftliche Potential immens war (Nobelpreis 2000).



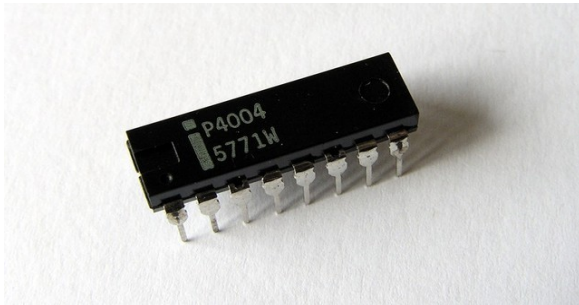
Der erste Mikroprozessor

- 1971 gelang es Intel erstmals alle Komponenten auf einem einzigen Stück Silizium zu integrieren: Der 4004 wurde in einem 10 μm Prozess mit 2250 Transistoren gefertigt.
- 1975 folgte der 8080 mit 8 Bit Busbreite als universell einsetzbarem Mikroprozessor.
- 1978 wurde der 8086 mit 16 Bit Busbreite vorgestellt. Er diente als Basis für den 8088 der im ersten IBM PC eingesetzt wurde und damit die Ära des PC einleitete.

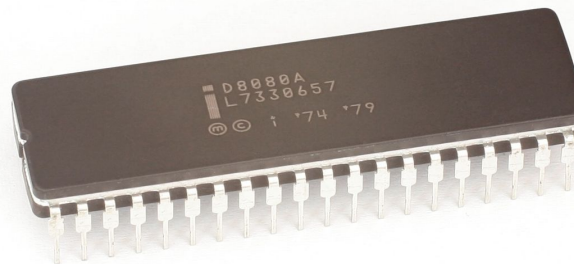


Chipevolution (Intel)

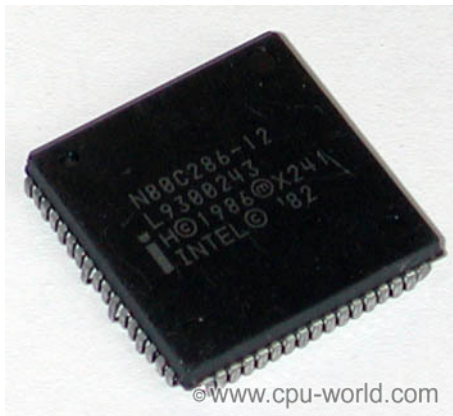
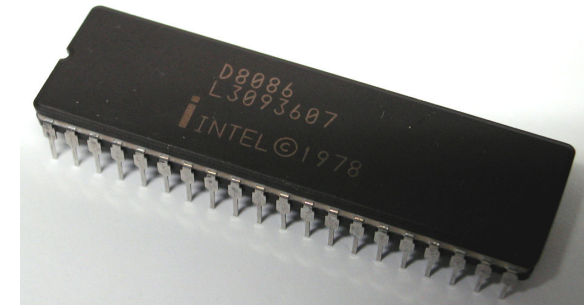
i4004 (1971)
2.300 Trans.



i8080 (1974)
6.000 T.



i8086 (1978)
29.000 T.



i80286 (1982)
134.000 T.



i80386 (1985)
274.000 T.



i80486 (1989)
1.200.000 T.

Chipevolution (Intel)

Pentium (1992)
4 Mio. Trans.



Pentium 4 (2001)
42 Mio. T.



Core i7 (2008) 1. Gen.
ca. 700 Mio. T.



Core i7 (2014) 5. Gen
ca. 2.600 Mio T.

Chipevolution (Motorola)

- Motorola 68000 (1979) 16/32 Bit 68.000 Trans. verwendet in Apple Macintosh, Atari ST, Commodore Amiga
- M 68020 (1984) 190.000 T.
- M 68030 (1987) 273.000 T.
- M 68040 (1991) 1.117.000 T.
- Da Apple 1994 auf Power PC Prozessoren wechselte verlor Motorola im PC-Sektor an Bedeutung



Chipevolution (Arm)

- ARM (Advanced RISC Machines) ist eine britische Firma die 1990 aus der Firma Acorn hervorging und Prozessordesigns verkauft, keine Chips.
- Acorn entwickelte seit 1983 eine 32-Bit Prozessorarchitektur die für geringen Energieverbrauch optimiert wurde.

Die ARM-Story: <http://www.heise.de/ct/artikel/Die-ARM-Story-1425834.html>

- Heute stecken in den meisten Tablets und Smartphones ARM-Prozessoren, z.B. im iPhone8 A10 von TSMC, im Galaxy S9 ein Exynos 9810, im HTC U12 ein Snapdragon 845 von Qualcomm und im Sony XZ2 ein Snapdragon 845

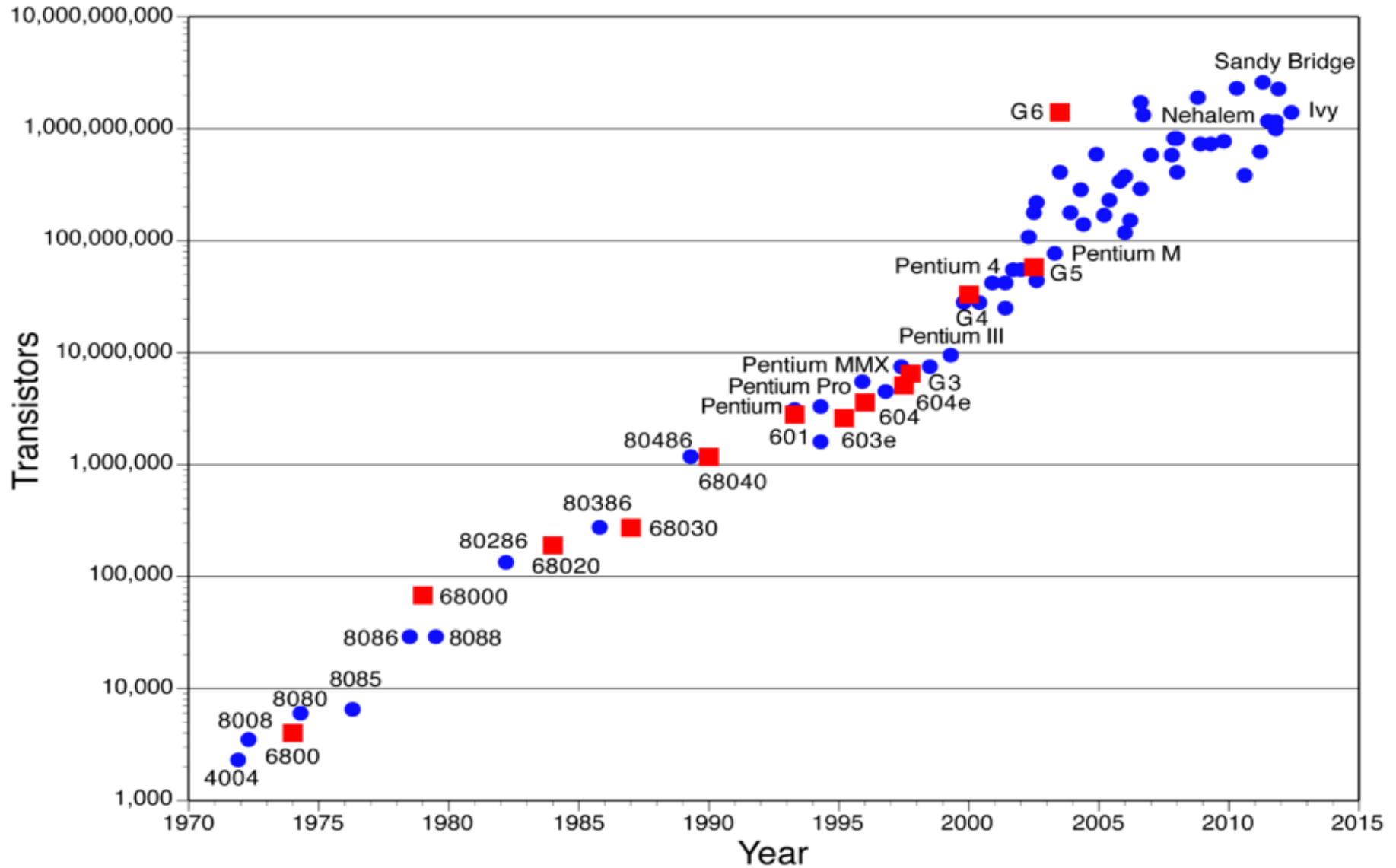


Moore's Law

- Moore's Gesetz besagt, dass die Anzahl der Transistoren (Schaltelemente) auf einem Chip sich alle 2 Jahre verdoppeln.
- Grund hierfür ist die laufende Verkleinerung der Strukturen auf dem Chip.
- Dieses Gesetz wurde 1965 von Gordon Moore aufgestellt. Es galt jedoch in ähnlicher Form bereits vorher und gilt bis über 2010 hinaus.
- Die Grenzen sind jedoch erkennbar, heute z.B. redet Intel von einer Verdoppelung alle 3 Jahre



Moore's Law



https://en.wikipedia.org/wiki/Transistor_count



Der IBM PC

- 1981 stellte IBM den ersten im eigenen Haus entwickelten **PC** fertig
- Offenes System, Betriebssystem von Microsoft, Prozessor von Intel, jedoch BIOS Eigenentwicklung IBM.
- Bus-System für Fremdhersteller offen



Der Industriestandard auf Basis des Prozessors i8088 und 8-Bit Bussystem

Der IBM PC

- Industriestandard
IBM kompatibel
- Leistung eher
bescheiden
- Aufgrund der
Anbietermacht für
Firmen interessant
- Riesiger Markt für
kompatible Hard- und
Software entstand

Welcome,
IBM.
Seriously.

Welcome to the most exciting and important marketplace since the computer revolution began 35 years ago.
And congratulations on your first personal computer.
Putting real computer power in the hands of the individual is already improving the way people work, think, learn, communicate and spend their leisure hours.


Computer literacy is fast becoming as fundamental a skill as reading or writing.

When we invented the first personal computer system, we estimated that over 140,000,000 people worldwide could justify the purchase of one, if only they understood its benefits.

Next year alone, we project that well over 1,000,000 will come to that understanding. Over the next decade, the growth of the personal computer will continue in logarithmic leaps.

We look forward to responsible competition in the massive effort to distribute this American technology to the world. And we appreciate the magnitude of your commitment.

Because what we are doing is increasing social capital by enhancing individual productivity.

Welcome to the task. 

IBM PC Evolution

- 1982 8086 PC XT (extended technology)
- 1984 80286 PC AT (advanced technology, 16 Bit)
- 1986 80386 32-Bit System
- 1987 **IBM PS/2**, mit Microchannel
- 1990 80486 mit integriertem Coprozessor für schnelle Fließkommaberechnung und Cache Speicher
- 1994 Pentium (80586 war nicht mehr als Markenzeichen frei)

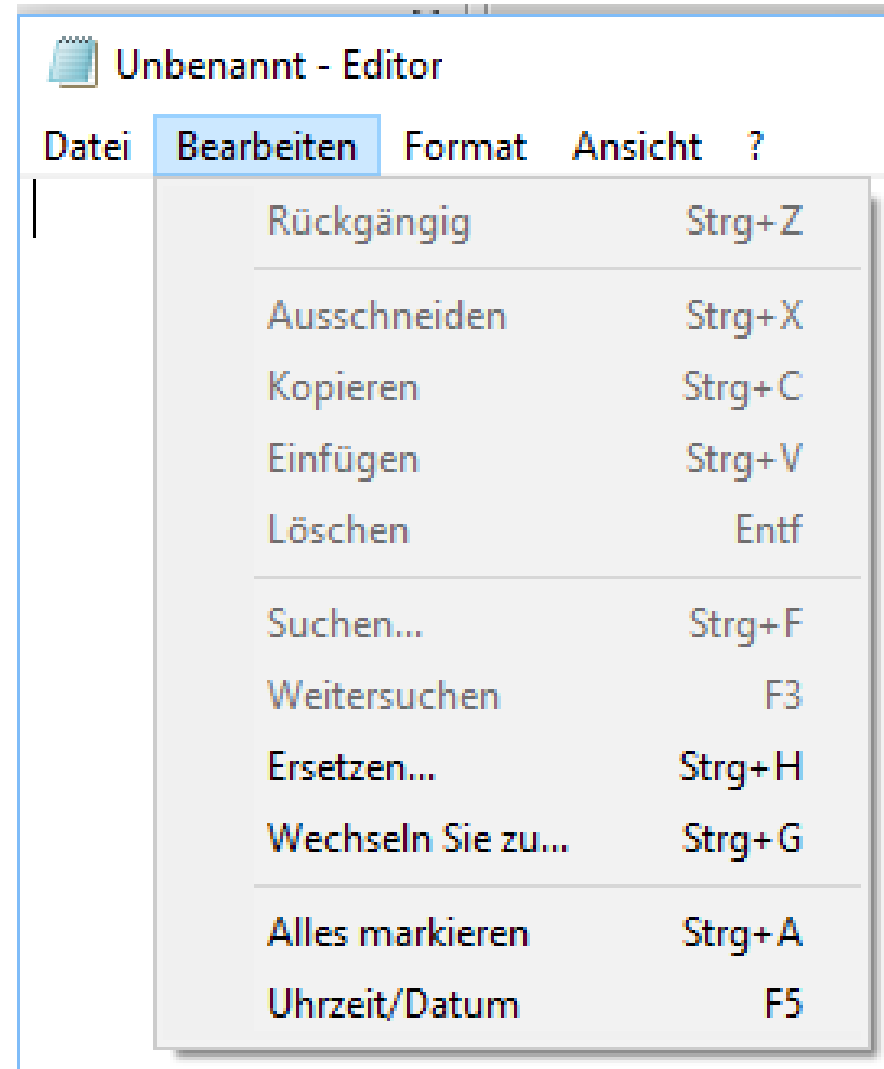
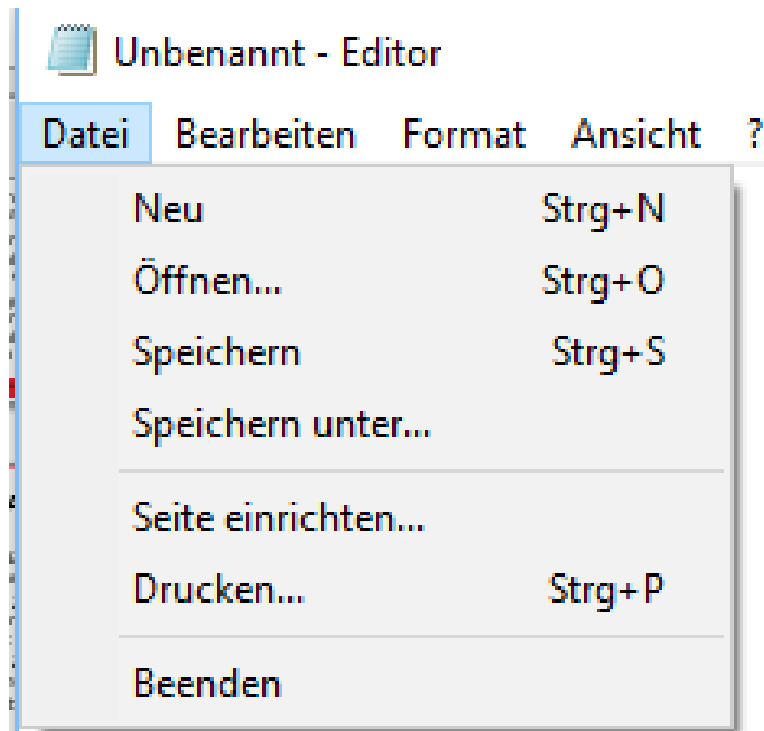


Übung hilfreiche Programme (Tools)

1. Notepad, Notepad++

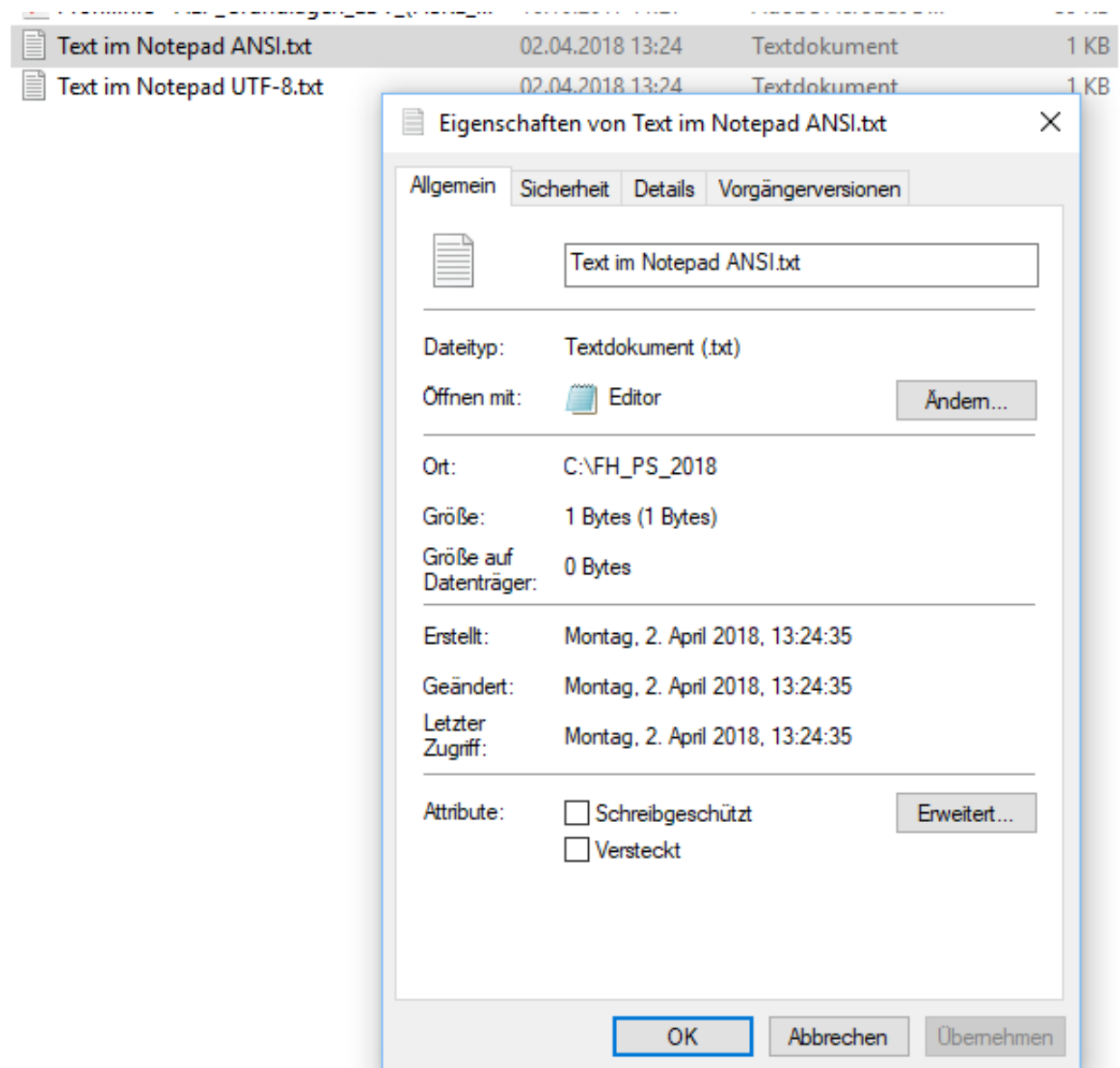
2. Ultraedit

3. Filezilla



Übung Kontextmenü

Rechte Maustaste
öffnet das
Kontextmenü



Übung Tastaturkürzel

1. Strg-A Alles Markieren
2. Strg-C Kopieren (in den Zwischenspeicher)
3. Strg-V Einfügen (aus dem Zwischenspeicher)
4. Strg-Z Rückgängig machen (Undo letzte Aktion)
5. Strg-Y Wiederherstellen (Redo nach Undo)
6. Strg-F Suchen (Find)
7. F3 Weitersuchen (nach einem Find)
8. Strg-S Speichern



Analog vs. Digital

- Analog: Wert- und zeitkontinuierlich, d.h. ohne Unterbrechungen und ohne Sprünge
- Digital: Diskrete Signale, duales Zahlenformat

	analog	digital
Musik	Schallplatte, Musikkassette	CD, MP3, Streaming
Film	Videorecorder	Festplattenrecorder
Fernsehen	Analogfernsehen	DVB-T, DVB-S
Radio	UKW Radio	DAB+ Radio
Uhren	Analoge Zeigeruhr	Digitaluhr
Telefon	Telefon	DECT, IP-Telefonie
Mobilfunk	C-Netz	GSM, UMTS, LTE
Kamera	Analoge Kamera	Digitalkamera



Digitale Evolution

- Die schnelle Entwicklung in der Digitaltechnik setzt ständig neue Standards.
- Z.B. Digitalradio DAB+ als Nachfolger von DAB oder DVB-T2 als Nachfolger von DVB-T. Dabei wurde der Vorgänger komplett eingestellt, z.B. DVB-T am 29.03.2017.
- Neue Standards erfordern weniger Speicherplatz, geringere Bandbreite (z.B. Frequenzen) bei der Übertragung oder ermöglichen höhere Auflösungen (Film, Foto...) und bietet Zusatznutzen



Analog vs. Digital

- Analoge Signale sind immer einem Rauschen ausgesetzt. Der Signal-Rauschabstand entscheidet über die Qualität des Signals. Jeder Kopiervorgang verschlechtert die Signalqualität.
- Die Information in digitalen Signalen ist eindeutig und wird auch durch Kopieren nicht verschlechtert.
- Aber: Viele Signale liegen nur in analoger Form vor (z.B. Tauchspulenmikrofon, Piezomikrofon) oder werden in analoger Form ausgegeben (Magnetlautsprecher, Piezolautesprecher)

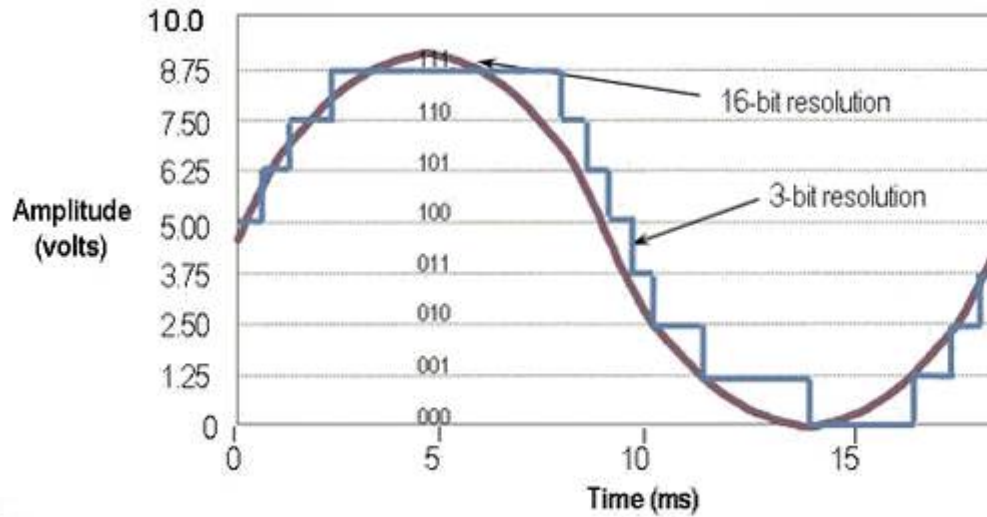


Warum Digital?

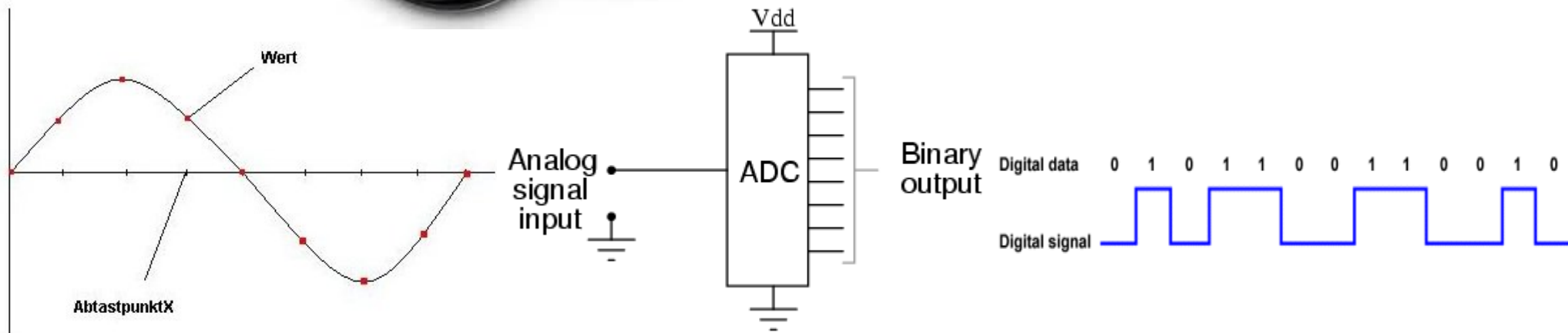
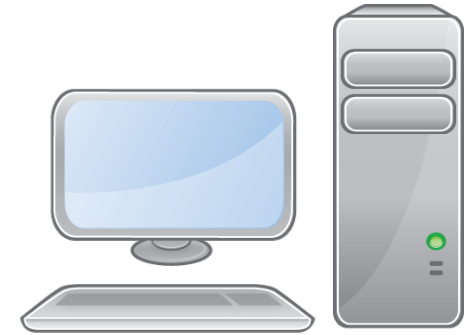
- Digitale Signale sind eindeutig (Spannung an oder Spannung aus. Der Abstand zwischen den beiden Zuständen ist sehr groß und damit nicht störanfällig.
- Digitaltechnik ermöglicht einfache Schaltungen mit sehr vielen sich wiederholenden Bauelementen, dadurch sehr flexibel. Analogtechnik setzt komplexe Schaltungstechnik voraus.
- Aber: Die Welt scheint analog zu sein



Analog-Digital-Wandlung



Analog-Digital-Wandlung



Analogsignal

Analog Digital Converter (ADC)

Digitalsignal

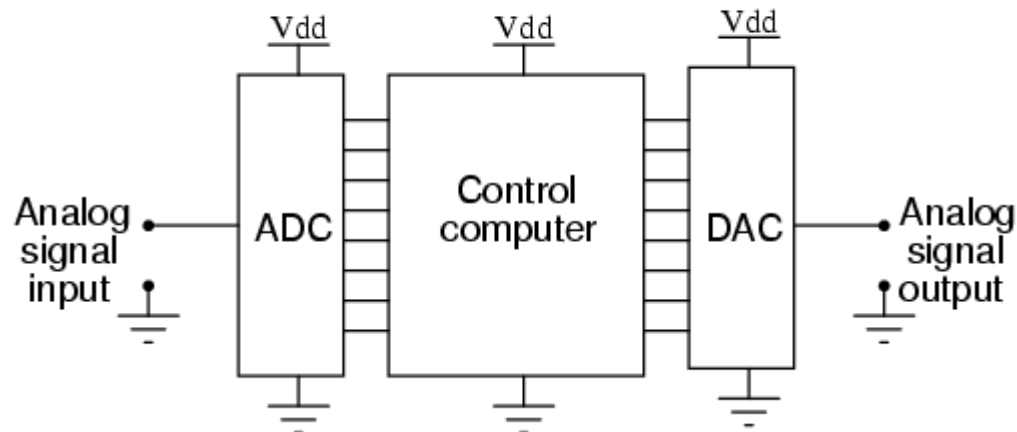
Die Abtastrate entscheidet über Signalqualität und Datenmenge

ADC - DAC Wandlung

EVA: Eingabe – Verarbeitung – Ausgabe



Digital control system with analog I/O



Rechnerinterne Zahlendarstellung

- Ein Rechner arbeitet intern mit dem dualen Zahlensystem (Binärsystem): Also mit **0** und **1**
Physikalische Bedeutung von 0 und 1:
0: Es liegt keine Spannung an
1: Es liegt eine definierte Spannung an
- Eine Binärzahl entspricht einem Bit
- In der EDV werden 8 Bit zu einem Byte zusammengefasst.
- Mit einem Byte können $2^8 = 256$ unterschiedliche Werte dargestellt werden.



Dualzahlen

- Jede Stelle einer Dualzahl hat eine Wertigkeit die der Stelle als 2er Potenz entspricht (analog den Zahlen im Dezimalsystem als 10er Potenz)
- Das Dualsystem verfügt über die Ziffern 0 und 1, das Zehnersystem hingegen die Ziffern 0..9

Stelle		7	6	5	4	3	2	1	0
Wertigkeit		128	64	32	16	8	4	2	1
Dualzahl		1	1	0	0	0	1	1	0
Summanden		128	64				4	2	
Summe	198								



Dualzahlen

- 16 Bit Dualzahl
- Umrechnung als Dezimalzahl einfach indem jede Stelle mit ihrer Wertigkeit multipliziert wird und alle Summanden aufaddiert werden

Stelle		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wertigkeit		32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
Dualzahl		0	1	1	0	1	0	0	0	1	1	0	0	0	1	1	0
Summanden			16384	8192		2048				128	64				4	2	
Summe	26822																



Dezimalzahlen - Addition

- Dualzahlen werden addiert indem jede Stelle einzeln, von rechts anfangend, addiert wird. Ist die Summe größer gleich 10, dann muss der Rest über zehn an diese Stelle geschrieben werden, die Anzahl der Zehner wird als Übertrag geschrieben.

- | | |
|------------------|------------|
| Zahl1: | 518 |
| Zahl2: | 649 |
| <u>Übertrag:</u> | <u>1 1</u> |
| Summe: | 1167 |



Dualzahlen - Addition

- Dualzahlen werden addiert indem jede Stelle addiert wird. Bei 2 Einsen wird eine Null an diese Stelle geschrieben und eine Eins an die nächste Stelle als Übertrag übergeben.
- Einfache Operation, daher sehr schnell

Stelle		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Wertigkeit		32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1	
Dualzahl 1		0	1	1	0	1	0	0	0	1	1	0	0	0	1	1	0	26822
Dualzahl 2		0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	21
Übertrag														1				
Summe		0	1	1	0	1	0	0	0	1	1	0	1	1	0	1	1	26843
Summanden			16384	8192		2048				128	64		16	8		2	1	
Summe	26843																	



Dualzahlen - Subtraktion

- Bei der Subtraktion wird zuerst die zu subtrahierende Zahl als negative Zahl dargestellt und dann die beiden Zahlen addiert
- Eine negative Zahl wird durch das **2er-Komplement** ermittelt. Dabei werden alle Bits negiert (aus Null wird Eins und umgekehrt) und eine Eins addiert.
- Das höchstwertige Bit ist dann immer 1, wohingegen es bei der positiven Zahl immer 0 ist (nur wenn negative Zahlen erlaubt sind)



Dualzahlen - Zweierkomplement

Stelle		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wertigkeit		32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
Dualzahl		0	1	1	0	1	0	0	0	1	1	0	0	0	1	1	0
Negation		1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	1
Addition 1		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Summe		1	0	0	1	0	1	1	1	0	0	1	1	1	0	1	0



Dualzahlen - Wertebereich

- Der Wertebereich einer Dualzahl ist immer begrenzt. Er ist abhängig von der Anzahl der Stellen, d.h. der Bits.
- Bei 8 Bit ergeben sich $2^8 = 256$ und bei 16 Bit $2^{16} = 65.536$ Möglichkeiten.
- Im Programm muss zur Interpretation bekannt sein um welche Art Zahl es sich handelt.

Wertebereich	mit Vorzeichen	ohne Vorzeichen
8 Bit	-128...+127	0...255
16 Bit	-32.768...+32.767	0...65.535
32 Bit	-2.147.483.648...+2.147.483.647	0...4.294.967.295



Wertebereich - Überlauf

- Da der Wertebereich von Zahlen mit fester Anzahl Stellen begrenzt ist kann es durch Rechenoperationen (z.B. Addition) zu einem Überlauf kommen.
- Beispiel 8 Bit Zahl, Addition von $255 + 1$ wäre 256, das ist aber eine 9-bit Zahl. D.h. das erste Bit wird ignoriert, es verbleiben die restlichen 8.



Jahr 2000 Problem

- Datumsangaben wurden früher häufig nur mit einer 2-stelligen Jahreszahl angezeigt und auch intern abgespeichert:
TT.MM.JJ
- In diesem Fall gab es im Jahr 2000 einen Überlauf der 2-stelligen Dezimalzahl ($99+1=00$)
- Ein Problem mit ähnlicher Ursache trat im Jahr 2010 unerwartet ein. Kreditkarten waren teilweise nicht mehr gültig und E-Mails wurden als Spam markiert.



Jahr-2038-Problem: Unixzeit

- In Unix und abgeleiteten Betriebssystemen (Linux, Android) wird die Zeit gezählt indem die Sekunden seit dem 01.01.1970 in einer vorzeichenbehafteten 32-bit Zahl gezählt werden
- Im Januar 2038 ist die Zahl voll und es findet ein Überlauf statt.
- Dies kann zu massiven Problemen, z.B. bei vielen embedded Sys. führen (lange Einsatzzeit)
- Abhilfe kann ein Wechsel auf 64-bit schaffen



Dualzahlen – Umrechnung Dezimal

- Dezimalzahlen können nach folgendem Schema in Dualzahlen umgerechnet werden:
 - Dividiere durch 2, merke das ganzzahlige Ergebnis und schreibe den Rest als Dualziffer.
 - Dividiere das vorhergehende Ergebnis wieder durch 2, merke das Ergebnis und schreibe den Rest als Dualziffer.
 - Solange bis das Ergebnis der Division Null ist
 -
 -



Umrechnung Dezimal- in Dualzahlen

- Beispielrechnung

Basis	Zahl	Rest	Stelle
2	195	1#	0
	97	1#	1
	48	0#	2
	24	0#	3
	12	0#	4
	6	0#	5
	3	1#	6
	1	1#	7



Hexadezimalzahlen

- Hexadezimalzahlen besitzen die Basis 16. D.h. für sie sind die Ziffern 0..9 und A..F definiert.
- Hexwerte ermöglichen eine gut lesbare Darstellung und leichte Umrechnung von Dualzahlen, da immer 4 Bit einer Dualzahl eine Ziffer einer Hexadezimalzahl definieren.
- Ein Byte wird also durch zwei Hex-Ziffern dargestellt.

9 = #09 10 = #A 20 = #14 30 = #1E



Hexadezimalzahlen – Umrechnung

Dezimal	Hexa	Binär		Dezimal	Hexa	Binär
0	0	0000		19	13	00010011
1	1	0001		20	14	00010100
2	2	0010				
3	3	0011				
4	4	0100		30	1E	00011110
5	5	0101		31	1F	00011111
6	6	0110		32	20	00100000
7	7	0111		33	21	00100001
8	8	1000		34	22	00100010
9	9	1001				
10	A	1010				
11	B	1011				
12	C	1100				
13	D	1101				
14	E	1110				
15	F	1111		252	FC	11111100
16	10	00010000		253	FD	11111101
17	11	00010001		254	FE	11111110
18	12	00010010		255	FF	11111111



Interpretation von Binärdaten

Binärdaten, also Kombinationen von "1" und "0", entsprechen immer einem Zahlenwert, in einem beliebigen Zahlensystem. Eine Speicherstelle mit diesem Zahlenwert kann jedoch unterschiedliche Bedeutung (Semantik) haben:

- Zahlen (z.B. Ganzzahl mit oder ohne Vorzeichen, Fließkommazahlen etc)
- Alphanumerische Zeichen (z.B. "a", "M", "\$" ...)
(Eingabe über Alt-Zahl auf dem Ziffernblock)
- Datum z.B. Sekunden seit 1970
- Farben



Datumsformate

- In Windows werden die Tage seit 0.1.1900 in einer Fließkommazahl gezählt
- In Unix werden die Sekunden seit 01.01.1970 in einer 32 Bit Ganzzahl (mit Vorzeichen) gezählt
- Excel, als Anwendungsprogramm, kann Datumsangaben in verschiedenen Formaten ausgeben z.B. 14.04.2016 oder 04/14/16 (je nach Sprachschema) oder den dahinterliegenden Zahlenwert anzeigen



Datum Grundlagen

- 1 Tag: Erde dreht sich einmal um ihre Achse
- 1 Jahr: Erde dreht sich einmal um die Sonne, ein Jahr dauert 365,2425 Tage
- Julianischer Kalender führt alle 4 Jahre ein Schaltjahr ein (eingeführt durch Julius Cäsar im Jahr 45 v. Chr.)
- Gregorianischer Kalender, basiert auf dem Julianischen, verfeinert aber die Regel für Schaltjahre (alle 100 Jahre keins, alle 400 Jahre doch wieder eins, seit 1582 durch Papst Gregor)



ASCII

American Standard Code for Information Interchange

- Ein Byte wird als alphanumerischer Wert bzw. Sonderzeichen interpretiert
- Die Zeichen im Bereich 0-127 sind eindeutig definiert, Nr. 32-126 sind druckbare Zeichen, z.B.: 0..9, a..z, A..Z, +, -, ,, * , # , % , (,) , = ...
- Die Zeichen im Bereich 128-256 werden unterschiedlich interpretiert
- Problem: Es gibt weltweit wesentlich mehr als 256 Buchstaben und Zeichen



ASCII Codepages

- Die ISO 8859 definiert 15 Teilnormen die jeweils eine Gruppe von Sprachen abbilden
 - 1: Westeuropäisch (L-1)
 - 2: Osteuropäisch
 - 3: Südeuropäisch
 - 4: Baltisch (L-4)
 - 5: Kyrillisch
 - 6: Arabisch
 - 7: Griechisch
 - 8: Hebräisch
 - 9: Türkisch
 - 10: Nordisch
 - 11: Thai
 - 12: - noch offen -
 - 13: Baltisch (L-7)
 - 14: Keltisch
 - 15: Westeuropäisch (L-9)
 - 16: Südosteuropäisch



ASCII-Tabelle nach ISO 8859-1

000	NUL	033	!	066	B	099	c	132	ä	165	ñ	198	ã	231	þ
001	Start Of Header	034	"	067	C	100	d	133	à	166	ª	199	Ä	232	Þ
002	Start Of Text	035	#	068	D	101	e	134	á	167	º	200	Å	233	Ú
003	End Of Text	036	\$	069	E	102	f	135	â	168	¸	201	Æ	234	Û
004	End Of Transmission	037	%	070	F	103	g	136	ê	169	¸	202	⌚	235	Ü
005	Enquiry	038	&	071	G	104	h	137	ë	170	ˆ	203	⌚	236	Ý
006	Acknowledge	039		072	H	105	i	138	è	171	½	204	⌚	237	Ÿ
007	Bell	040	(073	I	106	j	139	í	172	¼	205	=	238	ˉ
008	Backspace	041)	074	J	107	k	140	î	173	¸	206	⌚	239	˘
009	Horizontal Tab	042	*	075	K	108	l	141	ï	174	«	207	⌚	240	-
010	Line Feed	043	+	076	L	109	m	142	Ä	175	»	208	δ	241	±
011	Vertical Tab	044	,	077	M	110	n	143	Å	176	¸	209	Ð	242	_
012	Form Feed	045	-	078	N	111	o	144	É	177	¸	210	È	243	¾
013	Carriage Return	046	.	079	O	112	p	145	Ê	178	¸	211	É	244	↑
014	Shift Out	047	/	080	P	113	q	146	Ë	179		212	Ê	245	§
015	Shift In	048	0	081	Q	114	r	147	Ì	180		213	Ë	246	÷
016	Delete	049	1	082	R	115	s	148	Í	181	À	214	Ì	247	,
017	-- frei --	050	2	083	S	116	t	149	Î	182	Á	215	Í	248	°
018	-- frei --	051	3	084	T	117	u	150	Ï	183	Â	216	Î	249	˙
019	-- frei --	052	4	085	U	118	v	151	Û	184	Ã	217	Ï	250	.
020	-- frei --	053	5	086	V	119	w	152	Ü	185	¸	218	⌚	251	˘
021	Negative Acknowledge	054	6	087	W	120	x	153	Ý	186	¸	219	⌚	252	˚
022	Synchronous Idle	055	7	088	X	121	y	154	Û	187	⌚	220	⌚	253	˚
023	End Of Transmission Block	056	8	089	Y	122	z	155	¸	188	⌚	221		254	■
024	Cancel	057	9	090	Z	123	{	156	£	189	¸	222		255	
025	End Of Medium	058	:	091	[124		157	∅	190	¥	223	■		
026	Substitute	059	;	092	\	125	}	158	×	191	⌚	224	Ó		
027	Escape	060	<	093]	126	~	159	f	192	⌚	225	Ô		
028	File Separator	061	=	094	^	127	¸	160	á	193	⌚	226	Õ		
029	Group Separator	062	>	095	_	128	Ç	161	í	194	⌚	227	Ö		
030	Record Separator	063	?	096	`	129	ü	162	ó	195	⌚	228	ø		
031	Unit Separator	064	@	097	a	130	é	163	ú	196	-	229	Ö		
032		065	A	098	b	131	â	164	ñ	197	†	230	μ		

Übung ASCII Zeichen eingeben

1. Erstellen Sie mit dem Notepad-Editor eine neue Textdatei.
2. Geben Sie in der Datei verschiedene Zeichen über die Tastatur ein.
3. Geben Sie Zeichen ein indem Sie die Alt-Taste drücken und auf dem Nummernblock eine Zahl eingeben, dann die Alt-Taste loslassen.
4. Speichern Sie die Datei unter einem anderen Namen und verwenden beim Speichern einmal die ANSI Codierung und einmal UTF-8.



ASCII Codepages

- ASCII Codierung wird auch als ANSI Code bezeichnet (ANSI = American National Standards Institute)
- Die ersten 0..127 Zeichen sind standardisiert, das entspricht einer 7-Bit Codierung
- Für die weiteren 128..255 Zeichen gibt es je nach Ländereinstellung Codepages mit Sonderzeichen für die einzelnen Länder bzw. Sprachen (insgesamt 8 Bit)
- Beispielsweise deckt ISO 8859-1 möglichst viele westeuropäische Sprachen ab, aber z.B. französische Sonderzeichen fehlen, daher wurde die ISO 8859-15 entwickelt



ASCII Datenaustausch

- Codepages werden normalerweise automatisch vom Betriebssystem anhand der Ländereinstellung vorgegeben
- Die verwendete Codepage kann in manchen Programmen auch manuell geändert werden
- Auf einem anderen PC könnte jedoch eine andere Codepage voreingestellt sein
=> d.h. Zeichen ab 128 haben eine andere Bedeutung
- ASCII Datenaustausch ist daher problematisch
- Lösung: Unicode



Unicode

- Der zugewiesene Binärcode ist auf jedem Betriebssystem und in jeder Programmiersprache der selbe, z.B. für €-Zeichen immer 0x20AC
- Code-Raum umfaßt 17 Ebenen zu je 65.536 Zeichen
- Die Basis Multilingual Plane (Ebene 0) enthält hauptsächlich Schriftsysteme die aktuell in Gebrauch sind (es gibt auch veraltete Schriftsysteme)
- Unicode in Windows seit Windows 2000 bzw. XP
- Bisher ca. 10 % der möglichen Zeichen zugeordnet, d.h. es gibt noch Luft für neue Zeichen wie z.B. Globo



Unicode BMP 0

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF
E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF
F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF

- Lateinische Schriften und Symbole
- Lautschriften
- Andere europäische Schriften
- Nahost- und Südwestasiatische Schriften
- Afrikanische Schriften
- Südasiatische Schriften
- Südostasiatische Schriften
- Ostasiatische Schriften
- CJK-Ideogramme
- Kanadische Silben
- Symbole
- Diakritika
- UTF-16-Surrogates und privater Nutzungsbereich
- Verschiedene Zeichen
- Nicht belegte Codebereiche



UTF Formate

- Unicode wird in Textdateien meistens im UTF Format wieder gegeben
(Universal Transformation Format)
- Man unterscheidet UTF 8, UTF 16 und UTF 32
(die Zahl bedeutet nicht, dass z.B. alle Zeichen in UTF-8 mit 8 Bit kodiert sind, sondern sie sind meistens mit 8 Bit kodiert, können jedoch auch mit 16 oder 32 Bit kodiert sein)
- UTF-8 ist in den ersten 128 Zeichen deckungsgleich mit ASCII, damit lassen sich bereits viele westliche Sprachen mit einem Byte pro Zeichen wieder geben
(im März 2015 verwendeten 83,1 % aller Webseiten UTF-8)



UTF-8

Unicode-Zeichen größer als 127 werden in der UTF-8-Kodierung zu Byteketten der Länge zwei bis vier kodiert. Unicode-Bereich (hexadezimal)	UTF-8-Kodierung (binär)	Bemerkungen	Möglichkeiten (theoretisch)	
0000 0000 – 0000 007F	0xxxxxxx	In diesem Bereich (128 Zeichen) entspricht UTF-8 genau dem ASCII-Code: Das höchste Bit ist 0, die restliche 7-Bit-Kombination ist das ASCII-Zeichen.	2^7	128
0000 0080 – 0000 07FF	110xxxxx 10xxxxxx	Das erste Byte beginnt immer mit 11, die folgenden Bytes mit 10. Die xxxxx stehen für die Bits des Unicode-Zeichenwerts. Dabei wird das niederwertigste Bit des Zeichenwerts auf das rechte x im letzten Byte abgebildet, die höherwertigen Bits fortschreitend <i>von rechts nach links</i> . Die Anzahl der Einsen vor der ersten 0 im ersten Byte ist gleich der Gesamtzahl der Bytes für das Zeichen. (In Klammern jeweils die theoretisch maximal möglichen.)	$2^{11} - 2^7$ (2^{11})	1920 (2048)
0000 0800 – 0000 FFFF	1110xxxx 10xxxxxx 10xxxxxx		$2^{16} - 2^{11}$ (2^{16})	63.488 (65.536)
0001 0000 – 0010 FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx		2^{20} (2^{21})	1.048.576 (2.097.152)



ASCII - Unicode

1. Wie ist der ASCII Standard organisiert und wieviele Zeichen sind damit darstellbar?
2. Wieviele Zeichen können in Unicode maximal definiert werden bzw. wie ist der Unicodezeichensatz organisiert?
3. Wieviele Bytes beansprucht ein Zeichen wenn es als UTF-8 abgespeichert wird?
4. Wie werden deutsche oder französische oder polnische Sonderzeichen über ASCII abgebildet (ÄÖÜß áàâ ?)

