

# 64-040 Modul IP7: Rechnerstrukturen

[http://tams.informatik.uni-hamburg.de/  
lectures/2011ws/vorlesung/rs](http://tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/rs)

## Kapitel 3

Andreas Mäder



Universität Hamburg  
Fakultät für Mathematik, Informatik und Naturwissenschaften  
Fachbereich Informatik

**Technische Aspekte Multimodaler Systeme**

Wintersemester 2011/2012



# Kapitel 3

## Moore's Law

System on a chip

Smart Dust

Roadmap und Grenzen des Wachstums





# Moore's Law

- ▶ bessere Technologie ermöglicht immer kleinere Transistoren
- ▶ Materialkosten sind proportional zur Chipfläche
- ⇒ bei gleicher Funktion kleinere und billigere Chips
- ⇒ bei gleicher Größe leistungsfähigere Chips

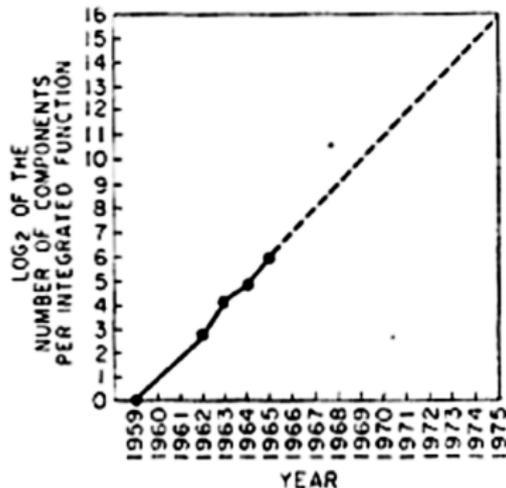
## Moore's Law

Gordon Moore, Mitgründer von Intel, 1965

Speicherkapazität von ICs vervierfacht sich alle drei Jahre

- ⇒ schnelles **exponentielles Wachstum**
  - ▶ klares Kostenoptimum bei hoher Integrationsdichte
  - ▶ trifft auch auf Prozessoren zu

## Moore's Law (cont.)

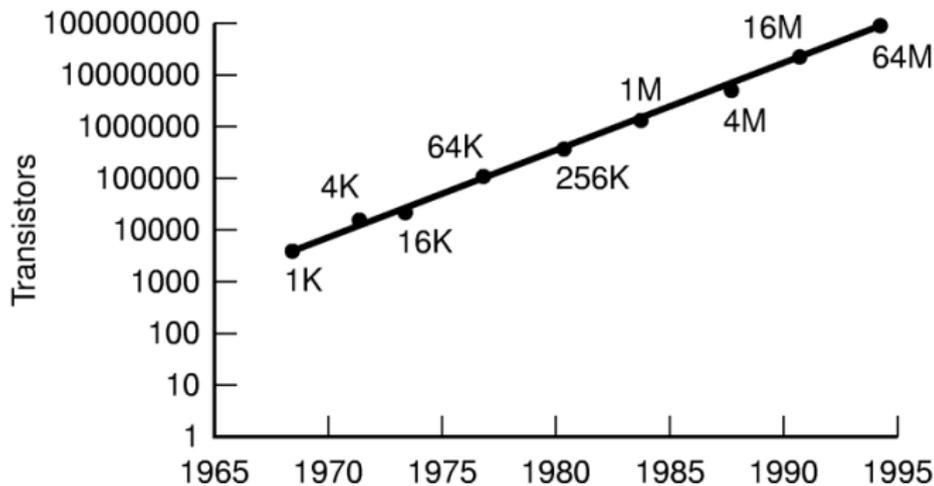


Gordon Moore 1965:  
 „Cramming more components onto integrated circuits“

*Wird das so weitergehen?*

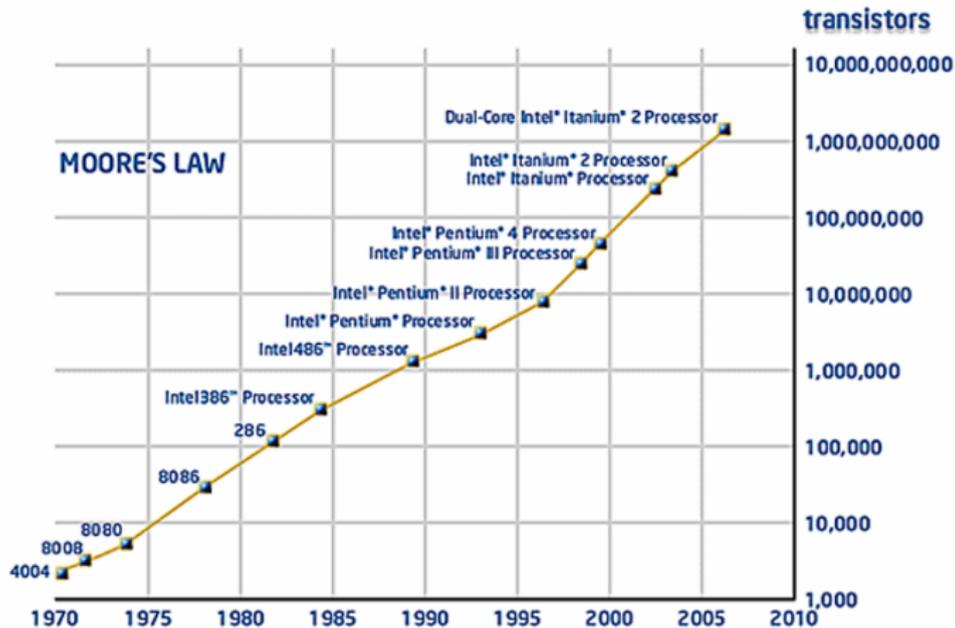
- ▶ Vorhersage gilt immer noch
- ▶ „ITRS“ Prognose bis über Jahr 2020 hinaus

# Moore's Law: Transistoren pro Speicherchip



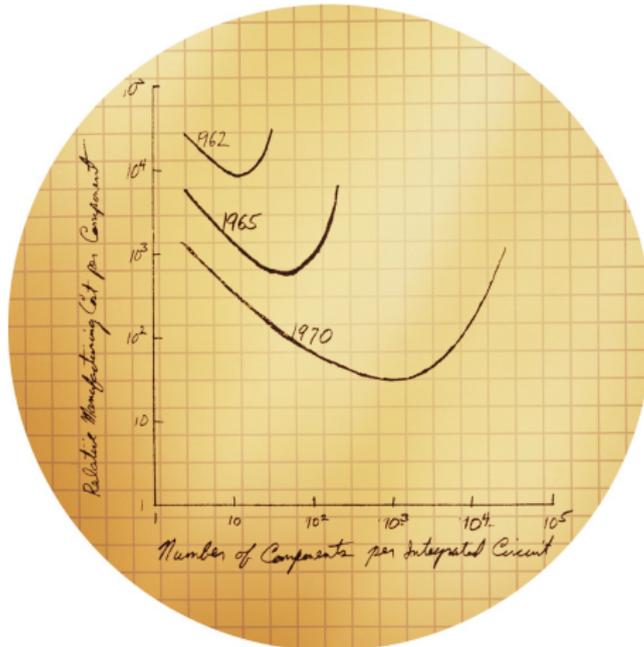
**Figure 1-8.** Moore's law predicts a 60 percent annual increase in the number of transistors that can be put on a chip. The data points given in this figure are memory sizes, in bits.

# Moore's Law: Evolution des Intel x86 (bis 2010)



Intel

# Moore's Law: Kosten pro Komponente



Originalskizze von G. Moore [www.intel.com](http://www.intel.com)



## Moore's Law: Formel und Beispiele

$$L(t) = L(0) \times 2^{t/18}$$

mit:  $L(t)$  = Leistung zum Zeitpunkt  $t$ ,  $L(0)$  = Leistung zum Zeitpunkt 0, und Zeit  $t$  in Monaten.

Einige Formelwerte:

Jahr 1:	1,5874
Jahr 2:	2,51984
Jahr 3:	4
Jahr 5:	10,0794
Jahr 6:	16
Jahr 7:	25,3984
Jahr 8:	40,3175



## Leistungssteigerung der Spitzenrechner seit 1993

Jahr	Rechner	Linpack in Gflop/s	Zahl der Prozessoren
1993	Fujitsu NWT	124	140
1994	Intel Paragon XP/S MP	281	6.768
1996	Hitachi CP-PACS	368	2.048
1997	Intel ASCI Red (200 MHz Pentium Pro)	1.338	9.152
1998	ASCI Blue-Pacific (IBM SP 640E)	2.144	5.808
1999	ASCI Intel Red (Pentium II Xeon)	2.379	9.632
2000	ASCI White, IBM (SP Power 3)	4.903	7.424
2002	Earth Simulator, NEC	35.610	5.104
2006	JUBL	45.600	16.384
2008	IBM Roadrunner	1.105.000	124.400 <sup>1</sup>
2009	Jaguar am ORNL, Cray	1.759.000	224.162 <sup>2</sup>

<sup>1</sup>Anzahl der Kerne (6.480 Opteron, 12.960 Cell)

<sup>2</sup>Anzahl der Kerne (Basis: Opteron)



## Moore's Law: Aktuelle Trends

- ▶ Miniaturisierung schreitet weiter fort
- ▶ aber Taktraten erreichen physikalisches Limit
- ▶ steigender Stromverbrauch, zwei Effekte:
  1. Leckströme
  2. proportional zu Taktrate

### Entwicklungen

- ▶ 4 GByte Hauptspeicher (und mehr) wird bezahlbar
- ▶ Übergang von 32-bit auf 64-bit Adressierung
- ⇒ Integration mehrerer CPUs auf einem Chip (Dual-/Quad-Core)
- ⇒ zunehmende Integration von Peripheriegeräten
- ⇒ ab 2011: CPU plus leistungsfähiger Grafikchip
- ⇒ **SoC**: „System on a chip“

## SoC: System on a chip

Gesamtes System auf einem Chip integriert:

- ▶ ein oder mehrere Prozessoren
- ▶ Befehls- und Daten-Caches für die Prozessoren
- ▶ Hauptspeicher (dieser evtl. auch extern)
- ▶ weitere Speicher für Medien/Netzwerkoperationen
- ▶ Peripherieblöcke nach Kundenwunsch konfiguriert:
  - ▶ serielle und parallele Schnittstellen, I/O-Pins
  - ▶ Displaysteuerung
  - ▶ USB, Firewire, SATA
  - ▶ Netzwerk kabelgebunden (Ethernet)
  - ▶ Funkschnittstellen: WLAN, Bluetooth, GSM/UMTS
  - ▶ Feldbusse: I<sup>2</sup>C, CAN, ...
- ▶ Handy, Medien-/DVD-Player, WLAN-Router, usw.

# SoC Beispiel: Bluetooth-Controller – Chiplayout

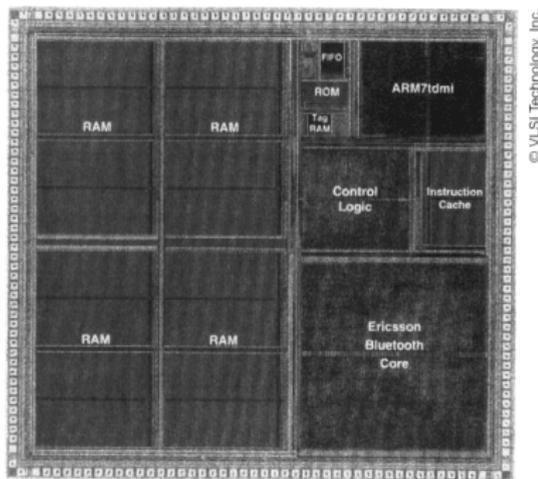


Figure 13.9 Bluetooth Baseband Controller die photograph.

Table 13.1 Bluetooth characteristics.

Process	0.25 $\mu\text{m}$	Transistors	4,300,000	MIPS	12
Metal layers	3	Die area	20 $\text{mm}^2$	Power	75 mW
V <sub>dd</sub> (typical)	2.5 V	Clock	0–13 MHz	MIPS/W	160

S. Furber, *ARM System-on-Chip Architecture*, 2000

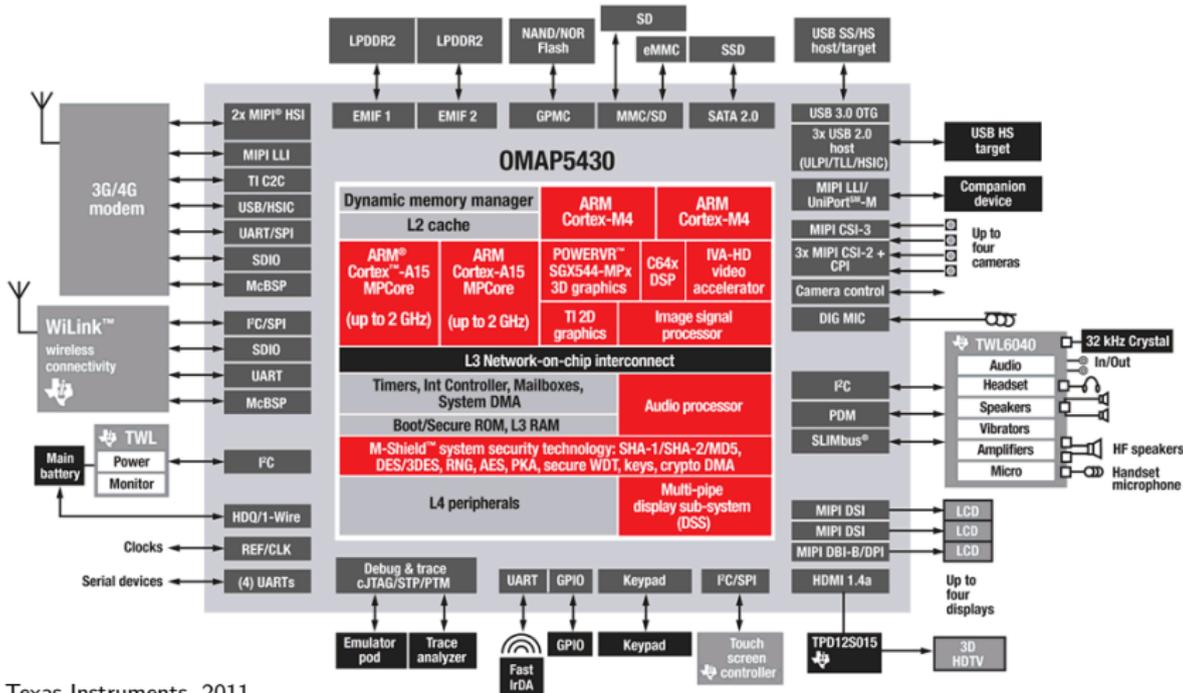
# SoC Beispiel: OMAP 5430

- ▶ mehrere (verschiedene) CPUs
- ▶ Grafikbeschleuniger
- ▶ Chipsatz (Speichercontroller, Interconnect, ...)
- ▶ Schnittstellen (WiFi, 4G, USB, Audio, I/O, ...)

## OMAP5430 Key Benefits

- Designed to drive Smartphones, Tablets and other multimedia-rich mobile devices
- Multi-core ARM® Cortex™ processors
  - Two ARM Cortex-A15 MPCore processors capable of speeds up to 2 GHz each
  - Two ARM Cortex-M4 processors for low-power offload and real-time responsiveness
- Multi-core POWERVR™ SGX544-MPx graphics accelerators drive 3D gaming and 3D user interfaces
- Dedicated TI 2D BitBlit graphics accelerator
- IVA-HD hardware accelerators enable full HD 1080p60, multi-standard video encode/decode as well as 1080p30 stereoscopic 3D (S3D)
- Faster, higher-quality image and video capture with up to 24 megapixels (or 12 megapixels S3D) imaging and 1080p60 (or 1080p30S3D) video
- Supports four cameras and four displays simultaneously
- Packaging and memory: 14mm x 14mm, 0.4mm pitch PoP dual-channel LPDDR2 memory

# SoC Beispiel: OMAP 5430 (cont.)



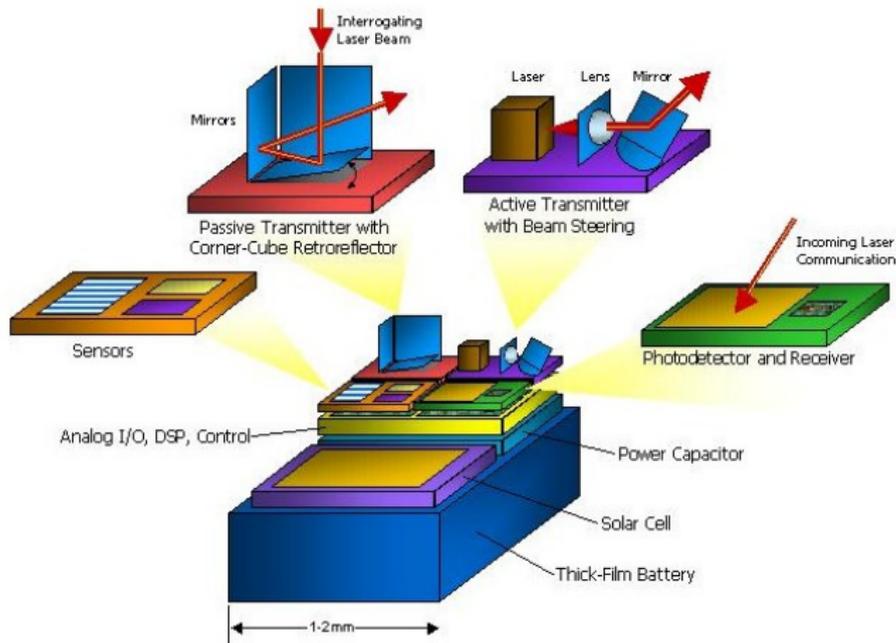
Texas Instruments, 2011

# Smart Dust

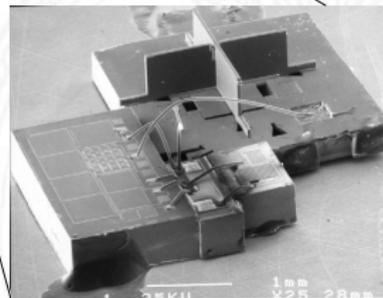
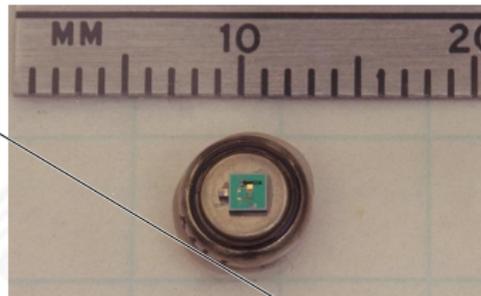
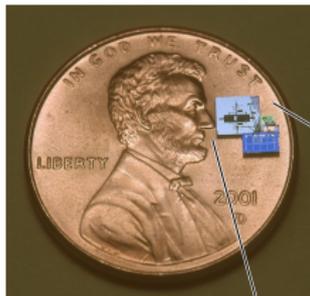
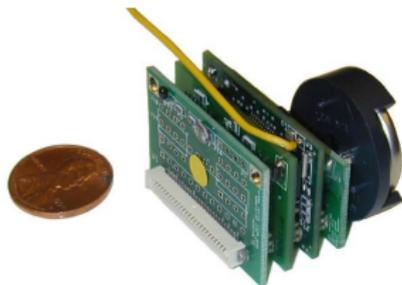
## Wie klein kann man Computer bauen?

- ▶ Berkeley Projekt: **Smart Dust** 2002-2006
- ▶ Integration kompletter Rechensysteme auf  $1 \text{ mm}^3$ 
  - ▶ vollständiger Digitalrechner CPU, Speicher, I/O
  - ▶ Sensoren Photodioden, Kompass, Gyro
  - ▶ Kommunikation Funk, optisch
  - ▶ Stromversorgung Photozellen, Batterie, Vibration, Mikroturbine
  - ▶ Echtzeit-Betriebssystem Tiny OS
  - ▶ inklusive autonome Vernetzung
- ▶ Massenfertigung? Tausende autonome Mikrorechner
- ▶ „Ausstreuen“ in der Umgebung
- ▶ vielfältige Anwendungen

# Smart Dust: Konzept



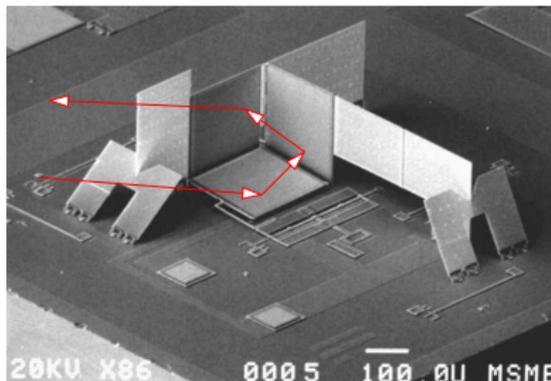
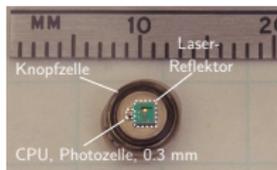
# Smart Dust: Prototypen



diverse Prototypen:

- vollwertige CPU / Sensoren / RF
- "out-door"-tauglich
- MEMS-"CCR" für opt. Kommunikation

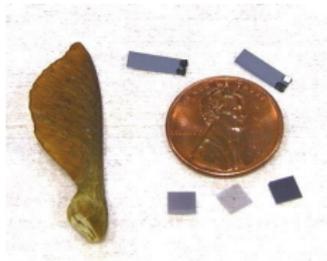
# Smart Dust: Corner-cube reflector („Katzenaugen“)



- ▶ CCR: seitlich zwei starre Spiegel, Gold auf Silizium
- ▶ untere Spiegelfläche beweglich (elektrostat. ca. 30 V)
- ▶ gezielte Modulation von eingestrahltm Laserlicht
- ▶ Reichweiten  $> 100$  m demonstriert

# Smart Dust: Energieverbrauch

Miniatur-Solarzellen  
 Wirkungsgrad ca. 3%  
 26  $\mu\text{W}/\text{mm}$  in vollem Sonnenlicht



Batterien:	$\sim 1\text{J}/\text{mm}^2$	
Kondensatoren:	$\sim 10\text{ mJ}/\text{mm}^2$	
Solarzellen:	$\sim 0.1\text{ mW}/\text{mm}$	$\sim 1\text{J}/\text{mm} / \text{day}$ (außen,Sonne)
	$\sim 10\text{ }\mu\text{W}/\text{mm}$	$\sim 10\text{mJ}/\text{mm} / \text{day}$ (innen)
Digitalschaltung	1 nJ/instruction	(StrongArm SA1100)
Analoger Sensor	1 nJ/sample	
Kommunikation	1 nJ/bit	(passive transmitter, s.u.)
opt. digitale ASICs:	$\sim 5\text{ pJ}/\text{bit}$	(LFSR Demonstrator, 1.4V)



## Grenzen des Wachstums

- ▶ Jeder exponentielle Verlauf stößt irgendwann an natürliche oder wirtschaftliche Grenzen.
- ▶ Beispiel: eine DRAM-Speicherzelle speichert derzeit etwa 100.000 Elektronen. Durch die Verkleinerung werden es mit jeder neuen Technologiestufe weniger.
- ▶ Offensichtlich ist die Grenze spätestens erreicht, wenn nur noch ein einziges Elektron gespeichert würde.
- ▶ Ab diesem Zeitpunkt gibt es bessere Performance nur noch durch bessere Algorithmen / Architekturen
- ▶ Annahme: 50 % Wachstum pro Jahr,  $a^b = \exp(b \cdot \ln a)$
- ▶ Elektronen pro Speicherzelle:  $100000 / (1.5^{x/\text{Jahre}}) \geq 1$ .
- ▶  $x = \ln(100.000) / \ln(1.5) \approx 28$  Jahre

# Roadmap: ITRS

## International **T**echnology **R**oadmap for **S**emiconductors

<http://www.itrs.net/reports.html>

- ▶ non-profit Organisation
- ▶ diverse Fördermitglieder
  - ▶ Halbleiterhersteller
  - ▶ Geräte-Hersteller
  - ▶ Unis, Forschungsinstitute
  - ▶ Fachverbände aus USA, Europa, Asien
- ▶ Jährliche Publikation einer langjährigen Vorhersage
- ▶ Zukünftige Entwicklung der Halbleitertechnologie
- ▶ Komplexität typischer Chips (Speicher, Prozessoren, SoC, ...)
- ▶ Modellierung, Simulation, Entwurfssoftware



# Moore's Law: Schöpferische Pause

Beispiel für die Auswirkung von Moore's Law.

Angenommen die Lösung einer Rechenaufgabe dauert derzeit vier Jahre, und die Rechenleistung wächst jedes Jahr um 60%.

*Wie lösen wir das Problem ?*



# Moore's Law: Schöpferische Pause

Beispiel für die Auswirkung von Moore's Law.

Angenommen die Lösung einer Rechenaufgabe dauert derzeit vier Jahre, und die Rechenleistung wächst jedes Jahr um 60%.

Ein mögliches Vorgehen ist dann das folgende:

- ▶ Wir warten drei Jahre, kaufen dann einen neuen Rechner und erledigen die Rechenaufgabe in einem Jahr.
- ▶ *Wie das ?*



# Moore's Law: Schöpferische Pause

Beispiel für die Auswirkung von Moore's Law.

Angenommen die Lösung einer Rechenaufgabe dauert derzeit vier Jahre, und die Rechenleistung wächst jedes Jahr um 60 %.

Ein mögliches Vorgehen ist dann das folgende:

- ▶ Wir warten drei Jahre, kaufen dann einen neuen Rechner und erledigen die Rechenaufgabe in einem Jahr.
- ⇒ Nach einem Jahr können wir einen Rechner kaufen, der um den Faktor 1,6 Mal schneller ist, nach zwei Jahren bereits  $1,6 \times 1,6$  Mal schneller, und nach drei Jahren (also am Beginn des vierten Jahres) gilt  $(1 + 60\%)^3 = 4,096$ .
- ▶ Wir sind also sogar ein bisschen schneller fertig, als wenn wir den jetzigen Rechner die ganze Zeit durchlaufen lassen.



## Wie geht es jetzt weiter?

Ab jetzt erst mal ein *bottom-up* Vorgehen: Start mit grundlegenden Aspekten, dann Kennenlernen aller Komponenten des Digitalrechners und Konstruktion eines vollwertigen Rechners.

- ▶ Grundlagen der Repräsentation von Information
- ▶ Darstellung von Zahlen und Zeichen
- ▶ arithmetische und logische Operationen
- ▶ ...
  
- ▶ Vorkenntnisse nicht nötig (aber hilfreich)