



# Sisteme Incorporate

Cursul 1  
Introducere

# Sisteme Incorporate

Ce au in comun o masina de spalat si un Boeing 747?



Amandoua sunt sisteme incorporate.

# Sisteme Incorporate

- Definitie
  - Un sistem de calcul incorporat in dispozitivele electronice.
- Sistemul implementeaza o anumita functionalitate
  - Este o combinatie de hardware si software proiectata sa indeplineasca o functie anume.

# Vedere de ansamblu

- Sisteme de calcul incorporate
  - Greu de definit. Aproape toate sistemele diferite de un calculator desktop obisnuit
  - Miliarde de unitati produse anual vs. milioane de unitati pentru sistemele desktop
  - In jur de 50 sisteme intr-o locuinta sau intr-un automobil obisnuit
  - Pret semnificativ mai mic decat al unui sistem desktop
  - Putere de calcul redusa ( nu in toate cazurile)

# Exemple Obisnuite

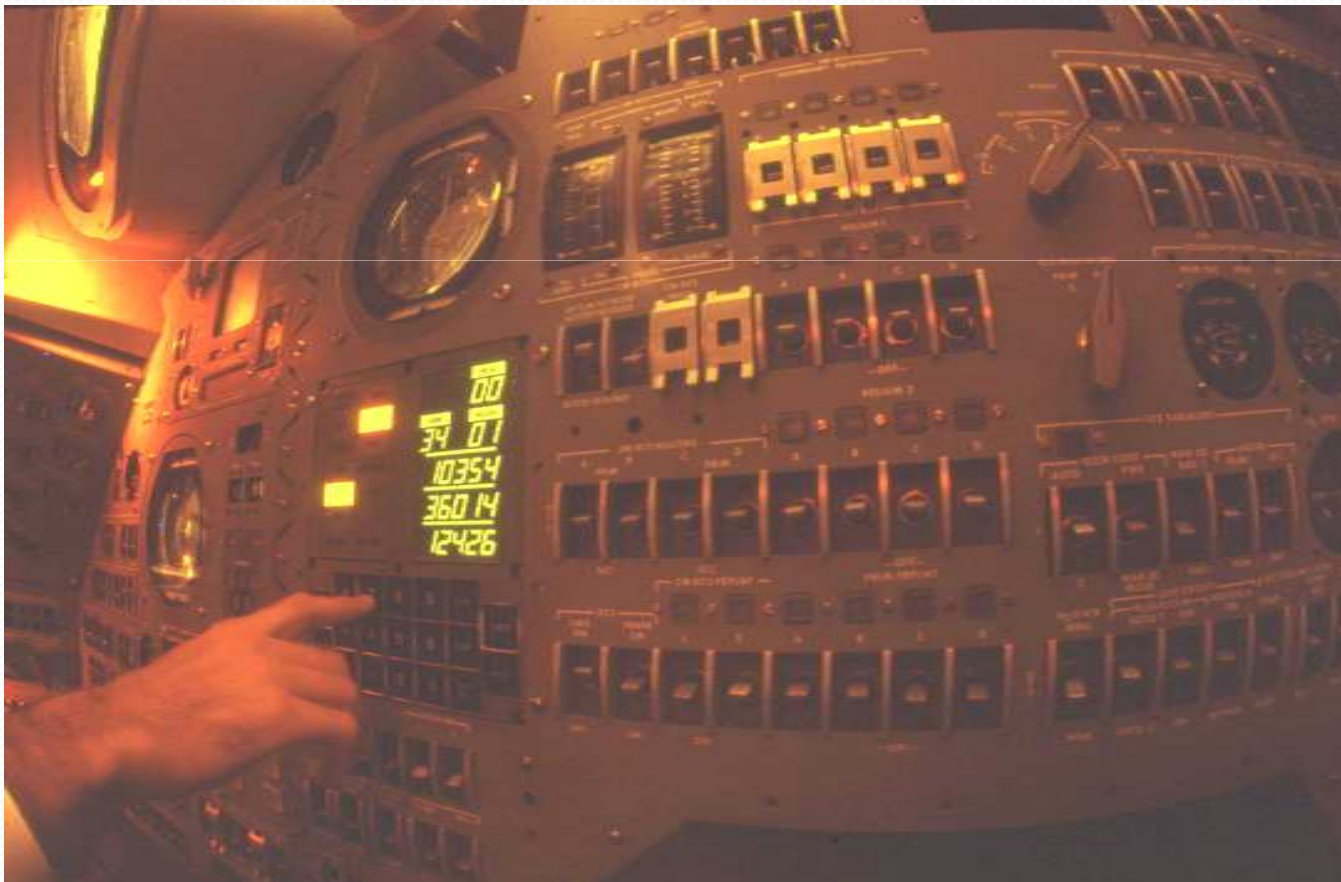


# Mai multe exemple

- Sisteme de procesare semnal
  - radar, sonar, real-time video, playere MP3/CD/DVD, echipament medical
- Sisteme critice
  - avionica, echipamente de control al navelor spatiale, controlul reactoarelor nucleare
- Sisteme de control distribuit
  - Routere si switch-uri de retea, sisteme de transport in masa
- Sisteme “mici”
  - Telefoane celulare, electrocasnice, jucarii, carduri, playere MP3, PDA-uri, camere digitale, senzori, ceasuri digitale, receptoare GPS

# Primul Sistem Embedded

AGC – construit de MIT Instrumentation Laboratory ('60)





# Apollo Guidance Computer

20 Iulie 1969 – Primul Computer care a ajuns pe Luna

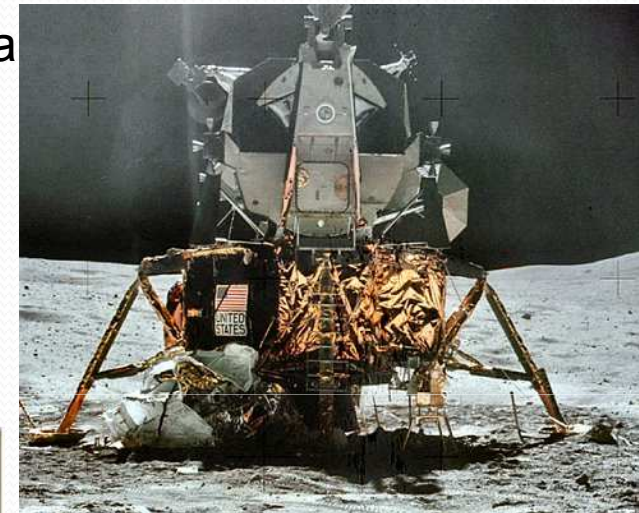
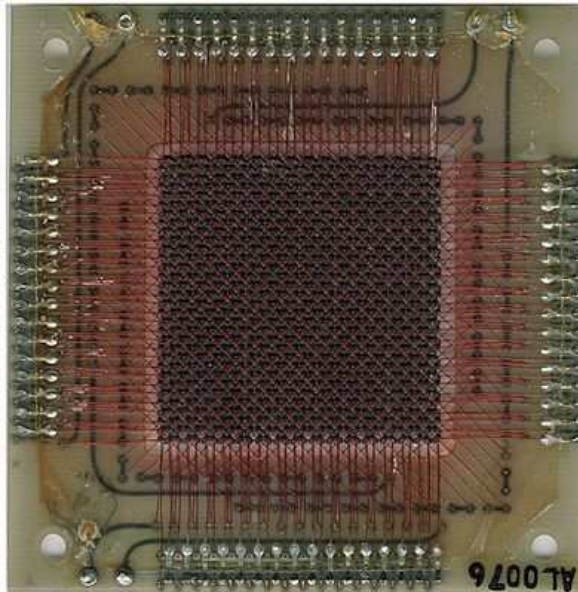
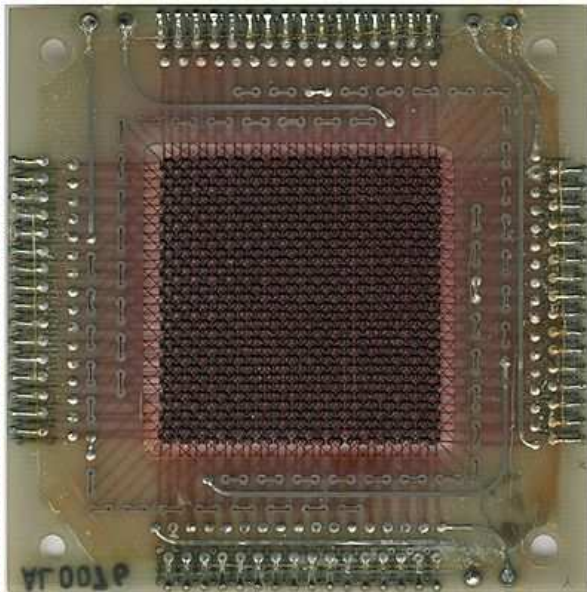
## Specificatii:

Frecventa de ceas: 1,024 MHz

Memorie RAM: 8KB memorie cu miez de ferita

Memorie program: 64KB ROM Core Rope

Primul sistem de calcul care a folosit circuite integrate





# Caracteristicile unui sistem embedded

- Are o singura functie
  - Executa un singur program, repetat
- Supus la constrangeri
  - Cost redus, consum mic de electricitate, dimensiuni, viteza de executie, etc.
- Reactioneaza la evenimentele de timp real
  - Reactioneaza la schimbarile din mediul inconjurator
  - Trebuie sa obtina rezultate in timp real cu un minim de intarzieri

# De ce ne pasa? Cateva cifre de piata

- Echipamentele embedded inlocuiesc PC-urile
  - Au diferite forme si functii specializate fata de generalismul unui PC.
  - Anul 2002: mai mult de 50% din dispozitivele de acces la Internet NU sunt PC-uri.
    - In 1997, 96% din dispozitivele de acces la Internet erau PC-uri
- Sistemele traditionale depind din ce in ce mai mult de sisteme de calcul
  - Automobilele moderne contin aproape 100 de procesoare care ruleaza software complex
    - Reglarea combustiei, controlul noxelor emise, sistem anti-derapare, monitorizarea consumului, transmisia automata, gestionarea afisajului de bord etc.
    - <http://www.howstuffworks.com/car-computer.htm>

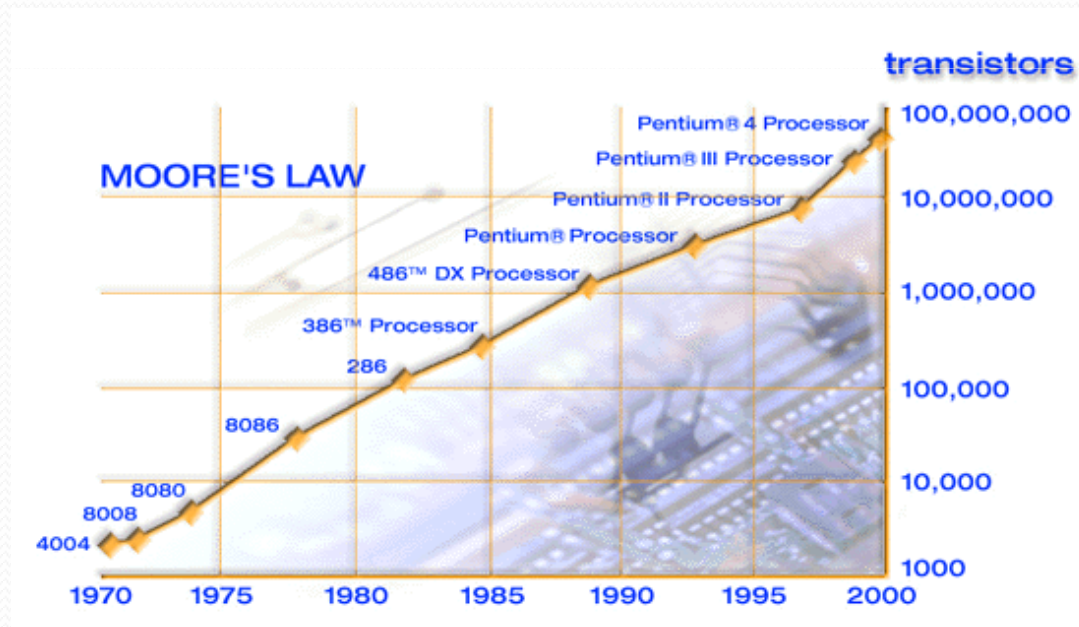


# Lucrurile privite din perspectiva CS

- In trecut stiinta calculatoarelor nu era interesata de domeniul embedded
  - Cantitate limitata de software
  - Programare la nivel de baza
- In prezent exista un interes in crestere pentru cercetare si software development in CS
  - Cresterea complexitatii sistemelor embedded necesita sistematizarea proiectarii

# Factorul decisiv in cresterea complexitatii

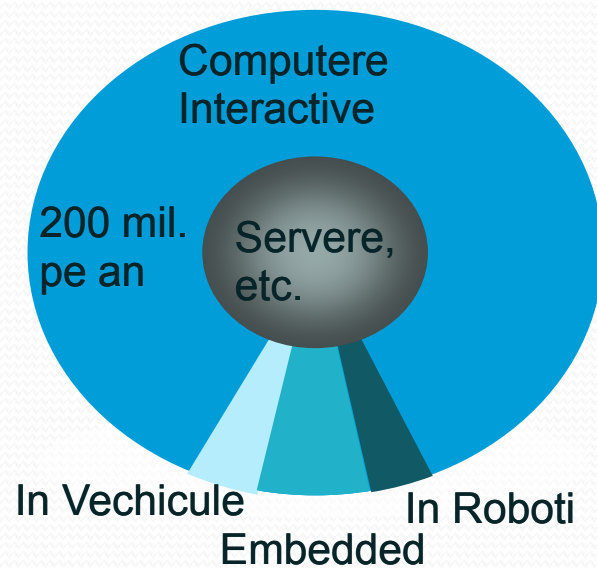
- Legea lui Moore: Dublarea numarului de tranzitoare dintr-un chip odata la 18 luni
  - Permite inlocuirea sistemelor analogice complexe cu sisteme digitale (ex: camera foto digitala)



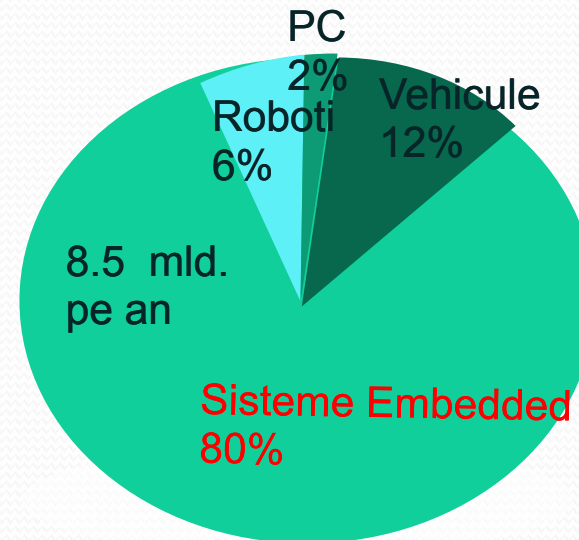
# Unde gasim procesoarele?

**98% din cele 8 miliarde de procesoare produse in 2000 au fost folosite pentru aplicatii embedded**

Punctele de interes ale CS



Unde sunt procesoarele?



Sursa: DARPA/Intel (Tennenhouse)



# Caracteristici tipice ale S. I.

- Parte a unui sistem mai mare
  - Aproape niciodata un “calculator cu ecran, tastatura etc.”
- Hardware si Software dedicat, nu universal
  - Aplicatia pt care e proiectat este stiuta dinainte
  - Dezvoltarea aplicatiei se face concurent
- Trebuie sa fie reprogramabil
  - Flexibilitate la upgrade, repararea erorilor, particularizare in functie de utilizator

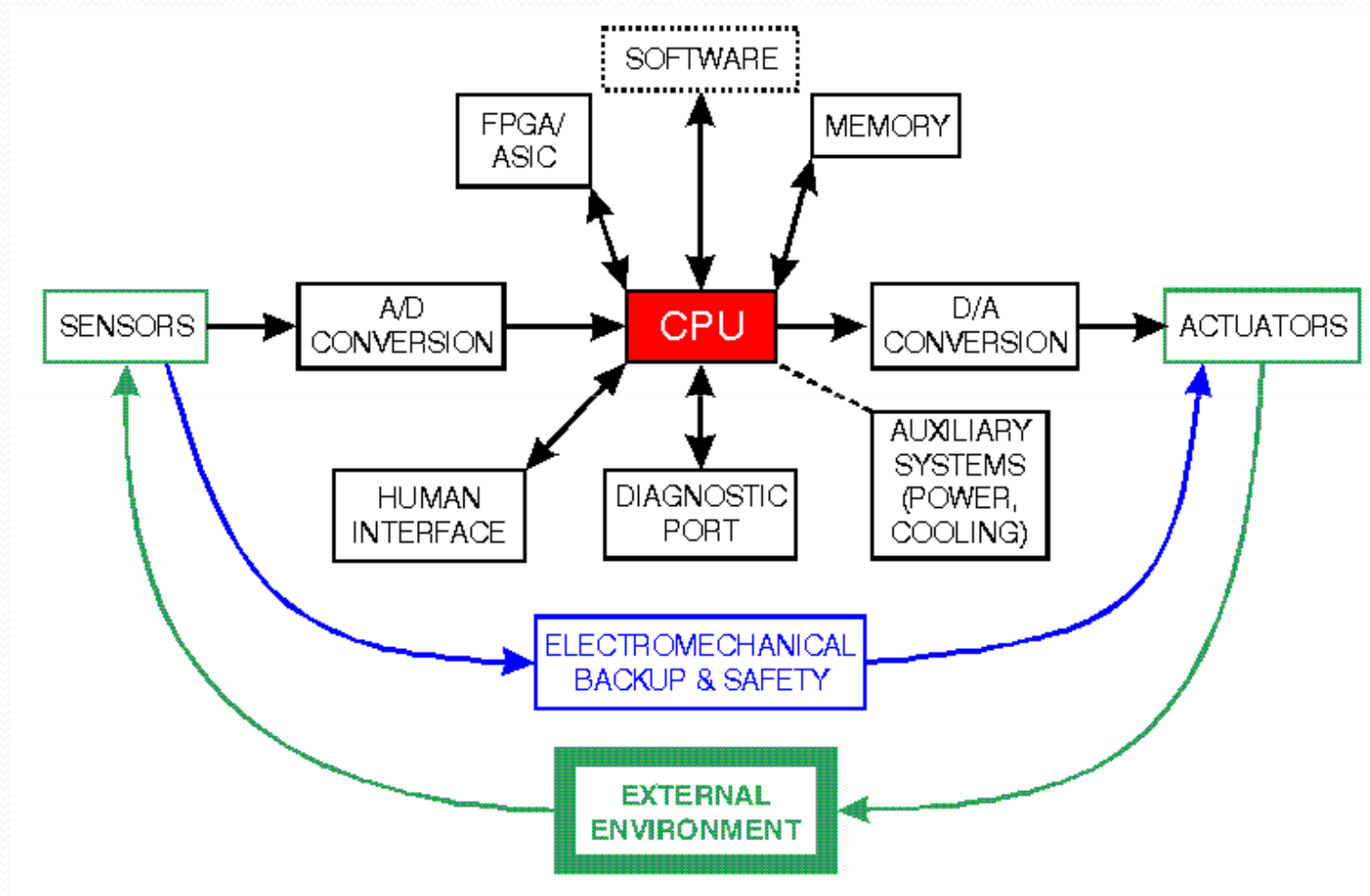
## Caracteristici tipice ale S. I. (2)

- Interactioneaza (simte, comunica, manipuleaza) cu lumea exterioara
- Nu se opreste niciodata din executie (caz ideal)
- Timpul de operare e constrans de productivitate si latentă in executie.
- Alte constrangeri: consum de energie, dimensiuni, greutate, temperatura, fiabilitate etc.
- Performanta crescuta si capacitate de conectare la Internet.

# Tendinte Recente

- Cerinte crescute de calcul
  - Procesare multimedia, HDTV
- Conectare in retea
  - Pentru monitorizare si depanare de la distanta.
  - Server Web embedded
    - e.g. Axis camera <http://neteye.nesl.ucla.edu>
    - e.g. Mercedes car with web server
  - Masini Java embedded
    - e.g. Java ring, smart card, imprimante
  - Camere foto/video care se conecteaza direct la retea
- Nevoie crescuta de flexibilitate
  - Timpul necesar dezvoltarii unui produs nou scade tot mai mult

# Sisteme Embedded Software Traditionale



# Sisteme Embedded Hardware Traditionale (ASIC)

## Application-Specific Integrated Circuit

### Caracteristici

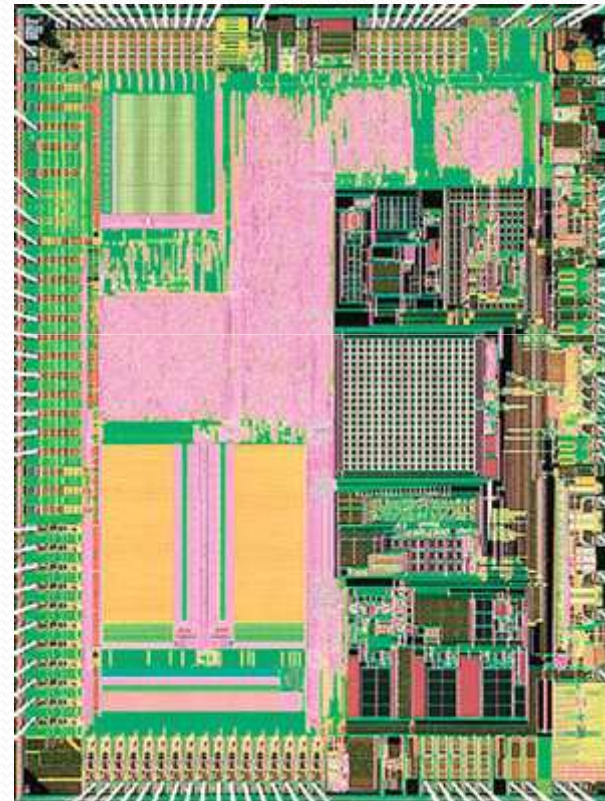
Suprafata: 4.6 mm x 5.1 mm

Frecventa: 20 MHz

Tehnologie: HP 0.5  $\mu\text{m}$

Consum: 16 mW - 120 mW @ 20 MHz, 3.3 V

Timp mediu de achizitie: 10  $\mu\text{s}$  to 300  $\mu\text{s}$

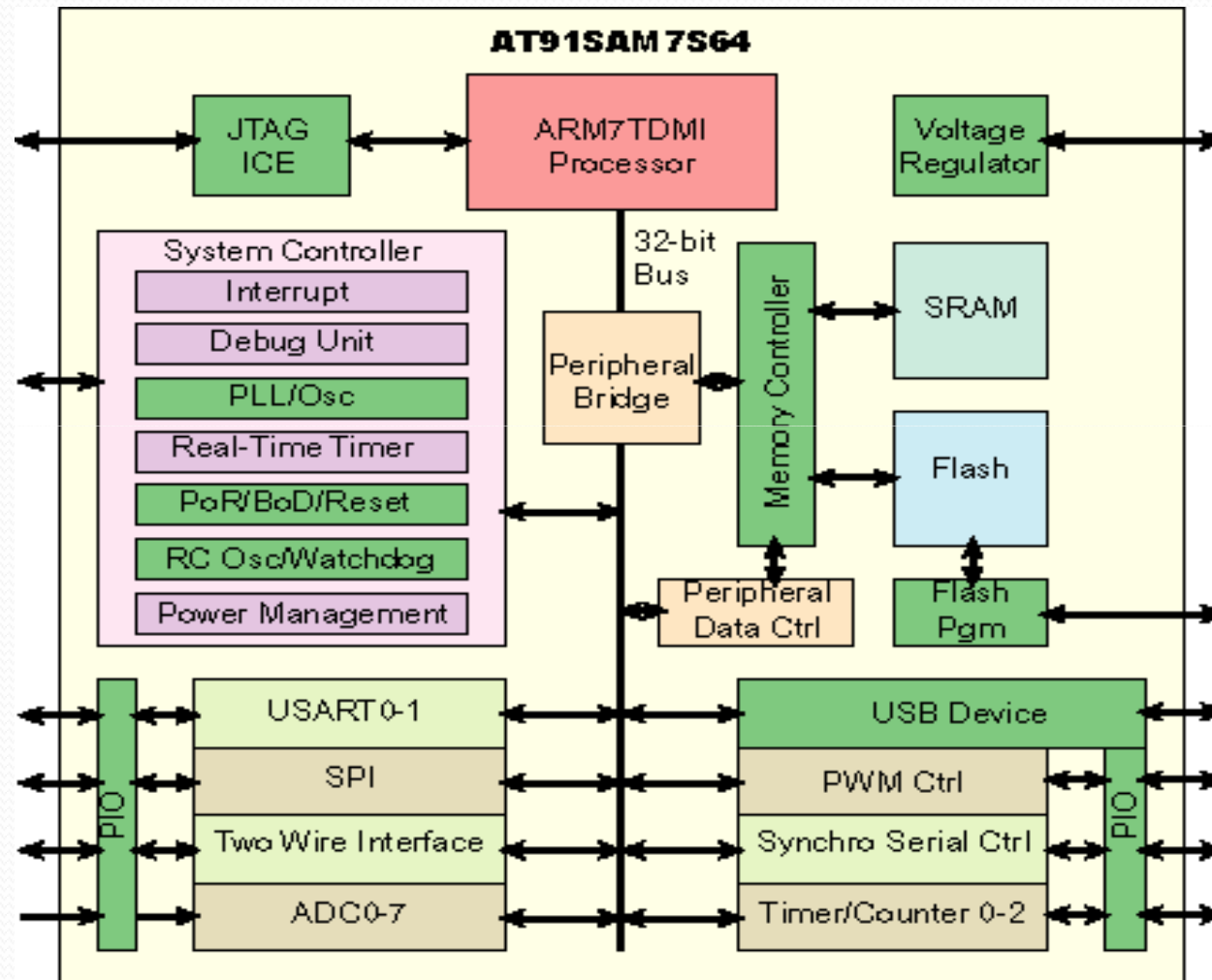




# System-on-Chip (SoC)

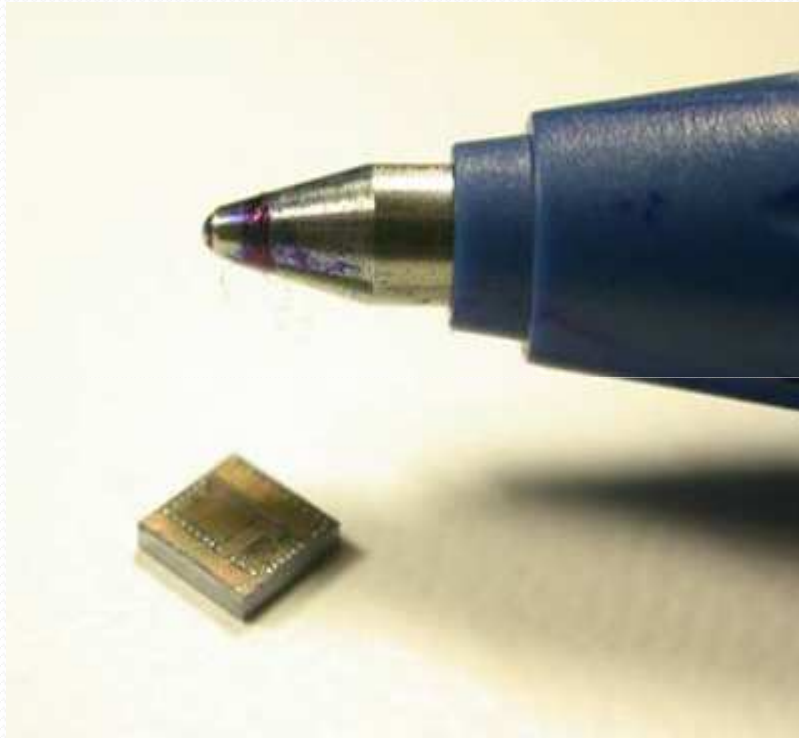
- Definitie: un chip care contine toate circuitele electronice necesare pentru un sistem complet. SoC au in compozitie memorie(RAM si ROM), microprocesorul, interfete periferice, interfete I/O, convertoare si alte componente necesare sistemului in cauza.
- SoC este in concordanta cu legea lui Moore.

# Exemplu de System-on-chip (SoC)



Atmel's \$3 SoC

# Mai multe exemple

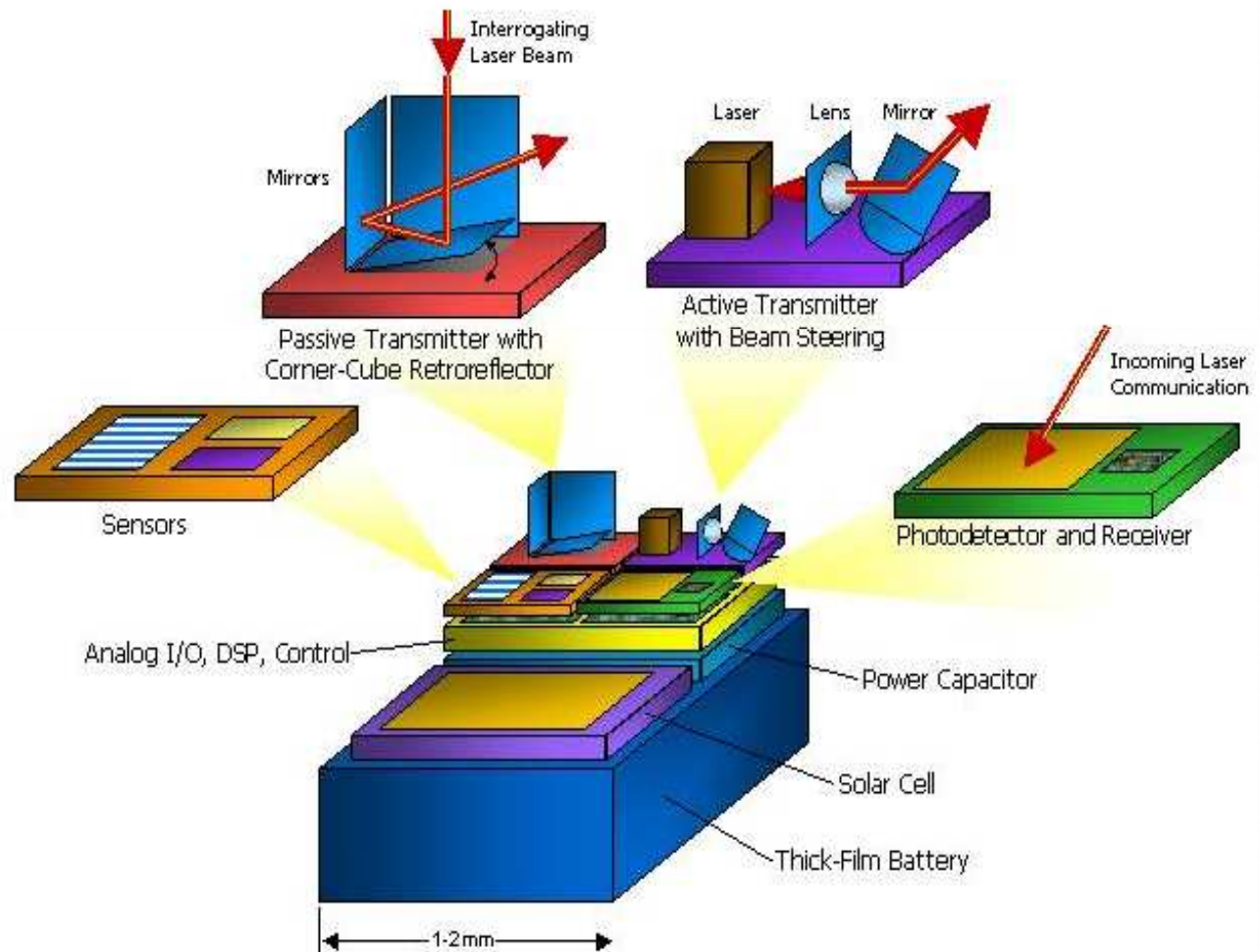
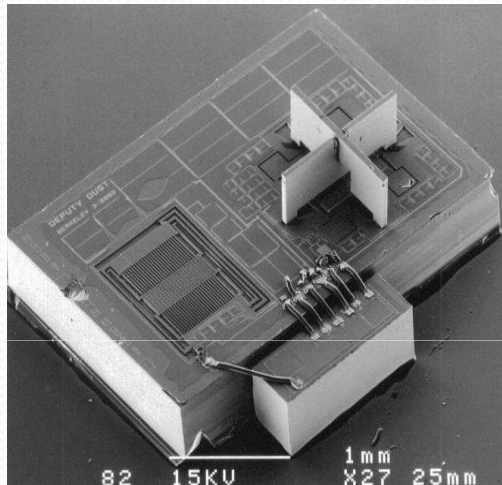


Spec Mote (Berkeley)



Solar-power Wireless Sensor (Berkeley)

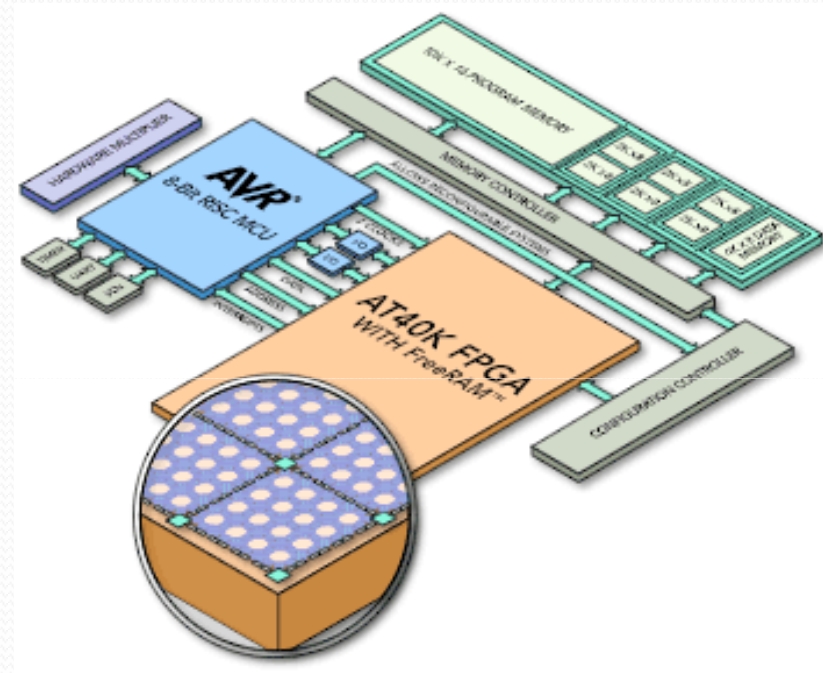
# SoC + mecatronica: Berkeley Smart Dust



# SoC Reconfigurabil

# FPSLIC: AVR cu FPGA

- 40,000 de porti logice  
AT40K FPGA
- 36K Bytes SRAM,
- 30 MIPS 8-bit AVR RISC  
microcontroller core
- Periferice





# Platform SoC

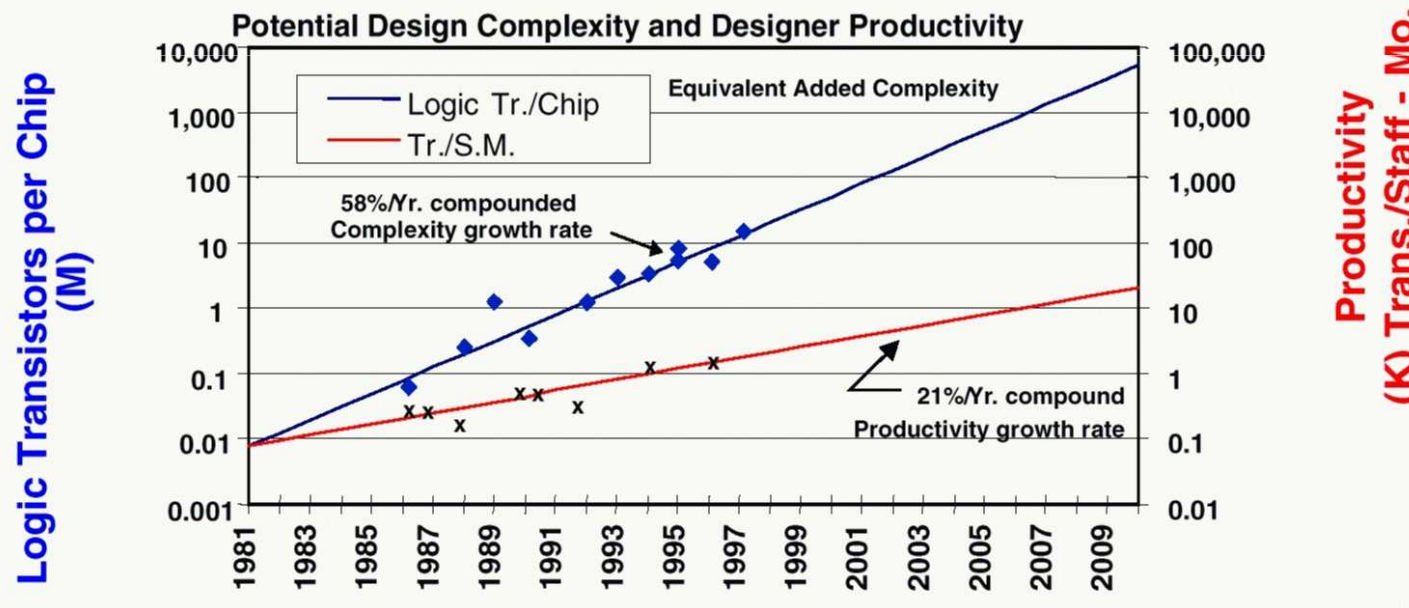
- Unele portiuni din chip sunt predefinite pentru un anumit domeniu de aplicatie
  - Procesor, blocuri periferice proprietate intelectuala (IP), memorie sau magistrala de un anumit tip, sistem de operare de timp real etc.
- Particularizarea designului prin adaugarea de module hardware IP sau software incorporat
- De ce exista?
  - Cresterea diferentei de productivitate
  - Pretul ridicat de fabricatie al ASIC

“+” Ciclu de design rapid din cauza folosirii unor subsisteme deja existente

“-” Flexibilitate redusa

# Diferenta de Productivitate

- Cresterea diferentei dintre ce poate tehnologia sa ofere si capacitatea noastra de-a proiecta sisteme care sa o foloseasca.



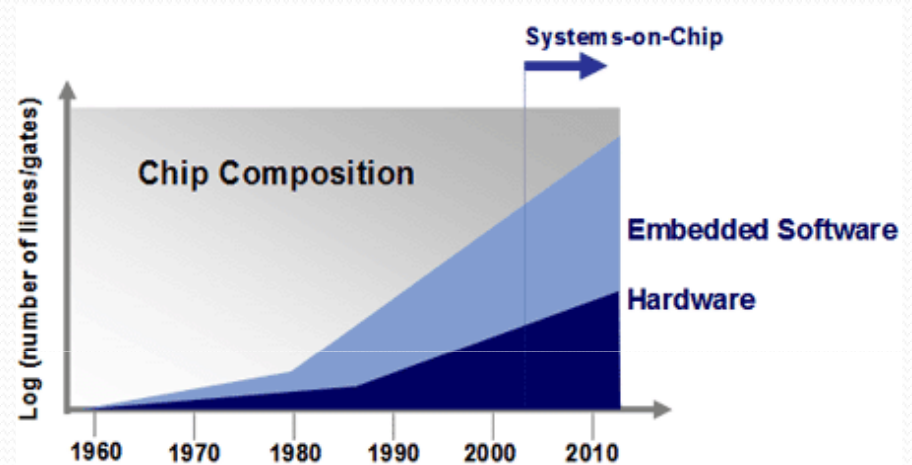
# Migrarea de la hardware la software embedded

Piata pentru software embedded creste cu 16% pe an.

- 1.6 miliarde \$ in 2004
- 3.5 miliarde in 2009

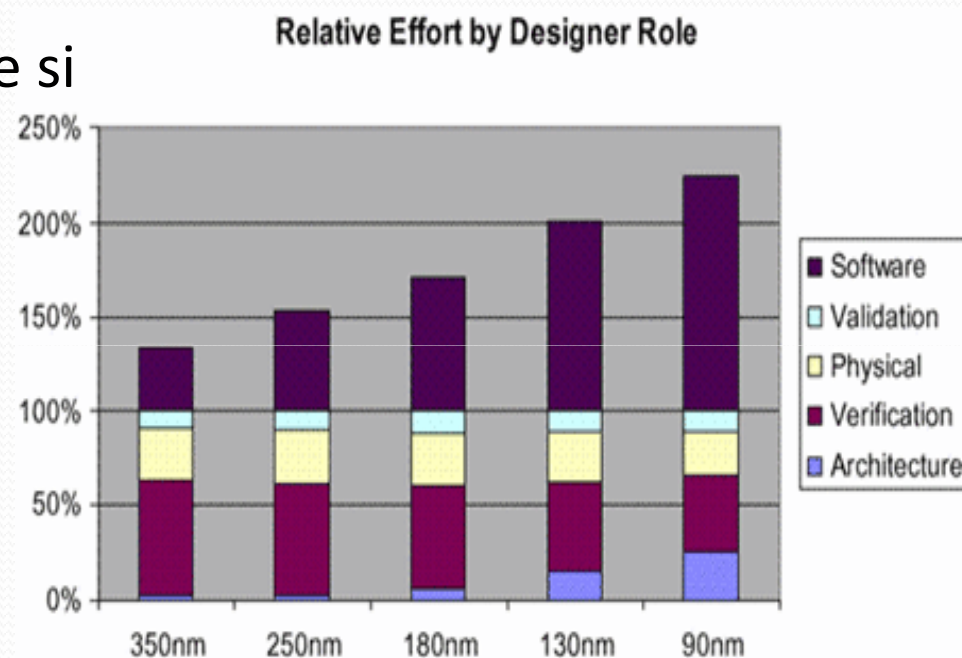
Introducerea tehnologiilor multimedia care cer o latime foarte mare de banda (ex. HDTV, GSM 3G) necesita incorporarea unui volum din ce in ce mai mare de software in dispozitivele de larg consum.

Sursa: ChipDesign Magazine



# Schimbari in efortul de dezvoltare

- 1996: 90% din efortul de dezvoltare SoC intra in hardware si abia 10% in software
- 2002: Proportia a ajunsese la 50-50



Costurile de dezvoltare software depasesc pe cele hardware in tehnologia de 130nm.

Sursa: ChipDesign Magazine

# Ubiquitous computing

Pervasive computing, ambient intelligence sau everywhere

- Model de interactiune om-masina in care procesarea informatiilor este adanc integrata in obiectele si activitatile zilnice.

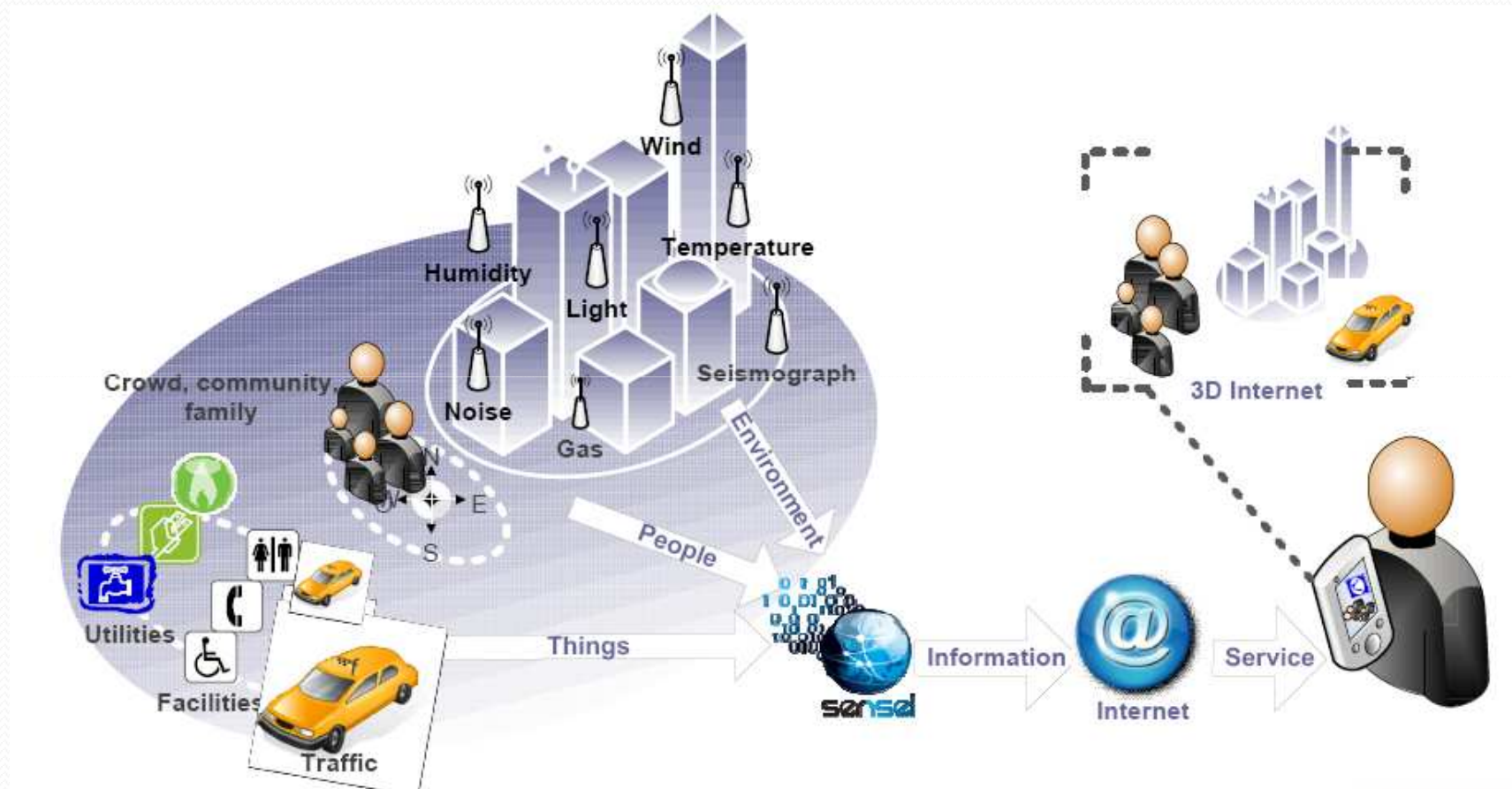
## Principii (M. Weiser, Xerox PARC):

- The purpose of a computer is to help you do something else.
- The best computer is a quiet, invisible servant.
- The more you can do by intuition the smarter you are; the computer should extend your *unconscious*.
- Technology should create calm.

## Clasificare device-uri:

- |          |        |
|----------|--------|
| ▶ Tabs   | ▶ Dust |
| ▶ Pads   | ▶ Skin |
| ▶ Boards | ▶ Clay |

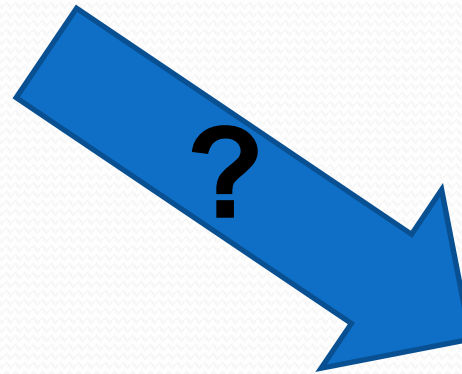
# Exemplu: Smart City



[www.sensei-project.eu](http://www.sensei-project.eu)



# Embedded Design – HowTo?



# Embedded System Design Flow

- **Modelare**

- ◆ Sistemul care urmeaza sa fie proiectat; experimente cu algoritmi ce vor fi folositi;

- **Partitionare (sau “rafinare”)**

- ◆ Functionalitate e impartita in mai multe module care interactioneaza;

- **Partitionare Hardware-Software**

- ◆ Elementele din modelul rafinat pot sa fie blocuri hardware sau software;

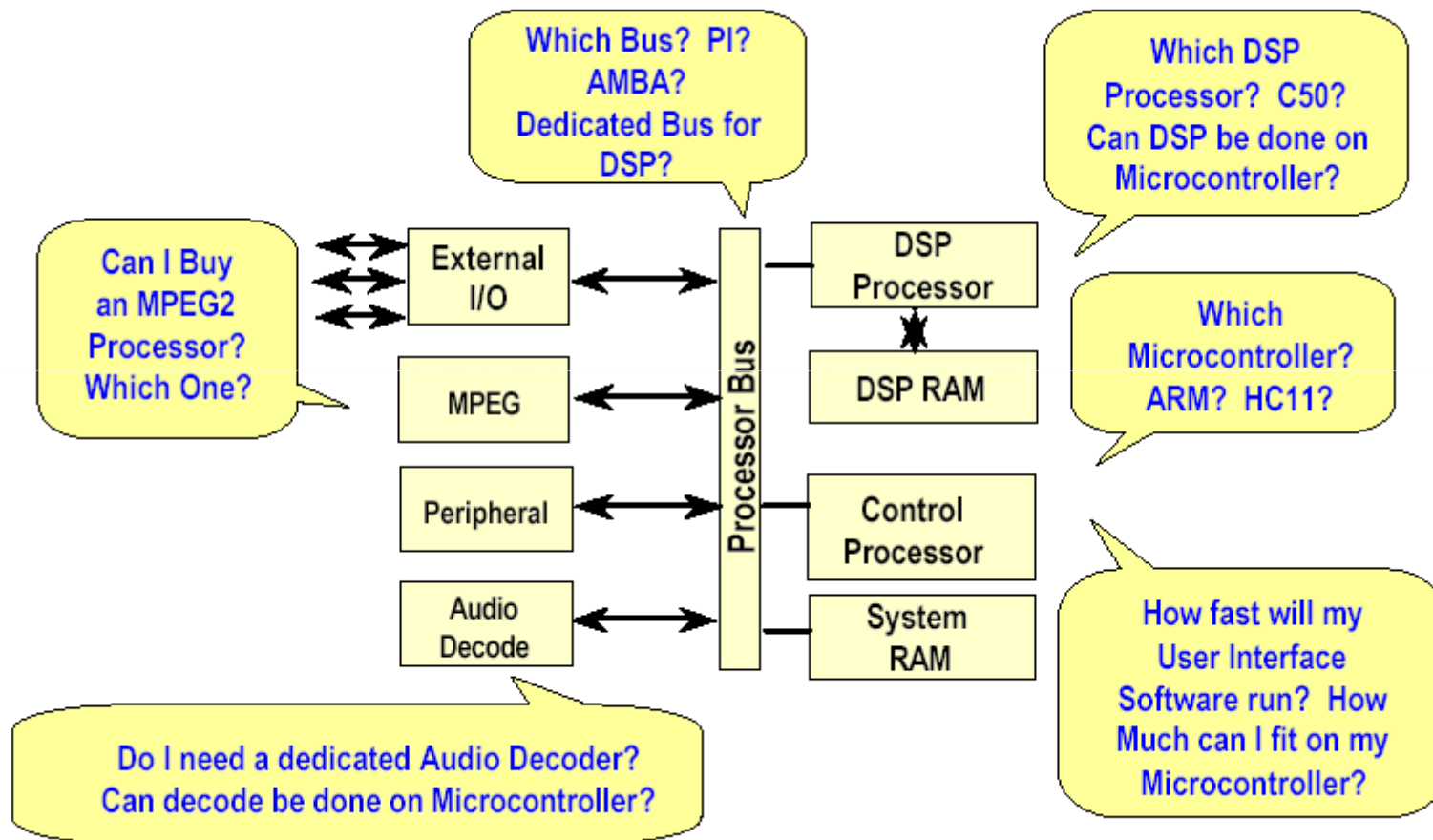
- **Gestionarea resurselor**

- ◆ Determinarea modulelor care folosesc aceeasi resursa (de cele mai multe ori hardware) si arbitrarea accesului la aceasta;

- **Implementarea**

- ◆ Scrierea de cod care sa ruleze pe un anumit procesor sau realizarea unei componente hardware specializate pe baza descrierilor functionale produse de etapele anterioare.

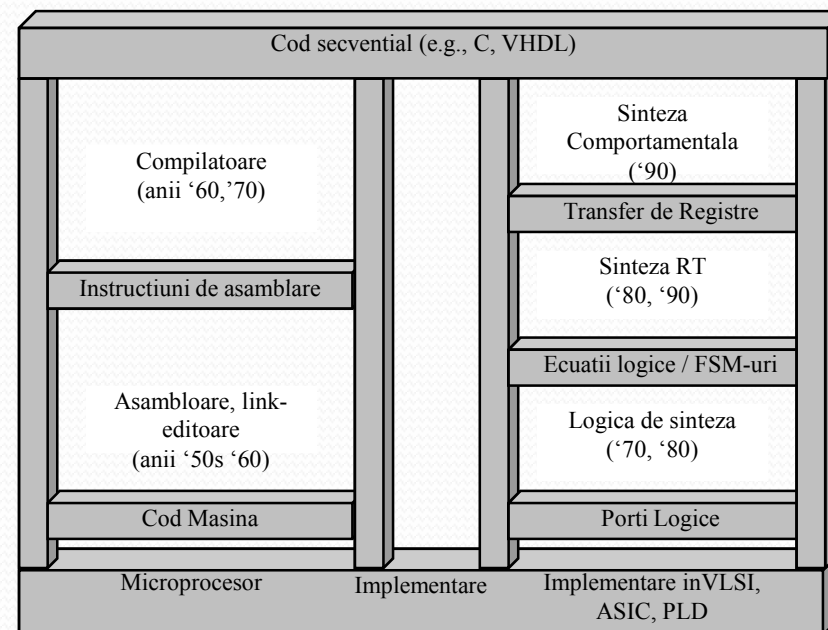
# Embedded System Design



# Evolutia paralela HW-SW

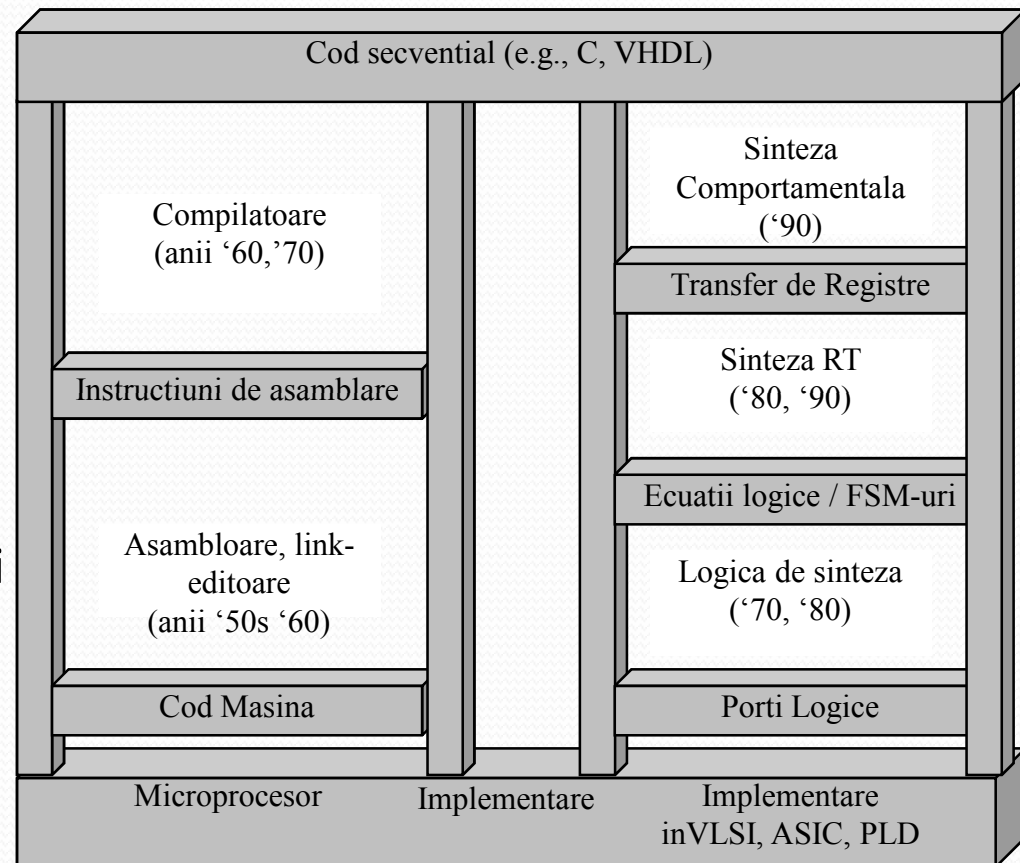
- **Evolutia proiectarii SW**
  - Cod Masina
  - Asambleare
  - Compilatoare
- **Evolutia proiectarii HW**
  - Porti logice interconectate
  - Sinteza logica
  - Sinteza transfer de registre
  - Sinteza comportamentala

## Scara co-evolutiei



# HW-SW co-design

- In trecut:
  - Metodologiile de proiectare hardware si software erau foarte diferite.
- Prezent:
  - Tehnica noua a sintezei permite redarea unei imagini unificate software+hardware



Source: Vahid/Givargis

# Notare Curs S. I.

- ✓ 4p – examen scris
- ✓ 3p – laborator
  - 2p – activitate
  - 1p – test final
- ✓ 1p – activitate la curs
- ✓ 2p – teme de casa

## Conditii de promovare:

1. Minim 2p in examen  
ȘI
2. Minim 1.5p in laborator  
ȘI
3. Minim 1.5p activitate + teme  
ȘI
4. Minim 6 prezente la laborator



# Copiatul Temelor / Examenelor

- Majoritatea studentilor nu imbratiseaza aceste metode
- Pentru toti ceilalti se aplica urmatoarele reguli:
  - Copiatul unei teme de casa anuleaza punctajul temei
  - Copiatul lucrarii de laborator = 0p la laboratorul respectiv
  - “Infractiuni” repetate -> anulara punctajului pentru toate temele / laboratoarele
  - Copiat la examen -> se repeta materia anul urmator

# Bibliografie

- Paper-uri de citit la fiecare curs
- Internet (Google & Wikipedia are your best friends)
- Michael Barr - **Programming Embedded Systems - With C and GNU Development Tools** 2e (O'Reilly, 2006)
- Jorg Henkel - **Designing Embedded Processors - A Low-Power Perspective** (Springer, 2007)
- Stuart Ball - **Embedded Microprocessor Systems - Real World Design** 3e (2002)
- **Embedded Linux Primer - A Practical Real-World Approach** (2006)
- O'Reilly - **Building Embedded Linux Systems**

# Cursuri Sisteme Embedded pe Net

- Stanford
  - Embedded Systems Design
    - <http://www.stanford.edu/class/ee281/course.html>
- U. California
  - Embedded System Design
    - <http://esd.cs.ucr.edu/index.html>
- Berkeley
  - EE 249: Design of Embedded Systems: Models, Validation, and Synthesis
    - <http://www-cad.eecs.berkeley.edu/~polis/class/index.html>
- U.T. Austin
  - EE382C-9 Embedded Software Systems
    - <http://www.ece.utexas.edu/~bevans/courses/ee382c/index.html>
- Berkeley
  - EE290N: Specification and Modeling of Reactive Real-Time Systems
    - <http://ptolemy.eecs.berkeley.edu/~eal/ee290n/index.html>
- UCI
  - ICS 212: Introduction to Embedded Computer Systems
    - <http://www.ics.uci.edu/~rgupta/ics212.html>
  - ICS 213: Software for Embedded Systems
    - <http://www.ics.uci.edu/~rgupta/ics213.html>

# Tema de Casa 1

1. Cititi lucrarea *The Computer for the 21<sup>st</sup> Century* de Mark Weiser
2. Scrieti un scurt eseu de 1-2 pagini in care sa aratati care sunt punctele in care Tennenhouse a avut sau n-a avut dreptate in legatura cu previziunile sale despre tehnologie si cota acesteia de piata. (lucrarea a fost scrisa in 2000, care sunt lucrurile care s-au adeverit in cei optsprezece ani care au trecut si nu in ultimul rand ce nu s-a adeverit.)
3. Nu ma intereseaza rezumatul lucrarii, vreau parerile voastre.
4. Tema trebuie predada in format doc sau pdf si trebuie sa respecte sablonul de pe site-ul de curs.
5. Nu ignorati lectura facultativa. Puteti gasi informatii utile si acolo.

**DEADLINE: 18 Octombrie (ora 23:55)**