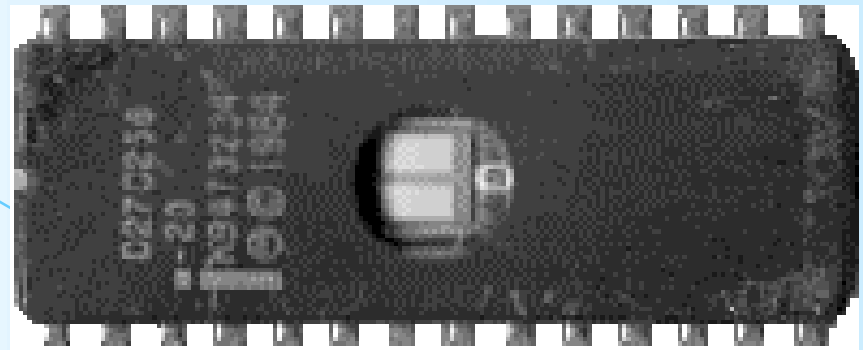
Introduction of Real-Time Embedded System Design

Gang Quan

Chet Kagel
FMTC, Orlando Office

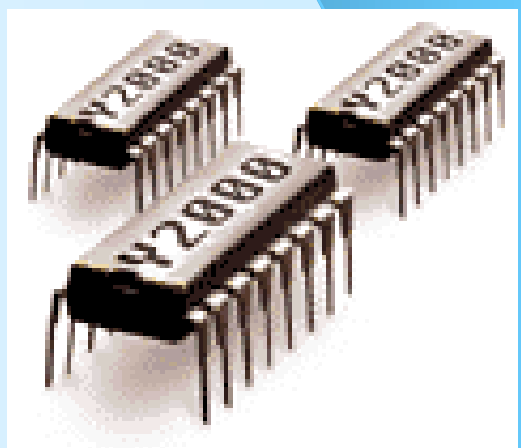
What are Embedded Systems?



- *Def.* - **A microprocessor-based control system which processes a fixed set of programmed instructions to control electromechanical equipment which may be part of an even larger system.**

Embedded Systems Defined

- Refers to either single or multi-purpose computerized devices that are literally embedded within some larger piece of engineering equipment or industrial product.



Examples of Embedded Systems

- Personal Computers (PCs)
- ATMs
- Heating, Cooling and Ventilating Systems
- Security Systems
- Elevators
- Bar Code Equipment
- Real Time Control Systems
- Computer Numeric Controls (CNCs)
- Telephone Exchanges and Switches (PBXs)
- Environmental Monitoring Equipment
- Global Positioning System (GPSs)
- Programmable Logic Controls (PLCs)
- Test Equipment
- Robotics
- Supervisory Control & Data Acquisition
- Systems (SCADAs)

Real-Time Embedded System Design

What ?

Why ?

How ???

What

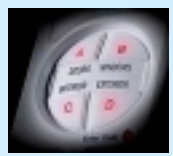
- What is the real-time embedded system?
  - Embedded System
    - Processor based
      - General processors
      - Micro controllers
      - DSP
    - A subsystem
    - Not a general programming computer
  - Real-Time
    - Not only deliver correct results but when these results are delivered

Examples In Your Daily Life

- …wake up …

- …have breakfast …

- …set home safety system …



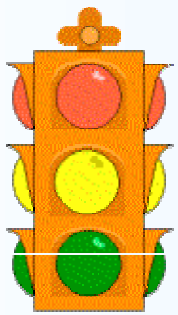
Examples In Your Daily Life (cont')

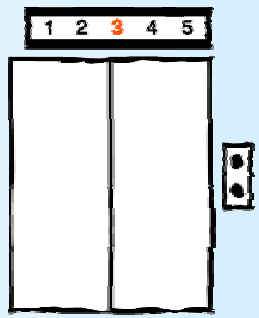
- …get into your car …

- A late model car can have as many as 65+ processors for engine control, A/C control, cruise control, ABS, audio, etc
- More than 30% of the cost of a car is now in electronics
- 90% of all innovations will be based on electronic systems

Examples In Your Daily Life (cont')

- …on your way to your office…





Examples In Your Daily Life (cont')

- …in your office …







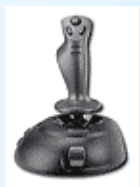
Examples In Your Daily Life (cont')

- …back home …










Several hundred processors can be involved in the course of one day for one person !

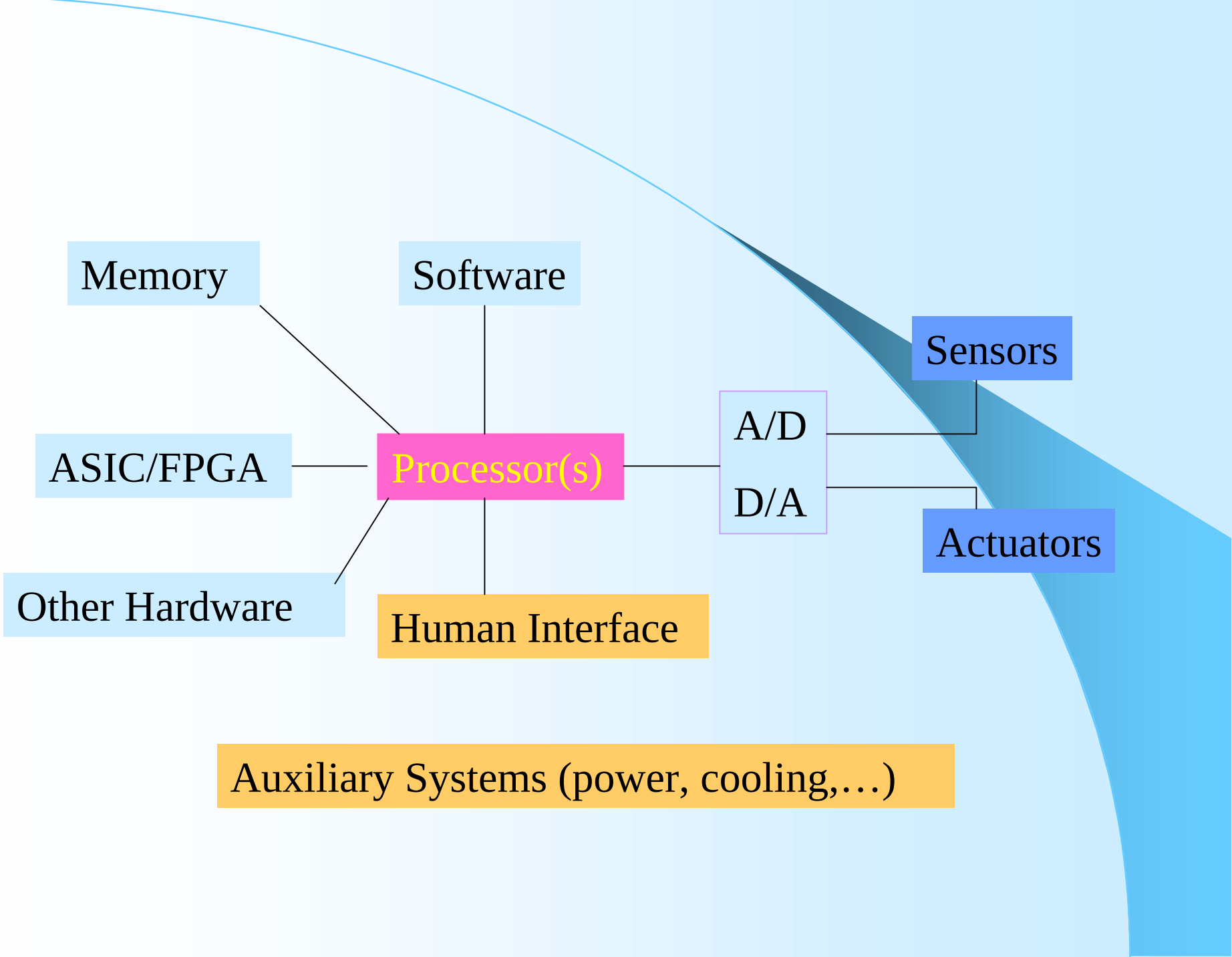
Other Examples

- Mission critical controls
  - Nuclear plant control, aircraft navigation ,military equipment
- Medical equipment
- Communication
- Toy, etc

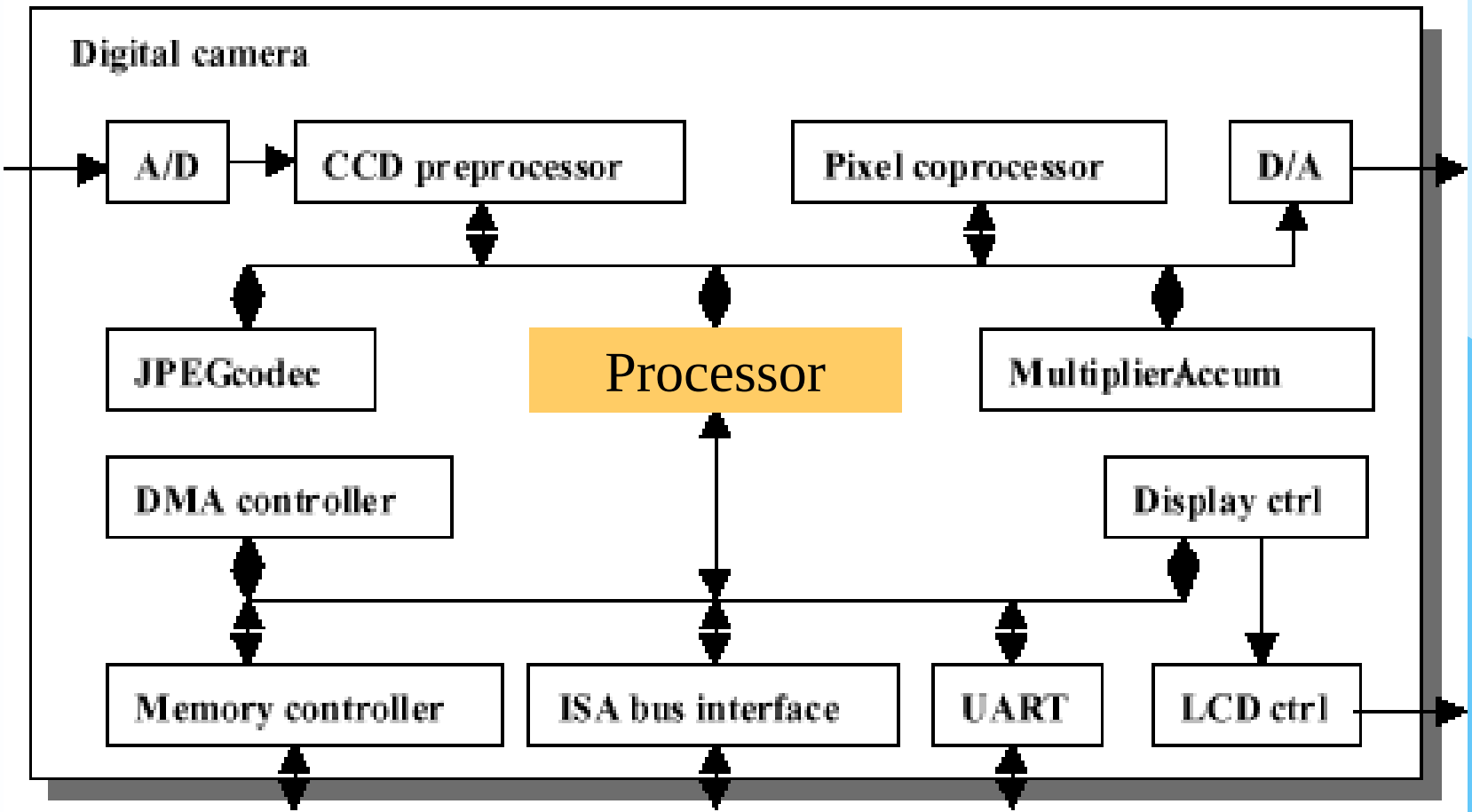
Real-time embedded systems have been deeply ingrained in our life

What (cont'd)

- What are in the real-time embedded system?


Memory
Software
Sensors
ASIC/FPGA
Processor(s)
A/D
D/A
Actuators
Other Hardware
Human Interface
Auxiliary Systems (power, cooling,…)

Example: Digital Camera



Why

- Why using the processor(s) in the real-time embedded systems?
  - Flexibility
  - Easy to upgrade
  - Easy to build complex system behavior
  - Maintainability

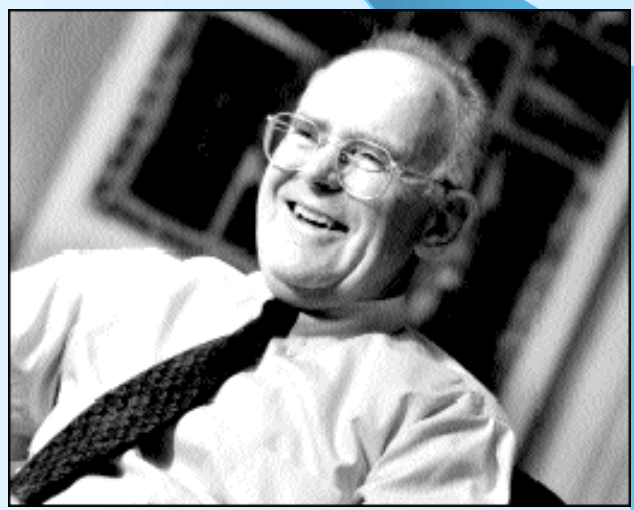
Why

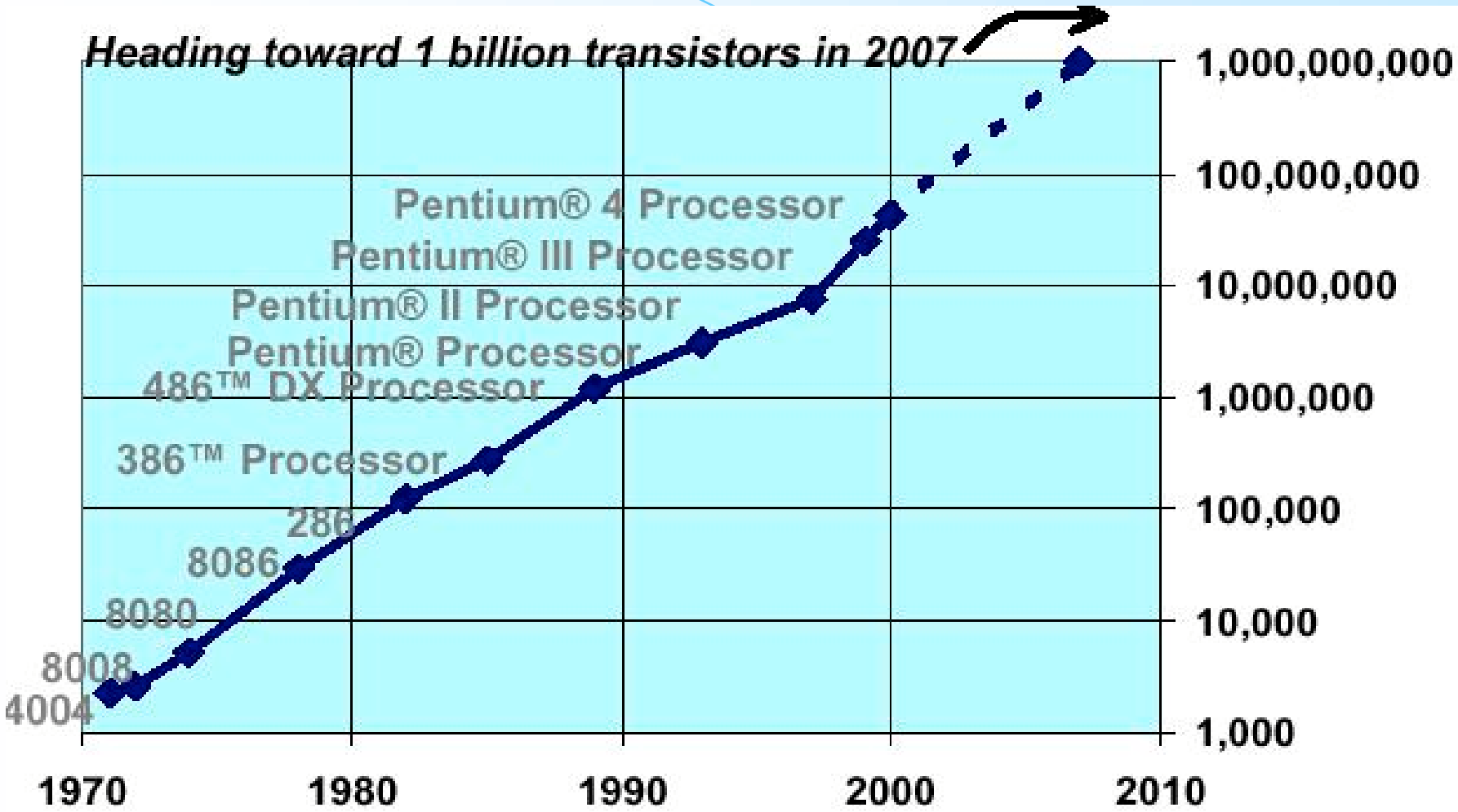
- Why is it so hard to design the real-time embedded system?
  - Moore's Law
    - Productivity Gap
  - More complex functionality and extreme diversity
  - Design Cost
  - Stringent Time-to-market
  - Design requirements (constraints)

Moore's Law

The transistor density of semiconductor chips would double roughly every 18 months.

--by Gordon Moore,1965
(co-founder of Intel)




Heading toward 1 billion transistors in 2007
Pentium® 4 Processor
Pentium® III Processor
Pentium® II Processor
Pentium® Processor
486™ DX Processor
386™ Processor
286
8086
8080
8008
4004
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
1970
1980
1990
2000
2010

The Future of Moore's Law

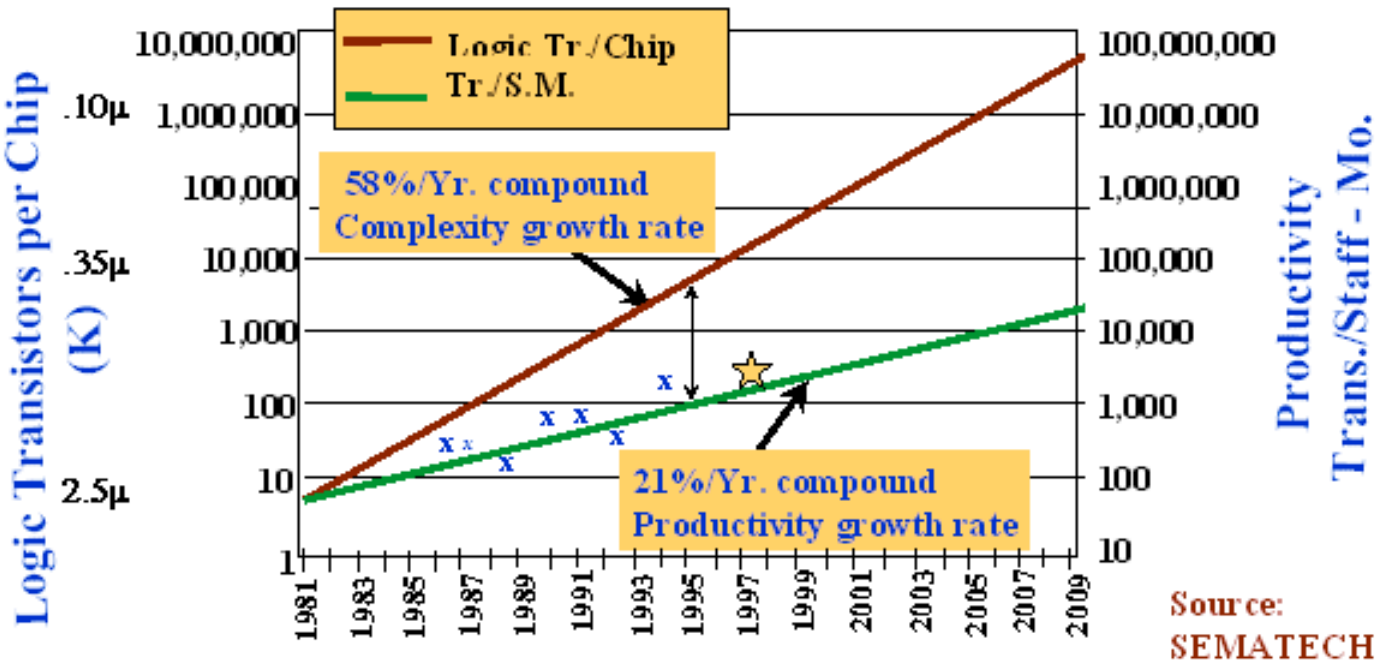
- The deep submicron technology will make most of the current IC technologies obsolete
  - Signal integrity, power consumption, etc
- However …
  - Intel announced its Terahertz Transistor Architecture will allow the continuation of Moore's Law
    - Push the IC feature size to as small as 15nm
  - Science's Top Ten: nanoscale computing circuits named the top scientific achievement of 2001
    - Each transistor can be as small as one or several molecules

Moore's law will still be true and continue to drive the development of IC technology !!

Productivity Gap

The gap between the availability of the IC technology (increasing computing power) and the application of the IC technology.

Crisis: Productivity Gap



Why (cont'd)

- Why is it so hard to design the real-time embedded system?
  - Moore's Law
    - Productivity Gap
  - More complex functionality and extreme diversity
  - Design cost
    - Reduce non-recurring engineering (NRE) cost
    - A superior human engineer may outperform the CAD tools in designing simple embedded systems but not for systems with hundred millions to billions gates
  - Stringent time-to-market
  - Design requirements (constraints)

Silicon Technology

	1997	1998	1999	2002
Technology	350nm	250nm	180nm	130nm
Cost	$1.5-2.0billion	$2-3billion	$3-4billion	$4+ billion
Design cycle	18-12mo	12-10mo	10-8mo	8-6mo
Complexity	200-500k	1-2M	4-6M	10-25M
Application	Cellar phone, DVDs	Wireless PDAs	Internet appliances	Ubiquitous computing

Why

- Why is it so hard to design the real-time embedded system?
  - Moore's Law
    - Productivity Gap
  - More complex functionality and extreme diversity
  - Design Cost
  - Stringent Time-to-market
  - Design requirements (constraints)

Design Requirements (Constraints)

- Timing
- Size&Weight
- Safety & Reliability
  - Low cost reliability with minimal redundancy
- Cost sensitivity
- Power consumption
- Others: component acquisition, upgrades, compatibility, etc.

How

- Not an easy answer, in general:
  - Increase the abstraction level
    - system level design is the key
  - Design reuse

What's Next

- Introduction
  - System level design, IP reuse, platform based design, real-time operating system
- Computation modeling
  - DFG, CDFG, FSM, Petri Net, etc
- Optimization Methods
- System Partitioning
- System Scheduling