

# **Rechnerarchitektur**



**Jürgen Ruf**

**Institut für Informatik**

# Ansprechpartner

- Jürgen Ruf
  - Raum 02-033
  - Gebäude 052
  - Sprechzeiten
    - Dienstags von 14:00-15:00
    - Donnerstags von 13:00-14:00
  - email: [jruf@informatik.uni-freiburg.de](mailto:jruf@informatik.uni-freiburg.de)

# Wer bin ich

- Vertretung der Professur für Betriebssysteme am Lehrstuhl für Rechnerarchitektur
- Informatik Studium in Karlsruhe
- Promotion in KA
- seit 1999 Postdoc in Tübingen
- Forschungsinteressen
  - HW/SW-Systementwurf (Vertiefungsvorlesung, MI 10-12)
    - | Systembeschreibungssprachen, Entwurfsmethodik
  - Verifikation/Simulation (Seminar, DO 10-12)
    - | formale Methoden
    - | Kombination

# Organisation

- Scheinvergabe:  
Bestehen der Klausur am Ende des Semesters
- Vorlesung/Übung donnerstags:  
im Wechsel jeweils 1,5 Stunden
- Vorlesung:  
45 Minuten, 5 Minuten Pause, 45 Minuten

# Literatur

- Patterson/Hennessy: Computer Organization and Design, **1997**
- Hennessy/Patterson: Computer Architecture, **1996**
- A. Tanenbaum: Structured Computer Organization (4th Edition), **1999**
- Hayes: Computer Architecture and Organization, **1998**

# Was ist Rechnerarchitektur?

# Was ist Rechnerarchitektur?

## ■ Rechner

# Was ist Rechnerarchitektur?

- Rechner
- -Architektur

# Rechner

- „Ein Rechner ist ein universell einsetzbares Gerät zur automatischen Verarbeitung von Daten“ [Duden]

# -Architektur

# -Architektur

## ■ „externe“

„The term architecture is used here to describe the attributes of a system as seen by the programmer ...“

[Amdahl, Blaauw, Brooks, siehe Bode/Händler 1980]

# -Architektur

## ■ „externe“

„The term architecture is used here to describe the attributes of a system as seen by the programmer ...“

[Amdahl, Blaauw, Brooks, siehe Bode/Händler 1980]

## ■ „interne“

„The study of computer architecture is the study of the organization and interconnection of components of computer systems ...“

[Stone, siehe Bode/Händler 1980]

# Zur Einstimmung

# Zur Einstimmung

- das „Szenario“
  - von den Anfängen zum Pentium,  
Chips mit 46 Millionen Transistoren  
**aber ...**

# das Szenario

# das Szenario

- „Frühgeschichte“

# das Szenario

- „Frühgeschichte“
- Erste frei programmierbare Rechner
  - 1934-41 Zuse Z1, Z3
  - 1946 ENIAC

# das Szenario

- „Frühgeschichte“
- Erste frei programmierbare Rechner  
1934-41 Zuse Z1, Z3  
1946 ENIAC
- Pentium 4

# Frühgeschichte

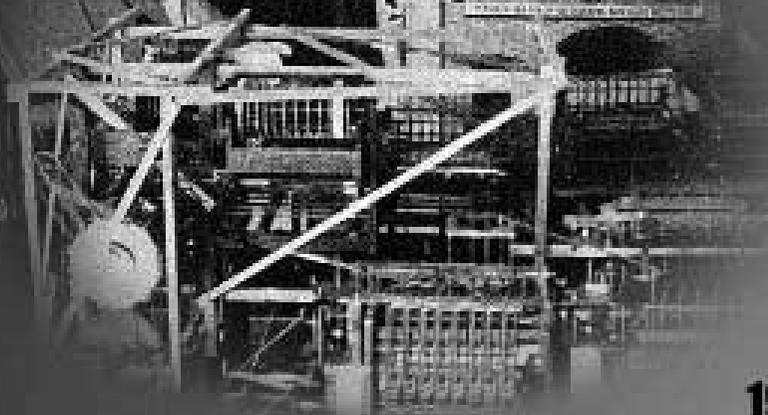
- Altertum: griechische Zahnräder, römisches Rechenbrett
- Rechenmaschine von Schickard
- Lochkartenmaschine (um die Jahrhundertwende)
- Differentialanalysator (Bush, **1930**)

# Frühgeschichte

- Theoretische Abhandlungen:
  - Turing (**1936**): "On Computable Numbers"
  - Shannon (**1936**): Verbindung zwischen symbolischer Logik und elektrischen Schaltungen
- Binäre Addiermaschine (Stibitz, **1937**)
- Allzweck-Computer (Zuse, **1938**)

# Rechnerarchitektur im Wandel der Zeit

**Zuse Z1**



1938

**Commodore 64**



1982

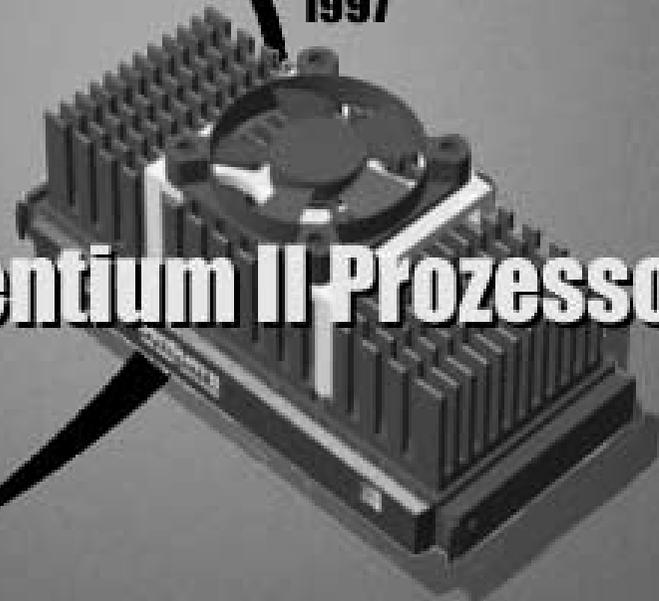
1965

**IBM 360**



1997

**Pentium II Prozessor**



2xxx

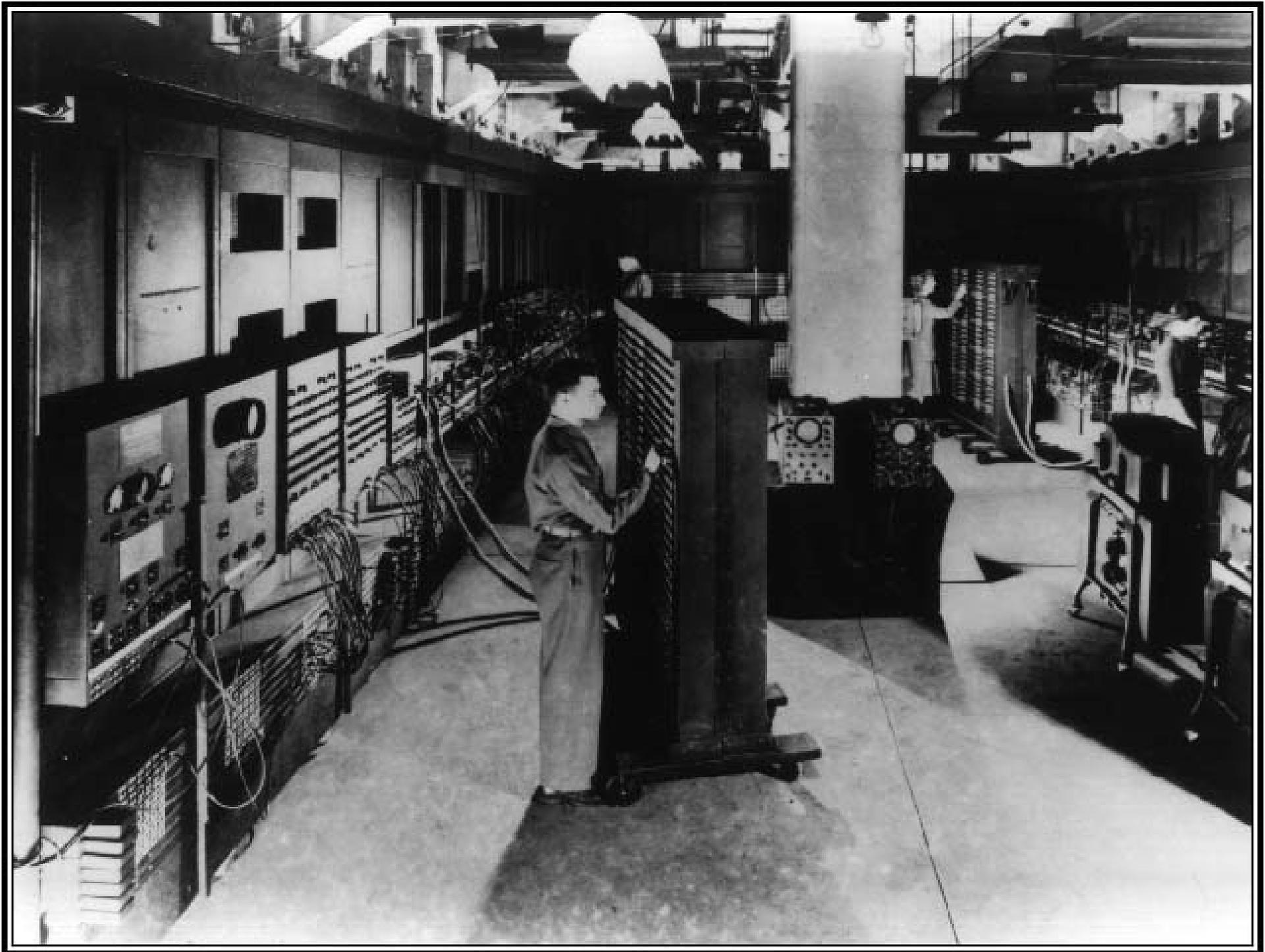
# Geschichte des Rechners

- ***Colossus*** (**1943**): Knacken von Geheimcodes
- von Neumann beschreibt wesentliche Elemente eines Computersystems (**1945**)
- ***ENIAC*** (**1946**): erster (großer) elektronischer Digitalrechner

# ENIAC

# ENIAC

- 30 Tonnen, 3m hoch, 24m breit
- 18 000 Elektronenröhren
- Multiplikationszeit: 3ms



# 1943-50

# 1943-50

- Thomas Watson (IBM):  
„I think there is a world market for  
maybe five computers.“

# 1943-50

- Thomas Watson (IBM):  
„I think there is a world market for maybe five computers.“
- Popular Mechanics:  
„Computers in the future may weigh no more than 1.5 tons.“

# Auf der anderen Seite:

- Erfindung des Transistors (**1947**) durch Bardeen, Shockley und Brattain
- **LEO** (**1951**): erster Rechner für kaufmännische Zwecke
- Computergestützte Wahlprognose in USA (**1952**)
- Erster Magnetspeicher (Forrester, **1953**), zuvor Elektronenröhrenspeicher

- **IBM 650 (1954)**: in 15 Jahren 1500mal verkauft
- Transistoren auf Silizium (Teal, **1954**)
- **TRADIC (1955)**: erster Transistor-Computer der Bell Labs mit 800 Transistoren
- Integrierte Schaltungen und MOS (metal oxide semiconductors)

- ***PDP-8*** (DEC, **1965**): erster Minicomputer mit kommerziellem Erfolg (20.000 Dollar)
- ***Intel 4004*** (Hoff, **1971**): Chip, mit Zentraleinheit als wesentlichen Bestandteil
- ***Altair*** (Roberts, **1974**): Bausatz eines Mikrocomputers für 397 Dollar

- BASIC für *Altair* (Gates, Allen, **1975**)
- PCs für den 'Massenmarkt' (**1977**), z.B. *Apple II, Radio Shack TRS-80, Commodore PET*
- VisiCalc (**1979**): erste kaufmännische Software für PC
- IBM PC (**1981**)

# 1995

# 1995

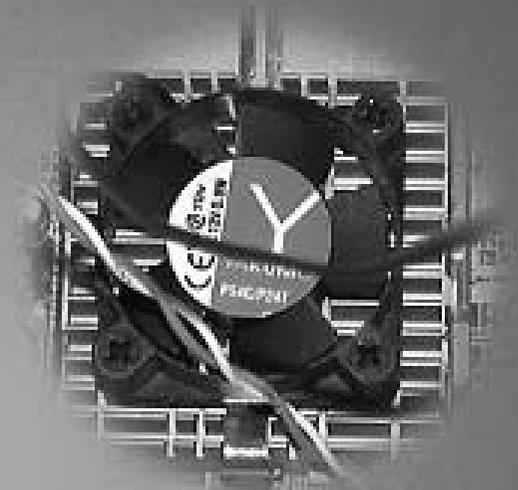
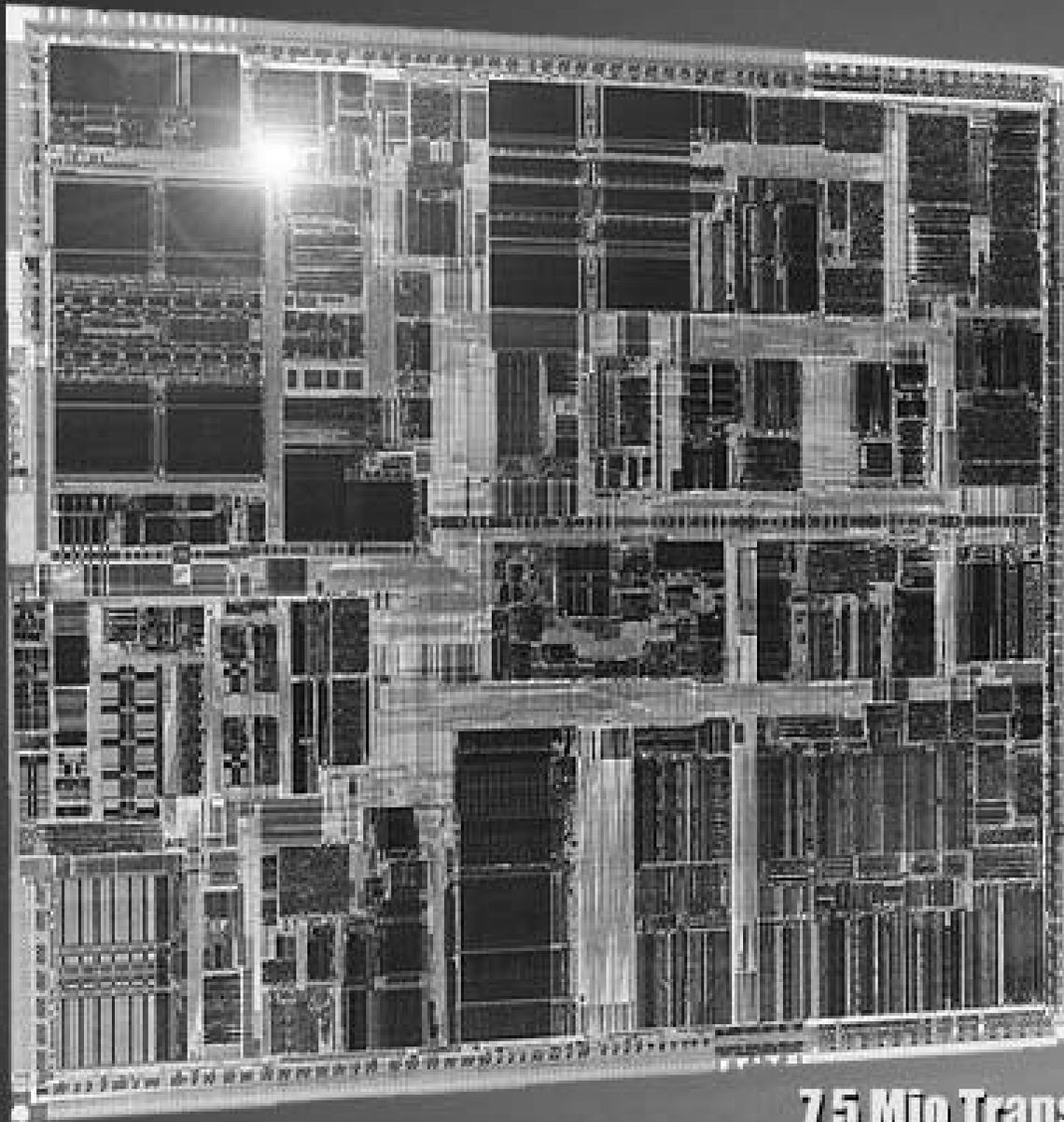
- ENIAC on a chip
- Laptop = alle Rechner der Welt in 1950

# Pentium II

# Pentium II

- 7.5 Millionen Transistoren
- Chip: 1.2 cm Kantenlänge
- 450 MHz

# Chip-Layout



**7.5 Mio Transistoren auf einem Chip**

# Pentium 4

- Chipgröße 217 mm<sup>2</sup>
- Zahl der Transistoren 42 Millionen
- Taktfrequenz 2 GHz
- Taktzyklen pro Instruktion 0.17

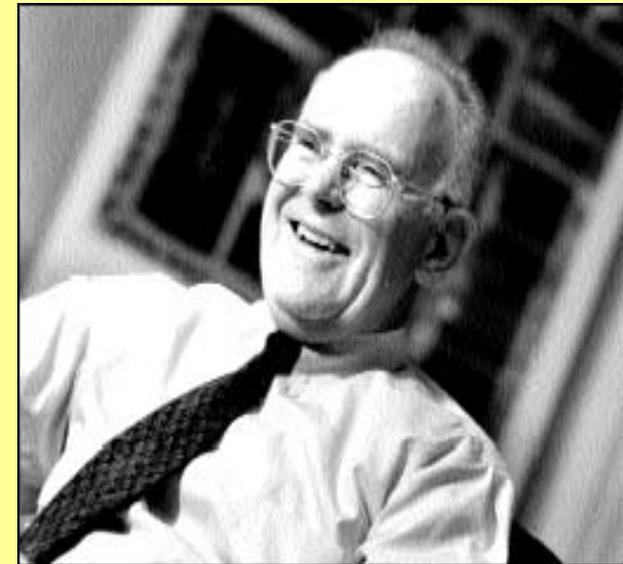
# Moore's Law

# Moore's Law

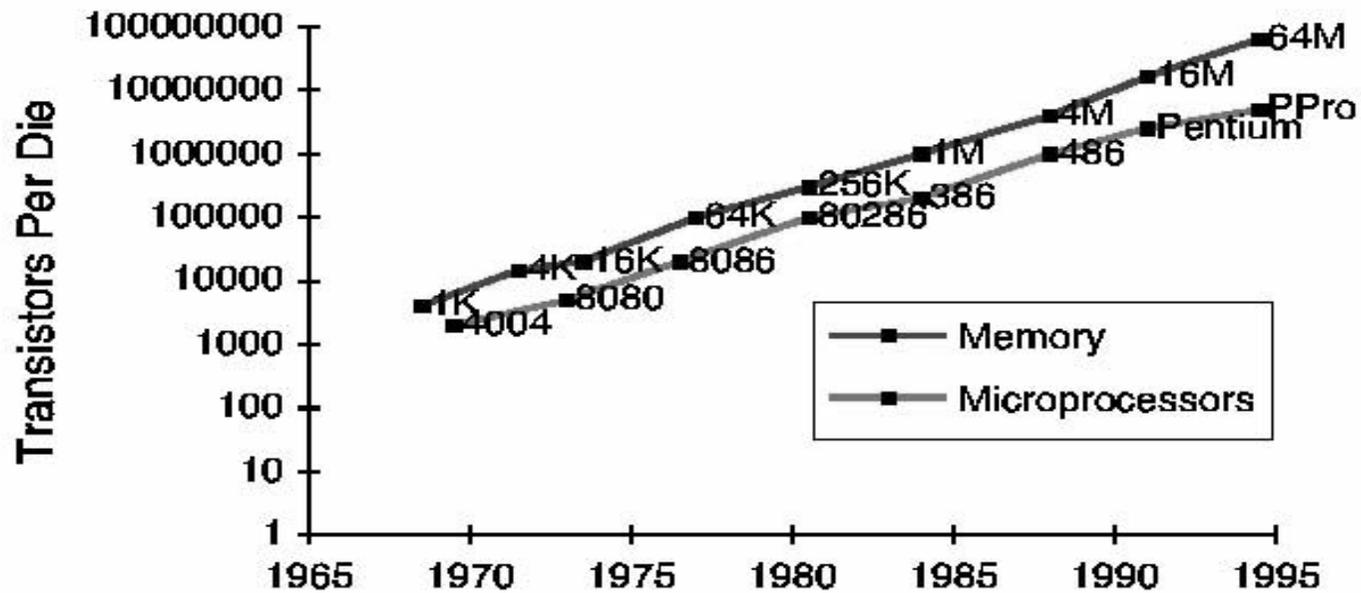
- Verdopplung der Transistor-Dichte alle 18 Monate  
(Gordon Moore, Mitbegründer von Intel, 1965)

# Moore's Law

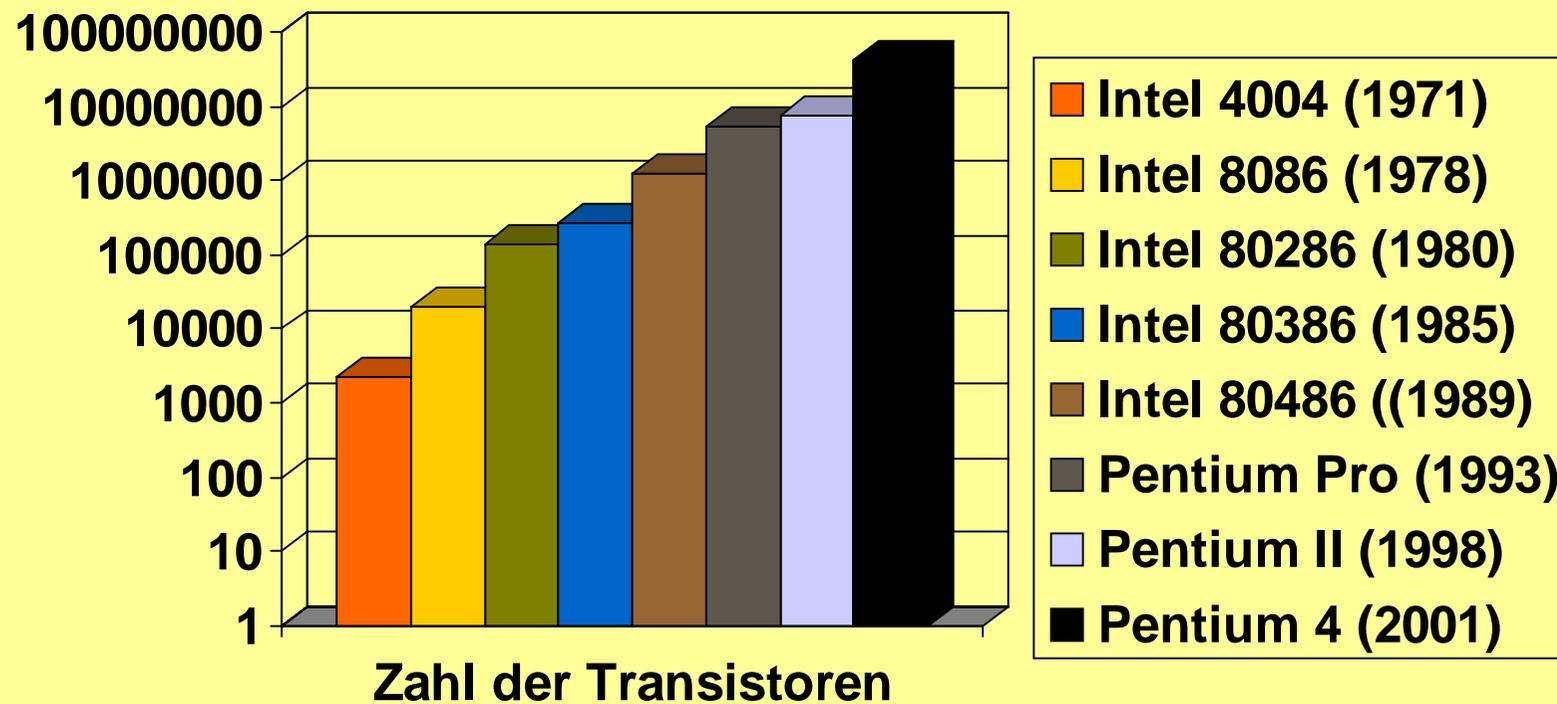
- Verdopplung der Transistor-Dichte alle 18 Monate  
(Gordon Moore, Mitbegründer von Intel, 1965)



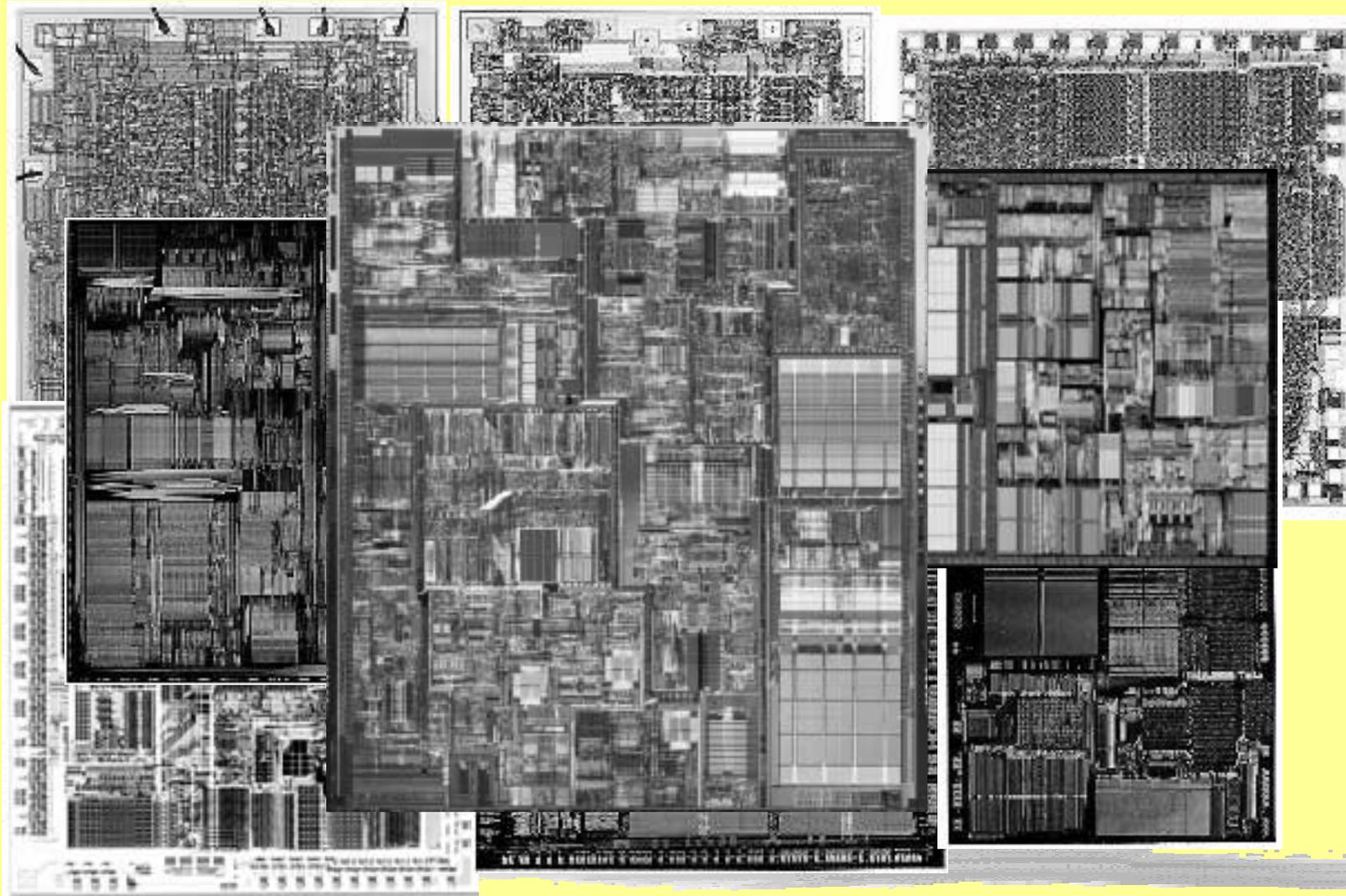
# Moore's Law



# Anstieg der Komplexität



# Anstieg der Komplexität



JR - RA - SS02

Einleitung

1/29

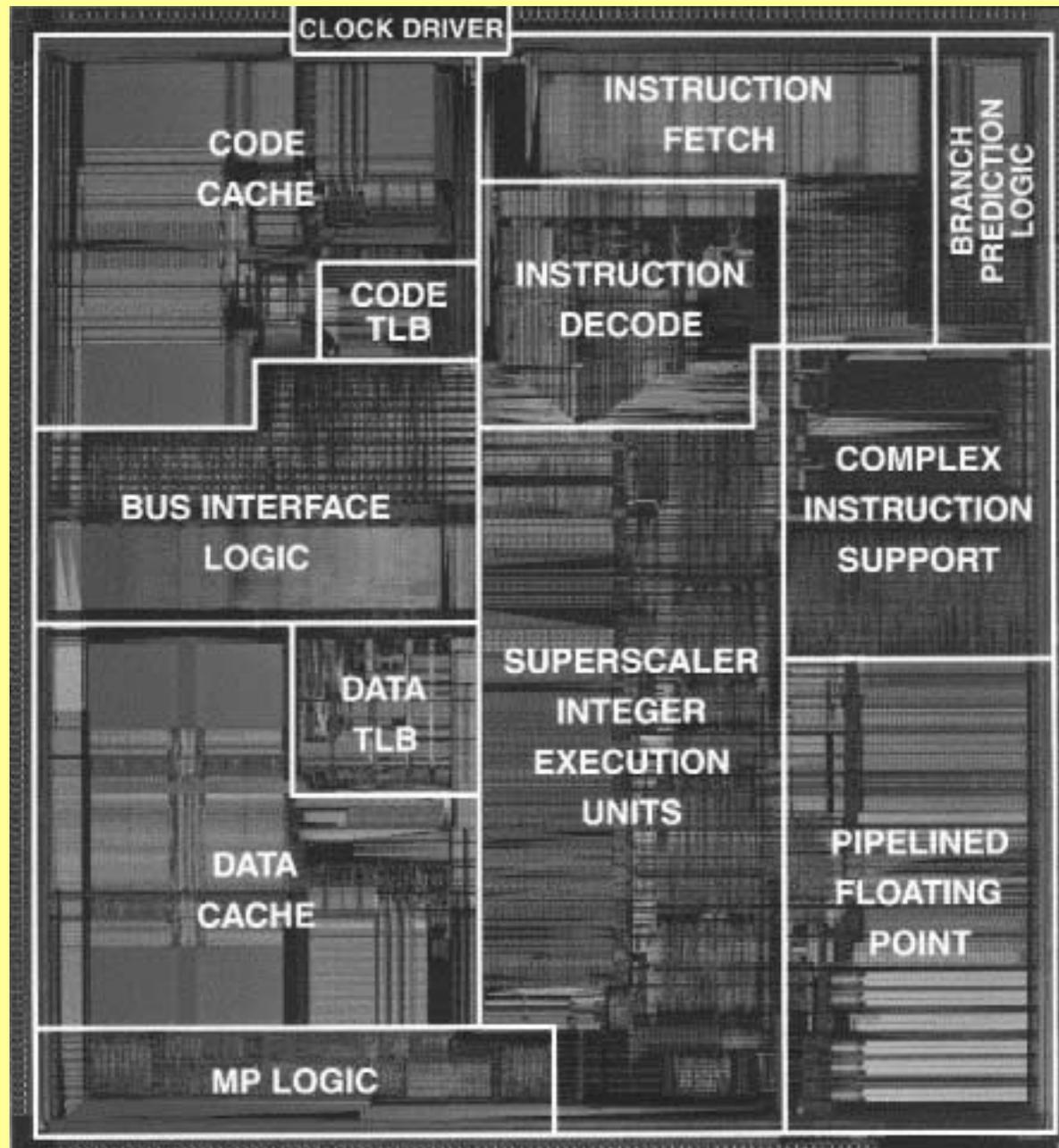
# aber:

- kein Gesetz,

sondern Voraussage,  
was technologisch möglich

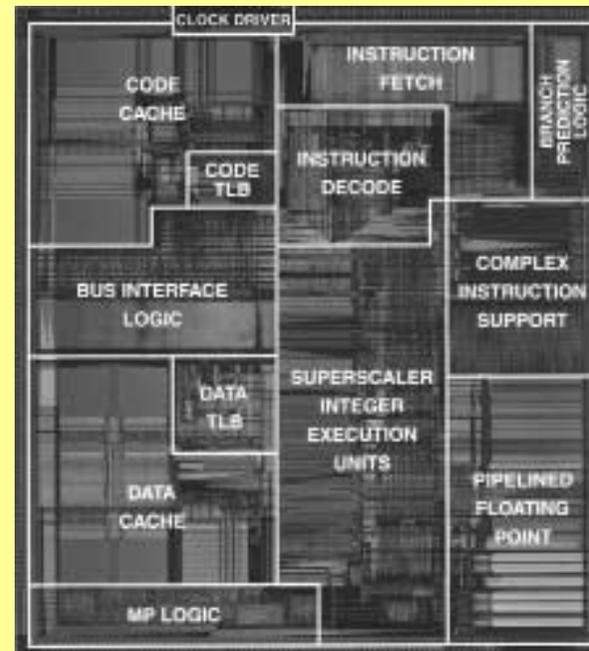
# Vorsicht!

- Mars Pathfinder Mission
- Challenger
- ...
- **Pentium Bug**

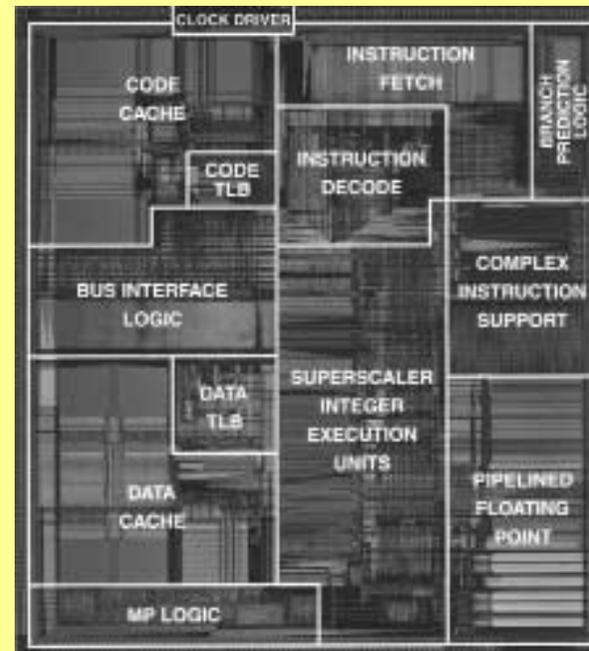


# Pentium Bug

- Fehler in FP-Unit
- Imageverlust
- 475 Millionen \$



# Pentium Bug

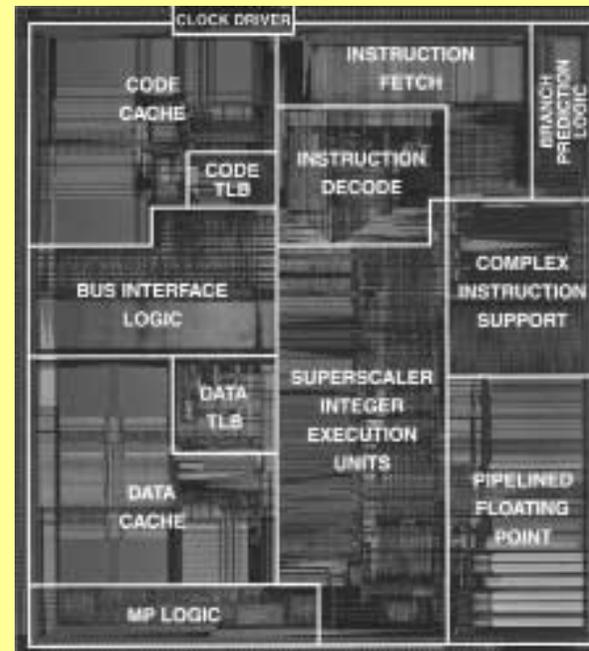


# Pentium Bug

$$x = 4195835$$

$$y = 3145727$$

$$z = x - (x/y)*y$$

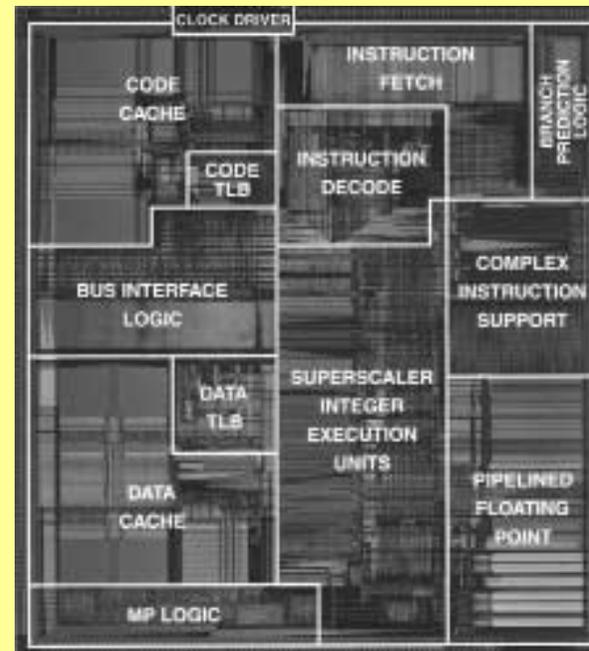


# Pentium Bug

$$x = 4195835$$

$$y = 3145727$$

$$z = x - (x/y)*y$$

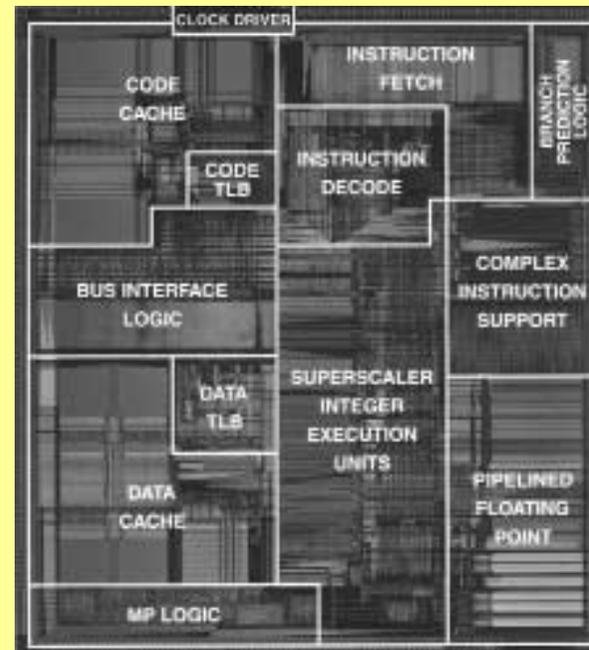


# Pentium Bug

$$x = 4195835$$

$$y = 3145727$$

$$z = x - (x/y)*y$$



# Pentium Bug

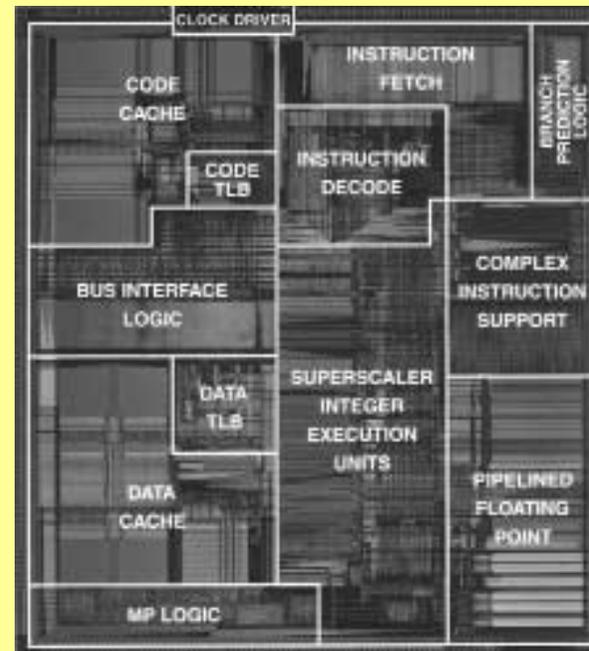
$$x = 4195835$$

$$y = 3145727$$

$$z = x - (x/y)*y$$



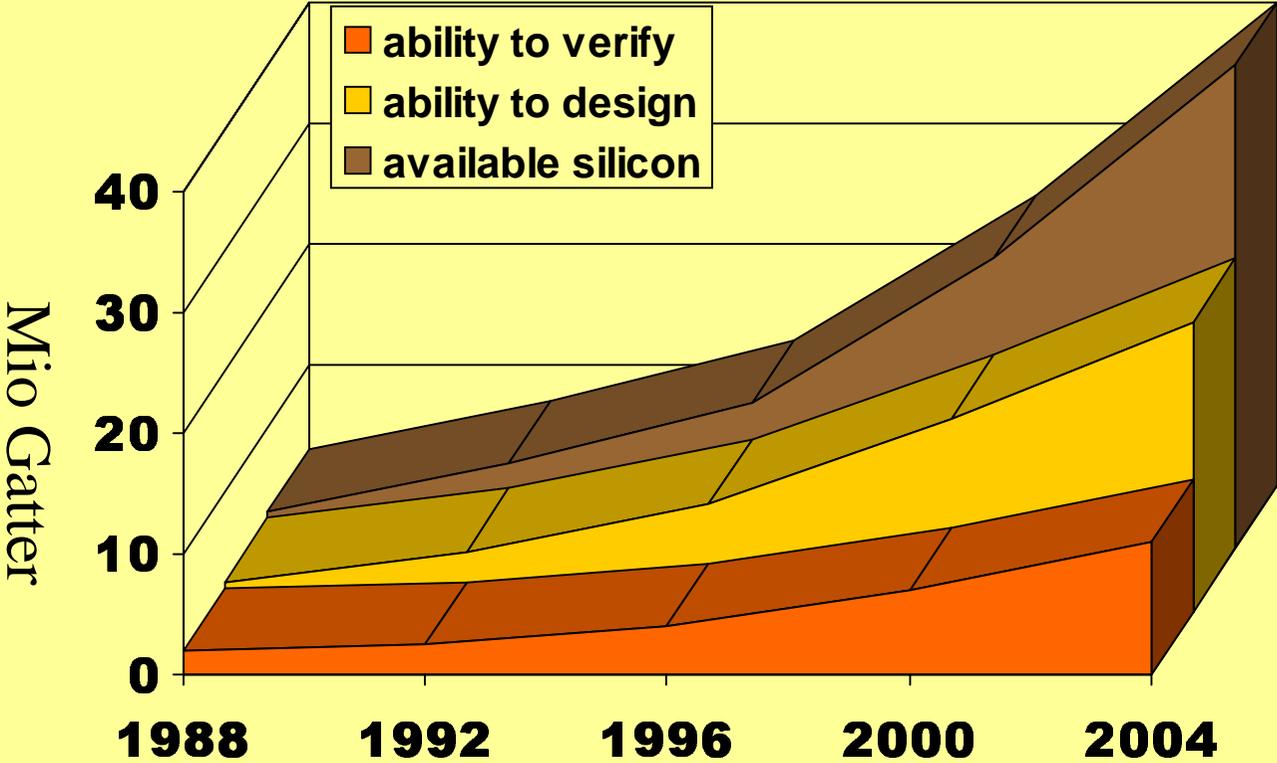
$$z = 256 !!$$



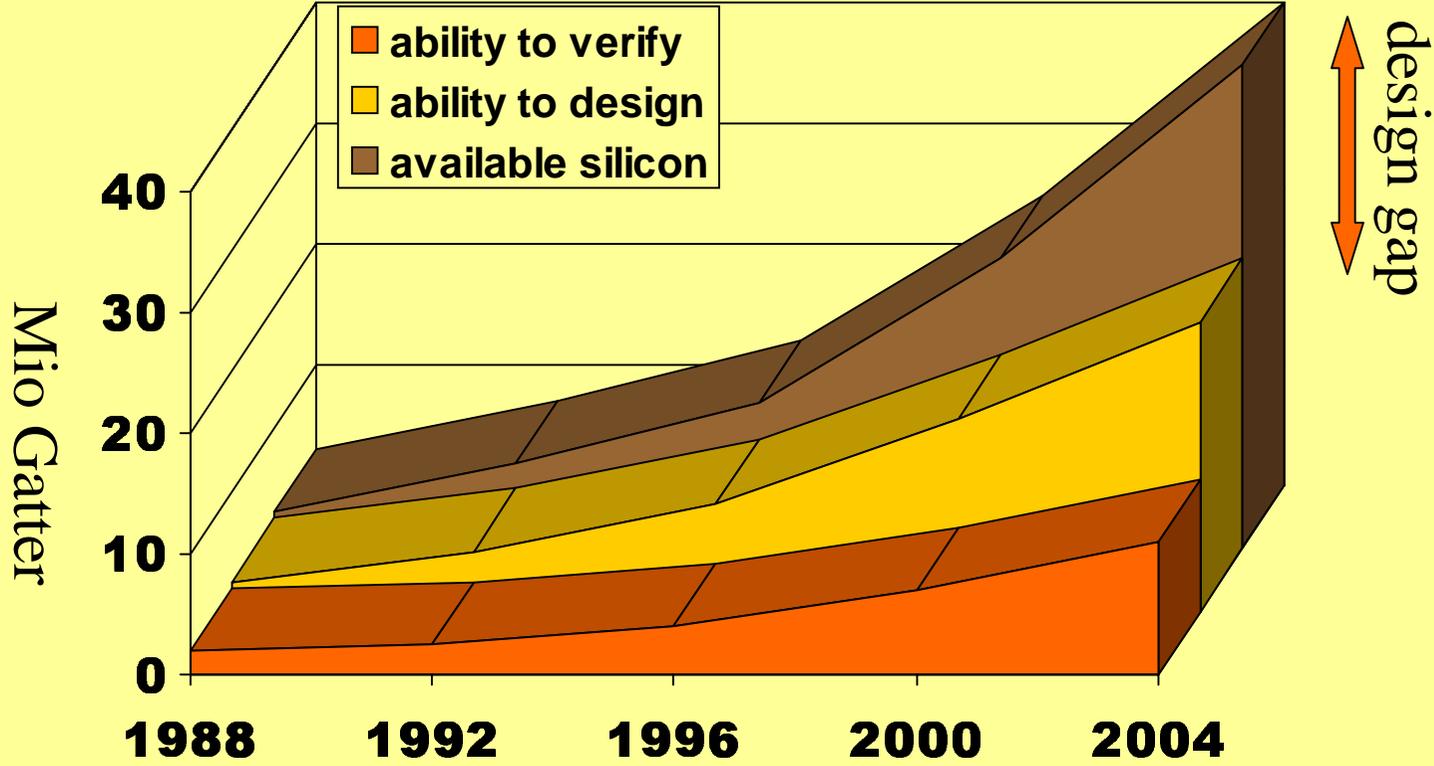
# Designngap

- Die Größe von Systemen übersteigt die Fähigkeiten im Entwurf auf niedrigen Abstraktionsebenen

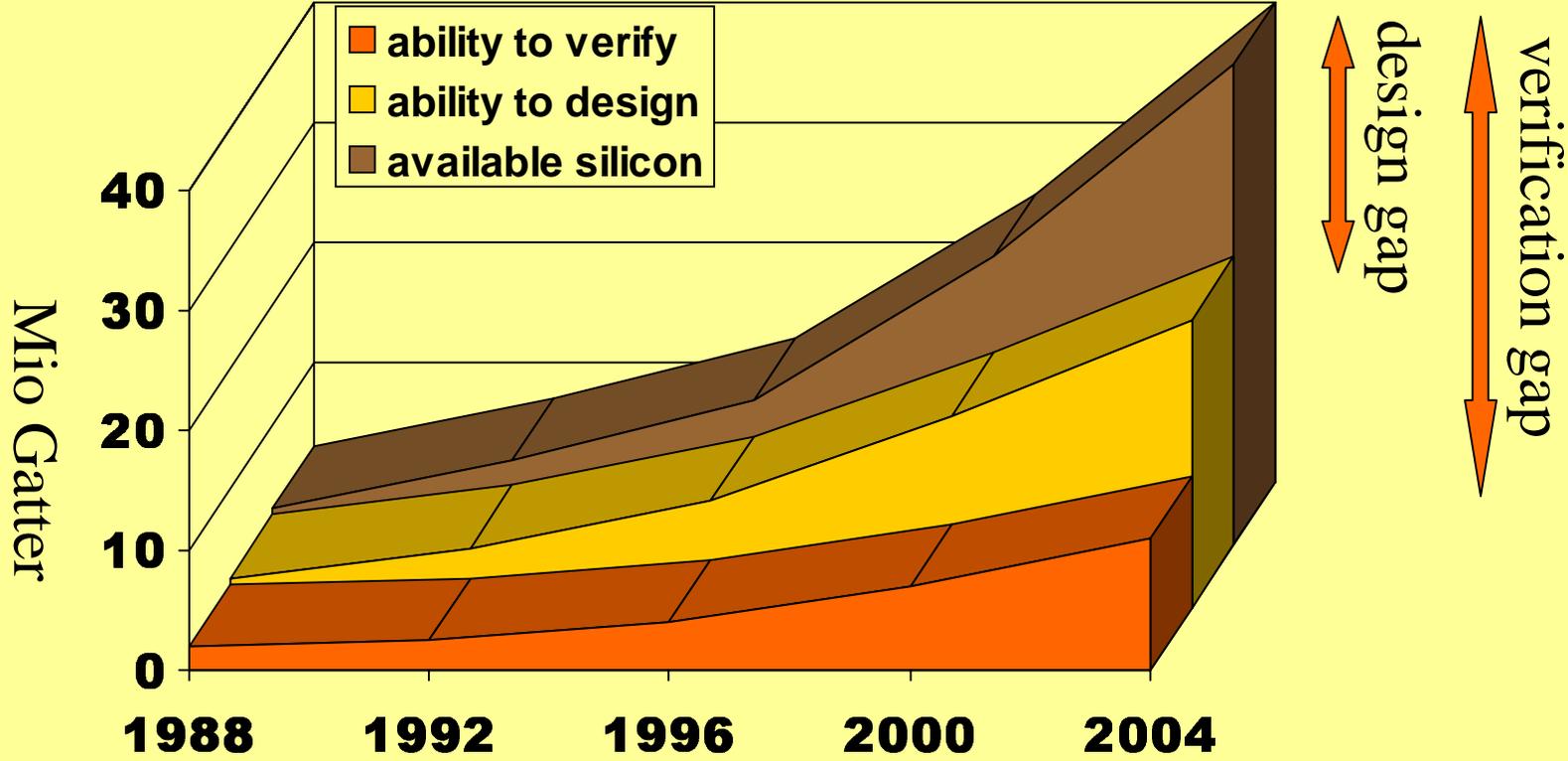
# Design-/Verification Gap



# Design-/Verification Gap



# Design-/Verification Gap







# ■ Was tun mit all den Transistoren?

- Was tun mit all den Transistoren?
- Wie entwerfen?