



Sharif University of Technology Department of Computer Science and Engineering

Lec. 4.1: Embedded System Hardware



M. Ansari
Spring 2025
According to Peter Marwedel's Lectures

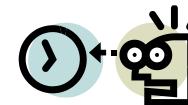
Motivation

(see lecture 1): "*The development of ES cannot ignore the underlying HW characteristics. Timing, memory usage, power consumption, and physical failures are important.*"

$$\int P \, dt$$

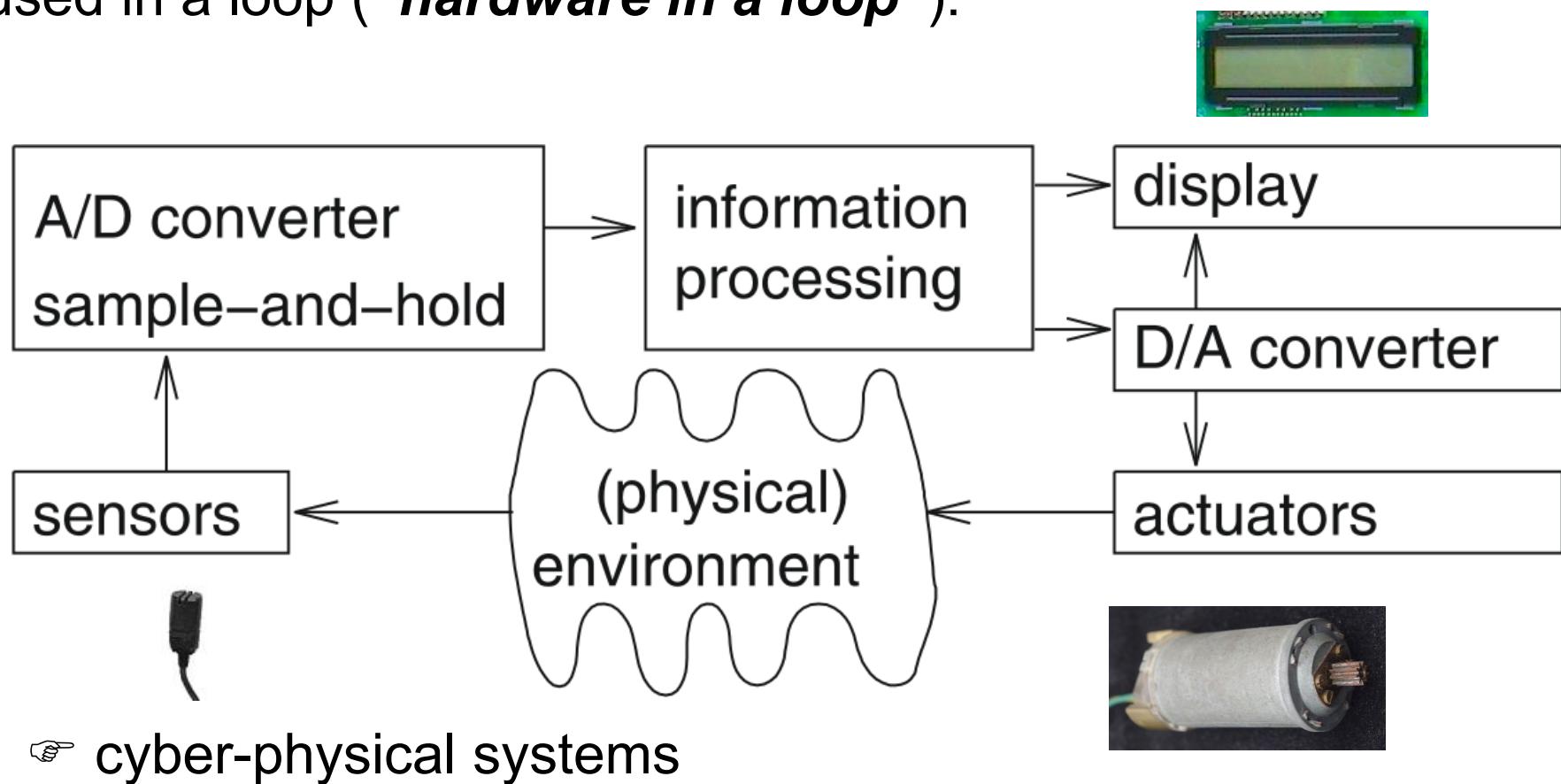
Reasons for considering hard- and software:

- Real-time behavior
- Efficiency
 - Energy
 - ...
- Security
- Reliability
 - ...



Embedded System Hardware

Embedded system hardware is frequently used in a loop (***“hardware in a loop”***):

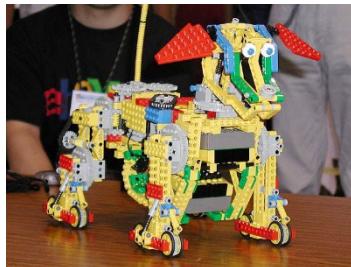


Many examples of such loops

- Heating
- Lights
- Engine control
- Power supply
- ...
- Robots



© P. Marwedel, 2011



Embedded System Hardware

- Sensors-



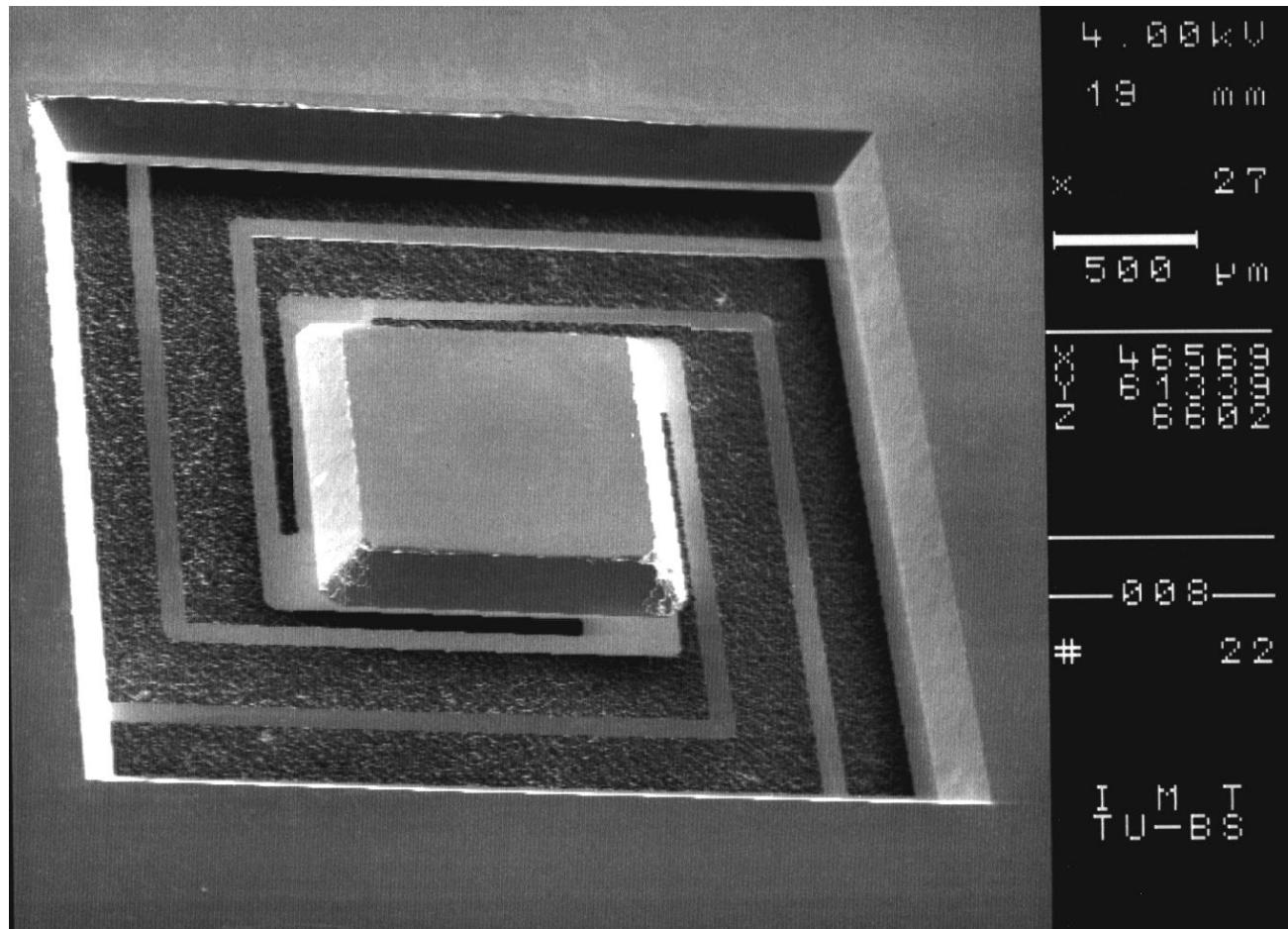
Sensors

- ❖ Processing of physical data starts with capturing this data. Sensors can be designed for virtually every physical and chemical quantity, including
 - weight, velocity, acceleration, electrical current, voltage, temperatures, and
 - chemical compounds.
- ❖ Many physical effects used for constructing sensors.

Examples:

- law of induction (generat. of voltages in a magnetic field),
- light-electric effects.
- ❖ Huge amount of sensors designed in recent years.

Example: Acceleration Sensor



Courtesy & ©: S. Bürgenbach, TU Braunschweig

Charge-coupled devices (CCD) image sensors

- ❖ CCD technology is optimized for optical applications.
- ❖ In CCD technology, charges must be transferred from one pixel to the next until they can finally be read out at an array boundary.
- ❖ This sequential charge transfer also gave CCDs their name. For CCD sensors, interfacing is more complex.
- ❖ Several application areas for CCDs have disappeared, but they are still used in areas such as scientific image acquisition.



CMOS image sensors

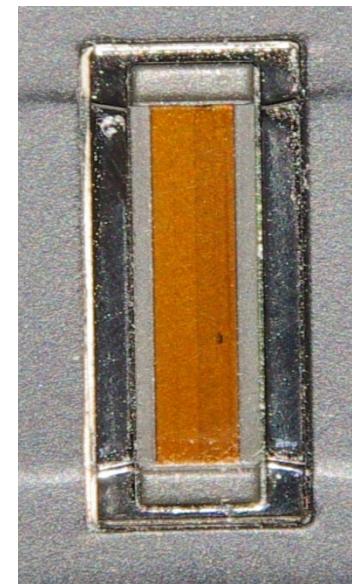
- ❖ The architecture of CMOS sensor arrays is similar to that of standard memories: individual pixels can be randomly addressed and read out.
- ❖ CMOS sensors use standard CMOS technology for integrated circuits.
- ❖ CMOS sensors require only a single standard supply voltage and interfacing in general is easy.
- ❖ CMOS-based sensors can be cheap.
- ❖ Based on standard production process for CMOS chips, allows integration with other components.

Comparison CCD/CMOS sensors

Property	CCD	CMOS
Technology optimized for	Optics	VLSI technology
Technology	Special	Standard
Smart sensors	No, no logic on chip	Logic elements on chip
Access	Serial	Random
Size	Limited	Can be large
Power consumption	Low	Larger
Video mode	Possibly too slow	ok
Applications	Situation is changing over the years	

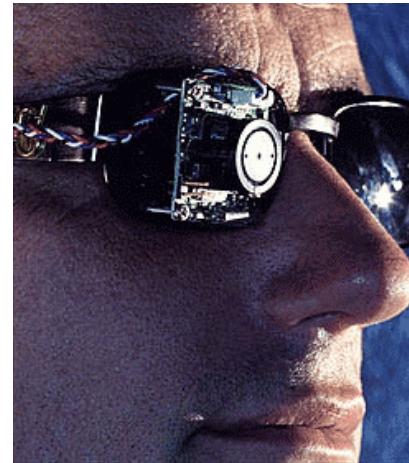
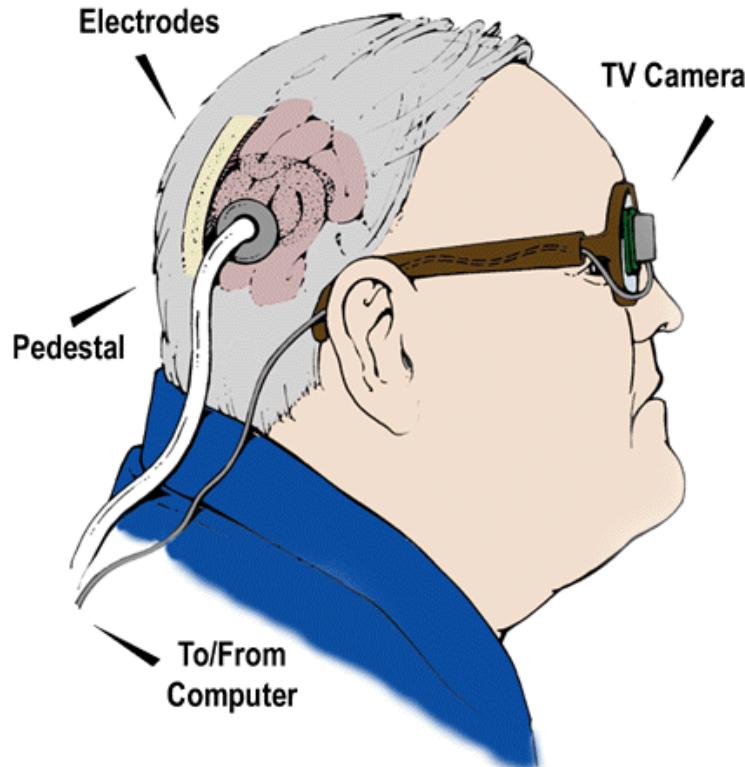
Example: Biometrical Sensors

- ❖ Demands for higher security standards as well as the need to protect mobile and removable equipment have led to an increased interest in authentication.
 - e.g.: Fingerprint sensor
- ❖ False accepts as well as false rejects are an inherent problem of biometric authentication.



© P. Marwedel, 2010

Artificial eyes (1)



© Dobelle Institute
(was at www.dobelle.com)

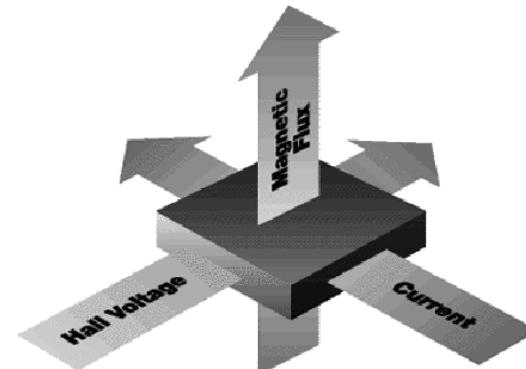
Artificial eyes (2)

- Translation into sound
[<http://www.seeingwithsound.com/etumble.htm>]



Other sensors

- Rain sensors for wiper control
("Sensors multiply like rabbits" [ITT automotive])
- Pressure sensors
- Proximity sensors
- Engine control sensors



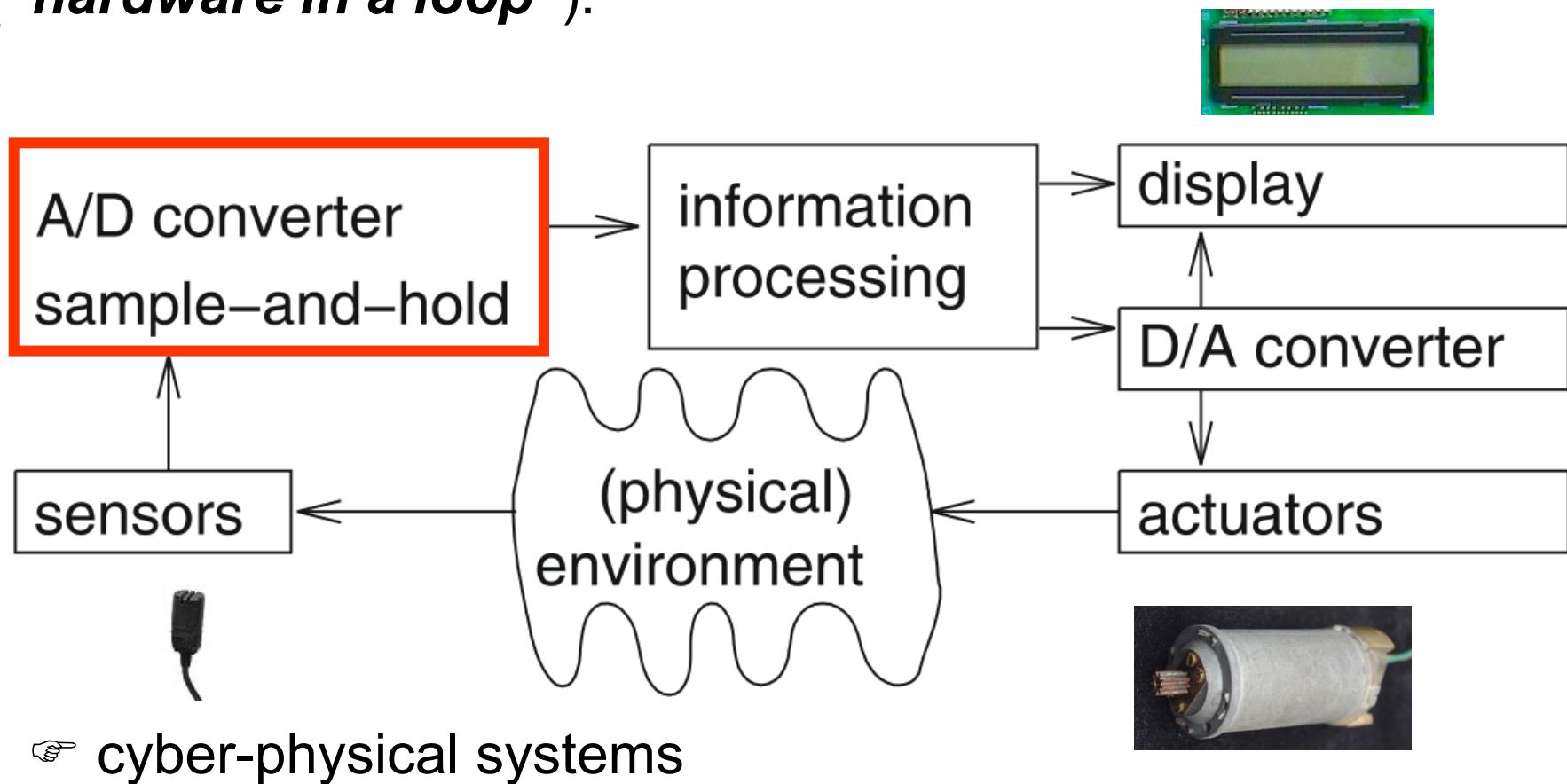
Embedded System Hardware

- A/D Converter-



Embedded System Hardware

Embedded system hardware is frequently used in a loop (“*hardware in a loop*“):



Signals

Sensors generate *signals*

Definition: a **signal** s is a mapping
from the time domain D_T to a value domain D_V :

$$s : D_T \rightarrow D_V$$

D_T : continuous or discrete time domain

D_V : continuous or discrete value domain.

Discretization of time

Digital computers require discrete sequences of physical values

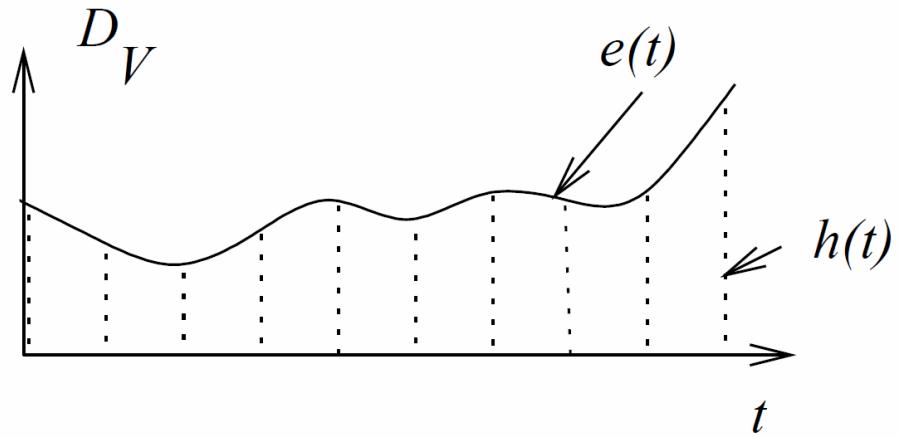
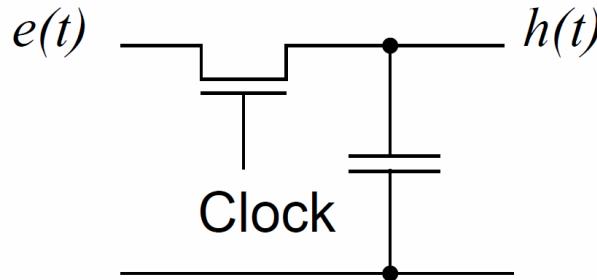
$$s : D_T \rightarrow D_V$$


Discrete time domain

- ☞ Sample-and-hold circuits

Sample-and-hold circuits

Clocked transistor + capacitor;
Capacitor stores sequence values



$e(t)$ is a mapping $\mathbb{R} \rightarrow \mathbb{R}$

$h(t)$ is a **sequence** of values or a mapping $\mathbb{Z} \rightarrow \mathbb{R}$

Discretization of values: A/D-converters

Digital computers require digital form of physical values

$$s: D_T \rightarrow D_V$$



Discrete value domain

- ☞ A/D-conversion; many methods with different speeds.

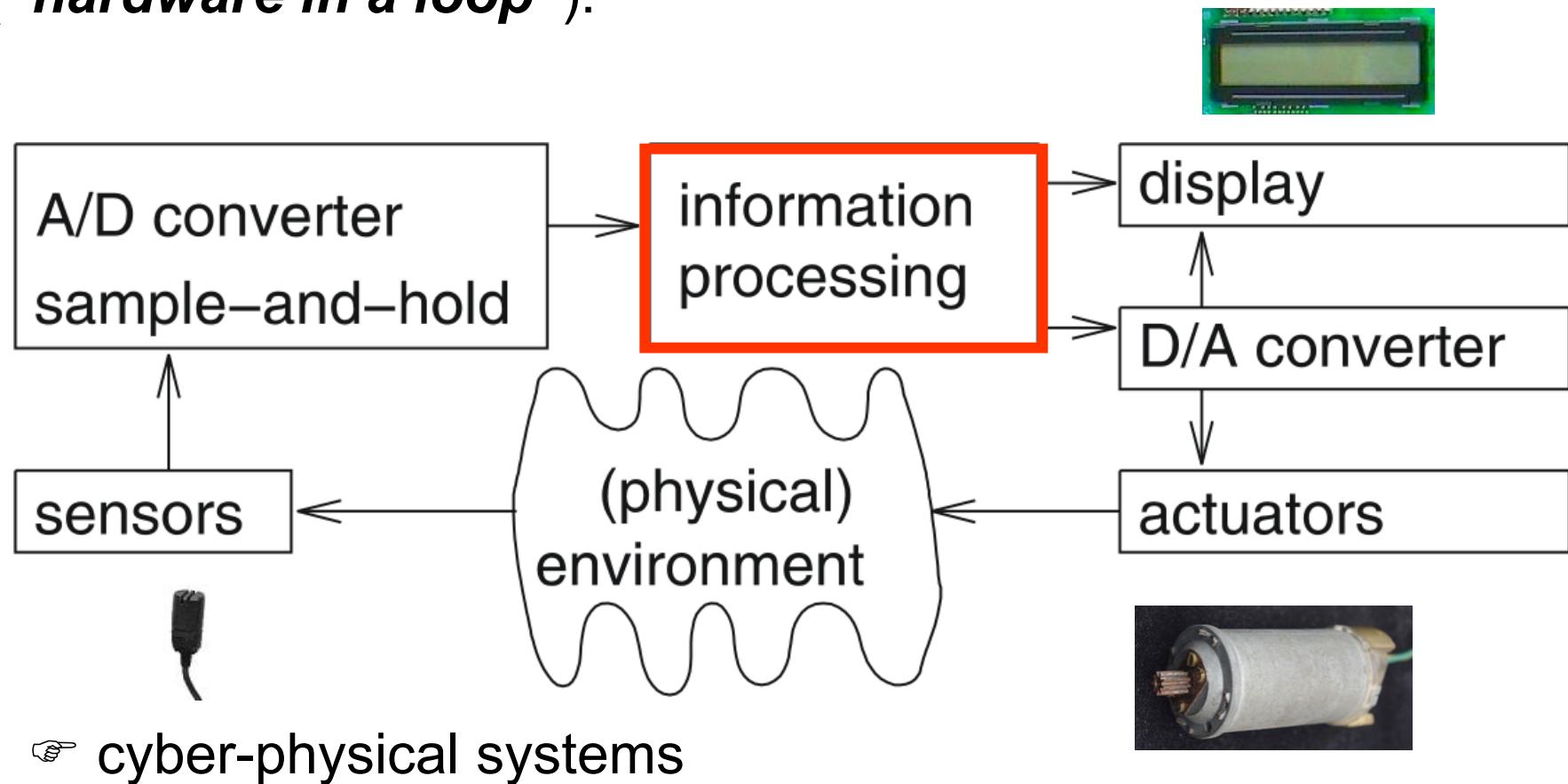
Embedded System Hardware

- Information Processing-



Embedded System Hardware

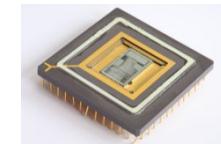
Embedded system hardware is frequently used in a loop (“*hardware in a loop*“):



Efficiency: slide from lecture 1 applied to processing

- CPS & ES must be **efficient**

- Code-size efficient
(especially for systems on a chip)



- Run-time efficient



- Weight efficient



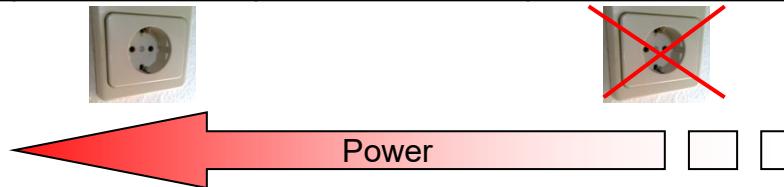
- Cost efficient

- Energy efficient



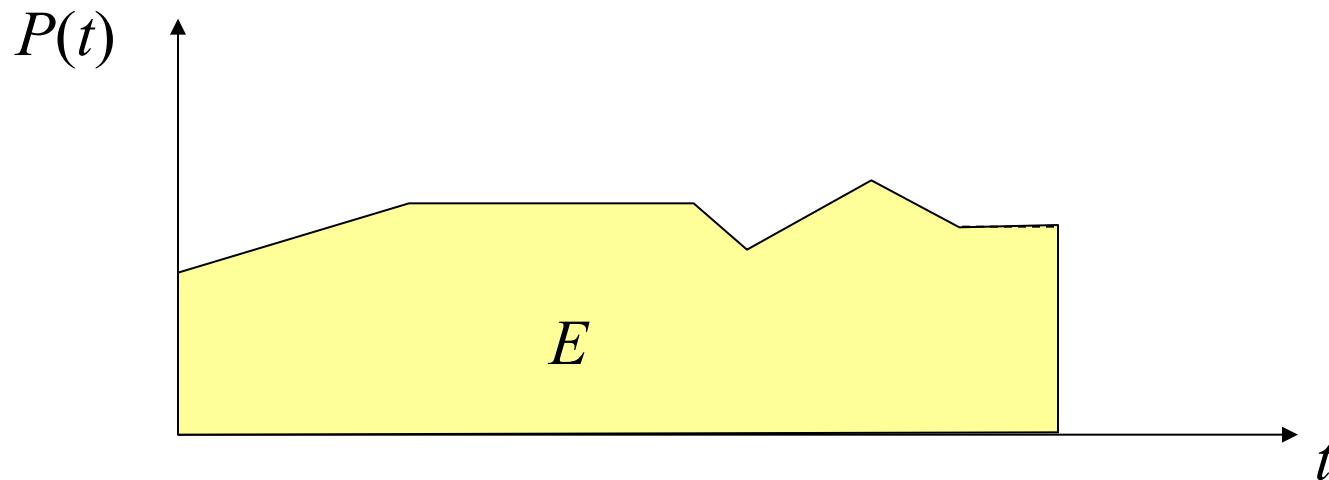
Why care about energy efficiency ?

		Relevant during use?		
Execution platform		Plugged	Uncharged periods	Unplugged
	E.g.	Factory	Car	Sensor
Global warming		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Cost of energy		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Increasing performance		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Problems with cooling, avoiding hot spots		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Avoiding high currents & metal migration		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Reliability		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Energy a very scarce resource		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Should we care about energy consumption or about power consumption?

$$E = \int P(t) dt$$



Both are closely related, but still different

Should we care about energy consumption or about power consumption (2)?

- Minimizing **power consumption** important for
 - design of the power supply & regulators
 - dimensioning of interconnect, short term cooling
- Minimizing **energy consumption** important due to
 - restricted availability of energy (mobile systems)
 - cooling: high costs, limited space
 - thermal effects
 - dependability, long lifetimes



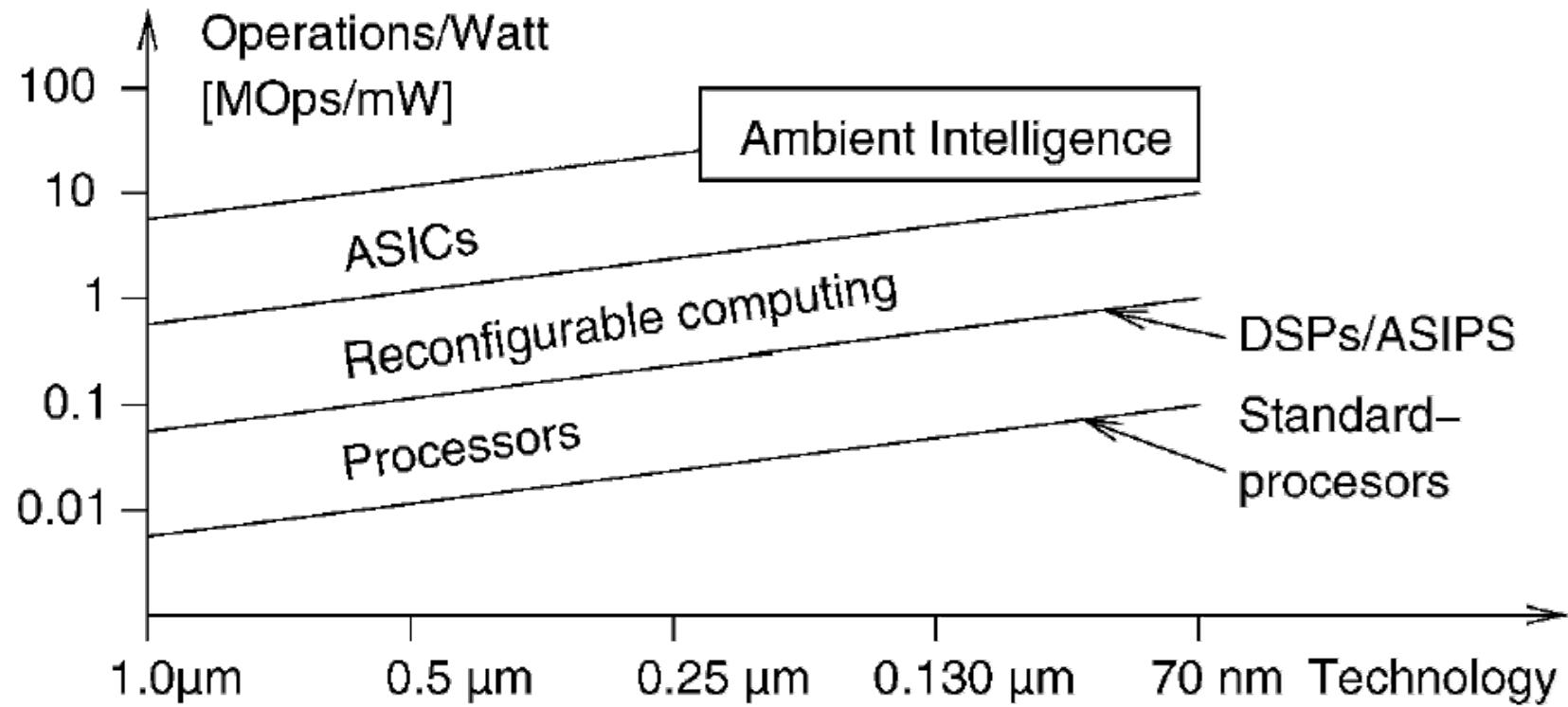
👉 **In general, we need to care about both**

Energy and Power Consumption

- ❖ Power consumption
 - Size of the power supply
 - Design of the voltage regulators
 - Dimensioning of the interconnect
 - Short term cooling

- ❖ Energy consumption
 - Mobile applications
 - Battery life time

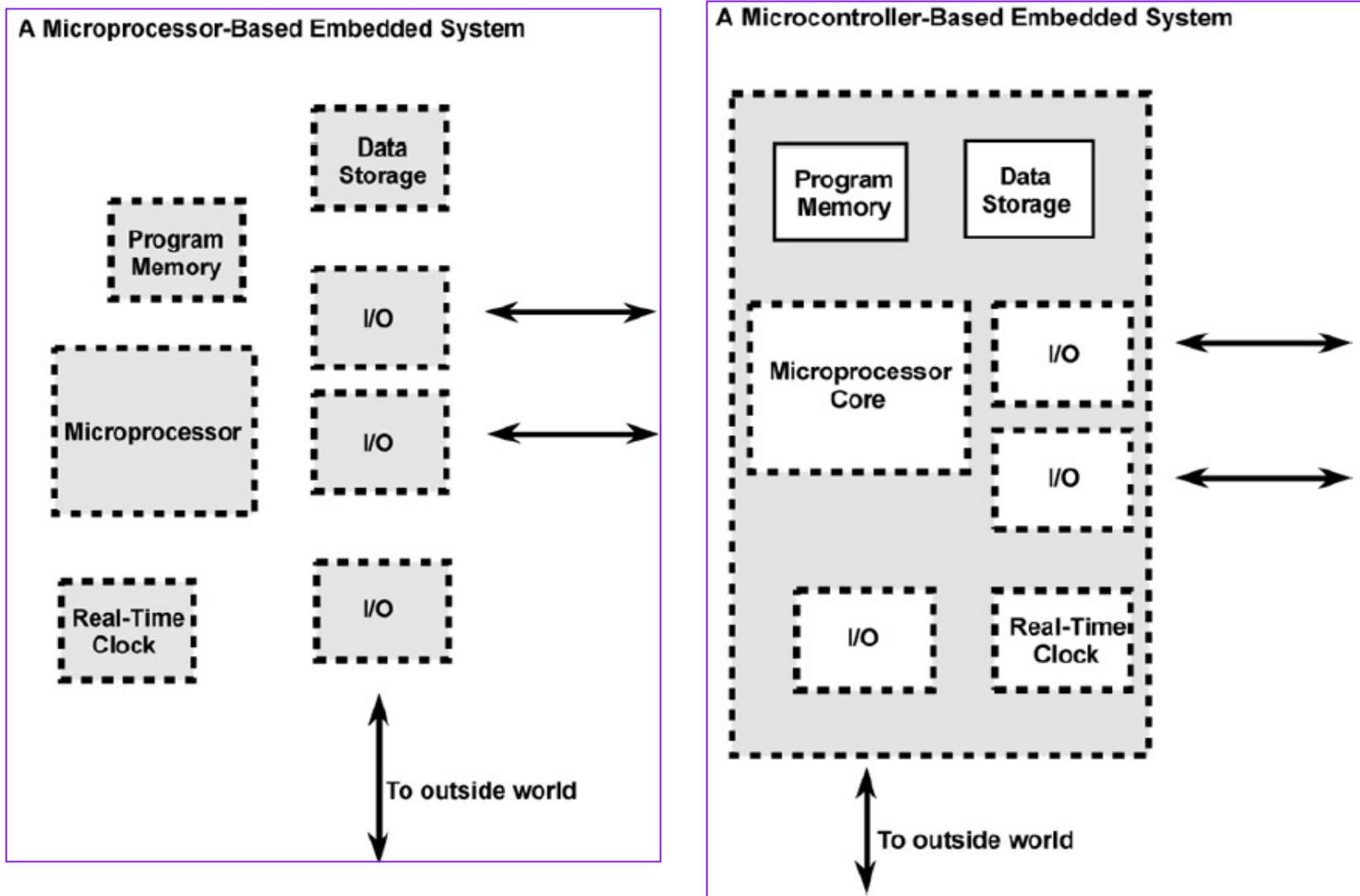
Impact of PUs on Energy and Power Consumption



Embedded Processors vs. PCs

- ❖ Embedded processors do not need to be instruction set compatible with PCs.
- ❖ Efficiency
 - Energy efficiency
 - Code-size efficiency
 - Run-time efficiency (e.g., ASIPs)

Microprocessor VS. Microcontroller



Microcontrollers

- ❖ Microcontrollers have become so prevalent and even dominate the entire embedded world.
- ❖ Lower cost: One part replaces many parts.
- ❖ More reliable: Fewer packages, fewer interconnects.
- ❖ Better performance: System components are optimized for their environment, Signals can stay on the chip.

Core-Based Microcontrollers

- ❖ 8086 processor
 - 80186 family of devices

- ❖ Motorola's 68000 and 68020
 - 68300 family of devices

Benefit ⇒ Cost Reduction

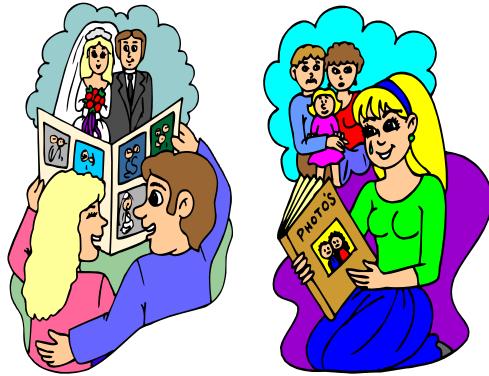
Embedded System Hardware

- Memory-

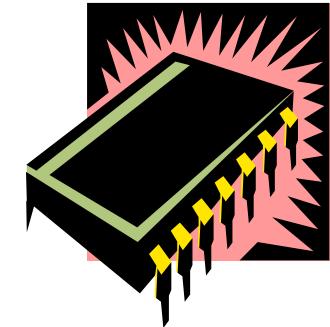


Memory

Memories?



Oops!
Memories!



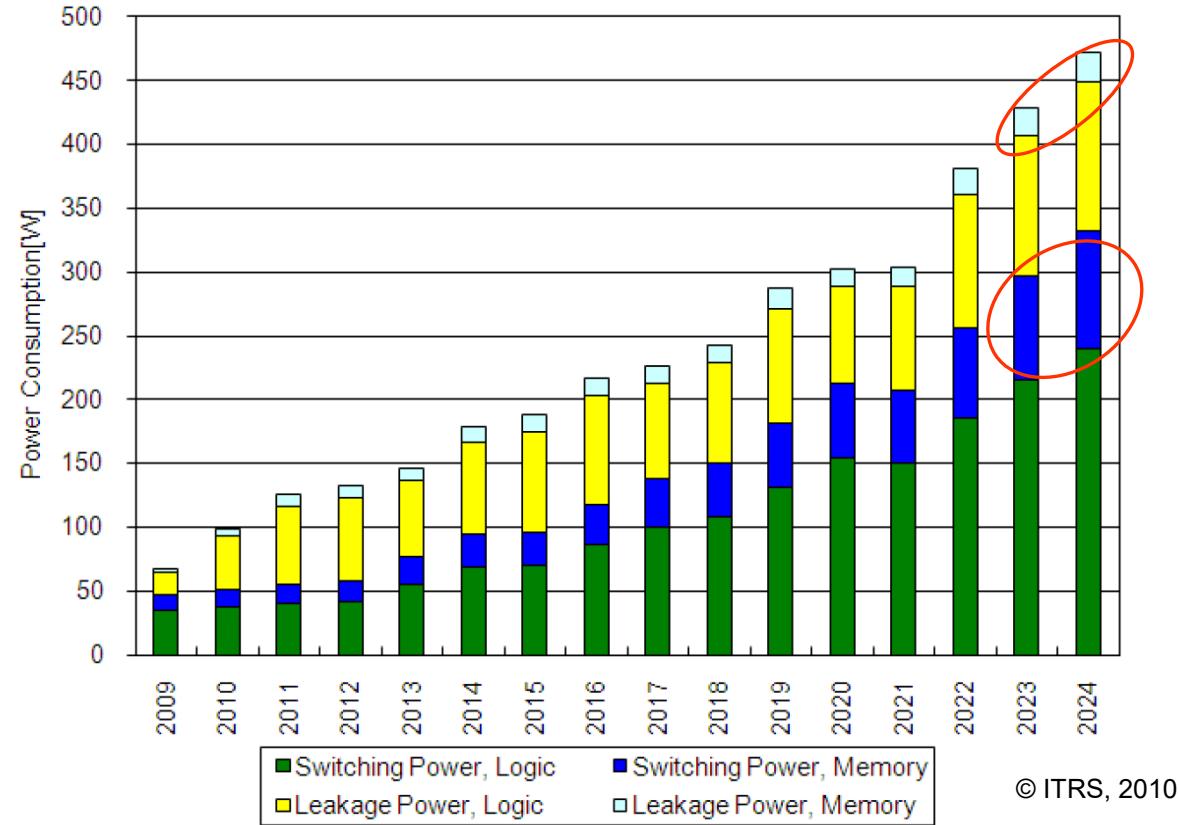
For the memory, efficiency is again a concern:

- capacity
- energy efficiency
- speed (latency and throughput); predictable timing
- size
- cost
- other attributes (volatile vs. persistent, etc)

Where is the power consumed?

- Stationary systems -

- According to *International Technology Roadmap for Semiconductors* (ITRS), 2010 update, [www.itrs.net]



- Switching power, logic dominating
- Overall power consumption a nightmare for environmentalists

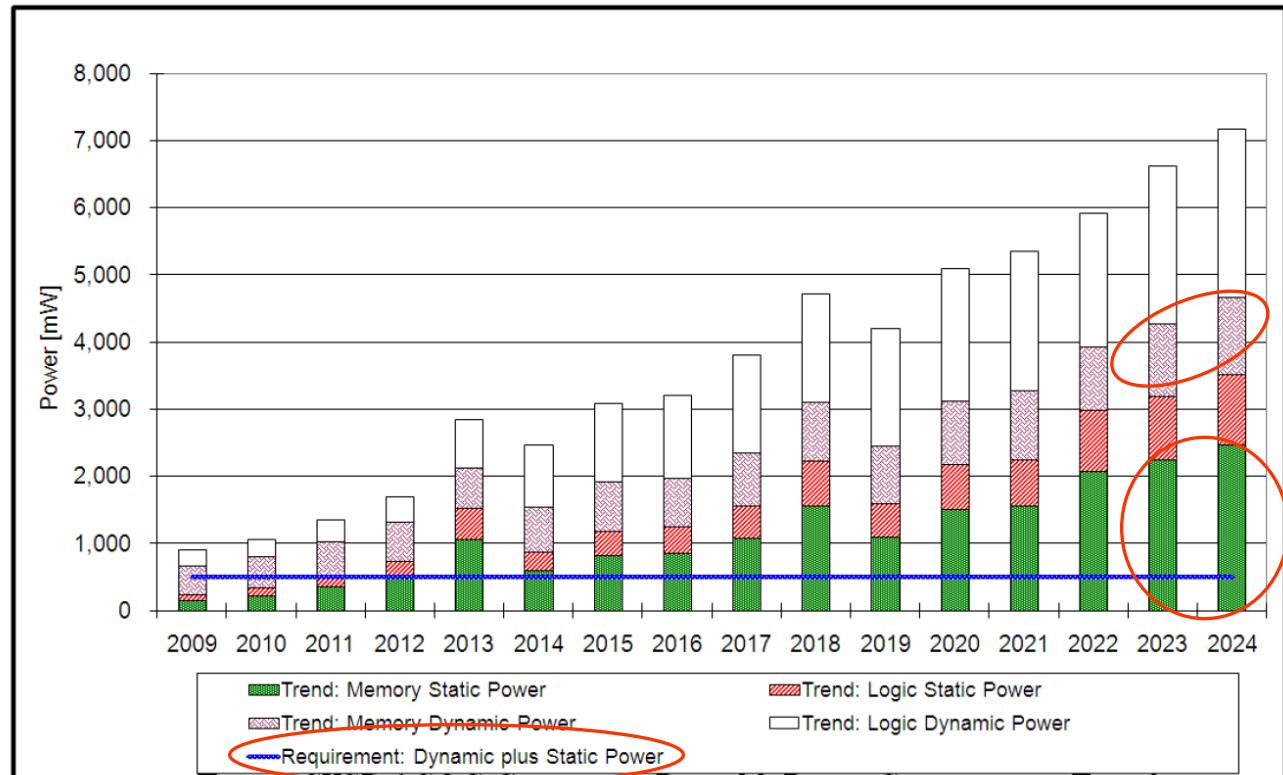
Where is the power consumed?

- Consumer portable systems -

- According to *International Technology Roadmap for Semiconductors* (ITRS), 2010 update, [www.itrs.net]

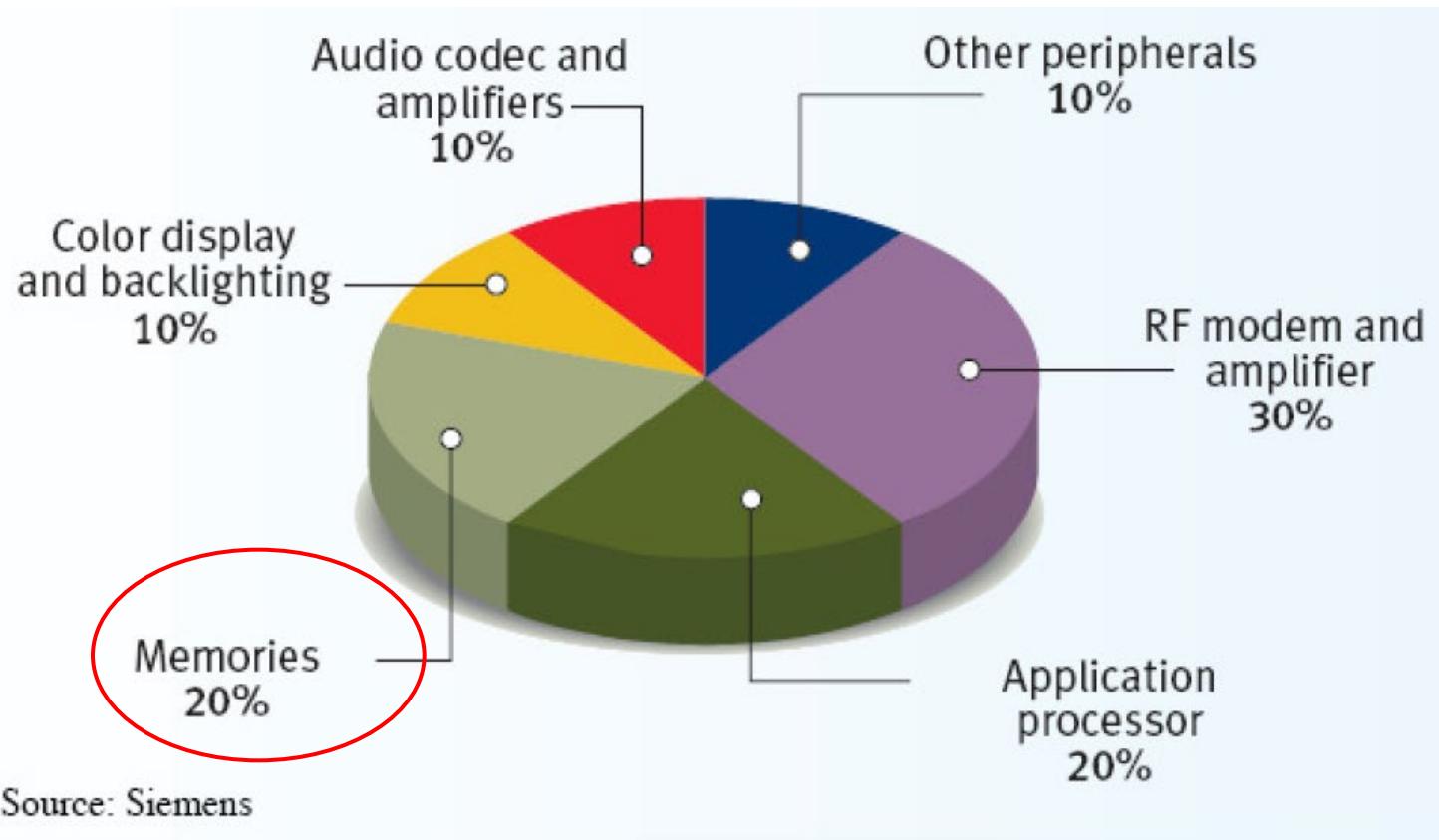
- Based on current trends

- Memory and logic, static and dynamic relevant
- Following current trends will violate maximum power constraint (0.5-1 W).



© ITRS, 2010

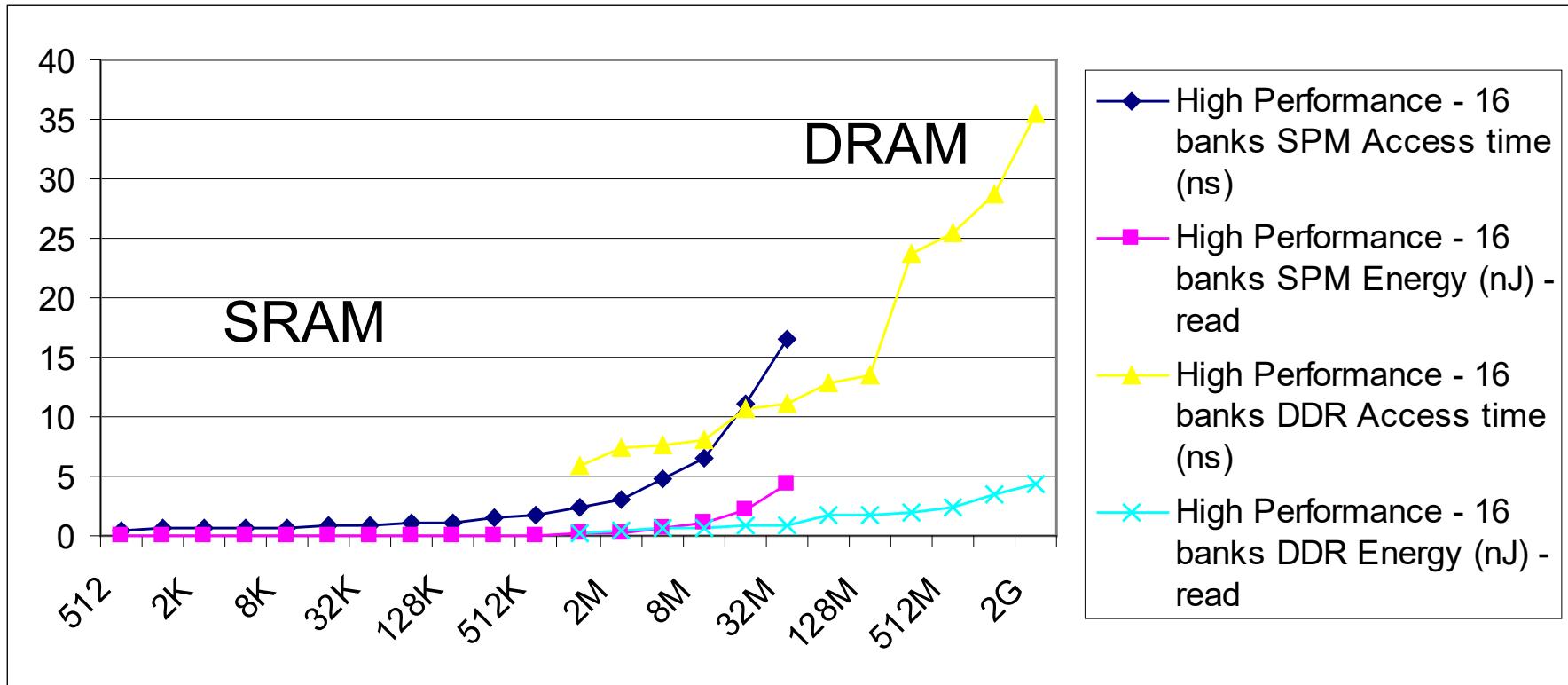
Memory energy significant even if we take display and RF of mobile device into account



[O. Vargas (Infineon Technologies): Minimum power consumption in mobile-phone memory subsystems; Pennwell Portable Design - September 2005;] Thanks to Thorsten Koch (Nokia/ Univ. Dortmund) for providing this source.

Energy consumption and access times of memories

Example CACTI: Scratchpad (SRAM) vs. DRAM (DDR2):

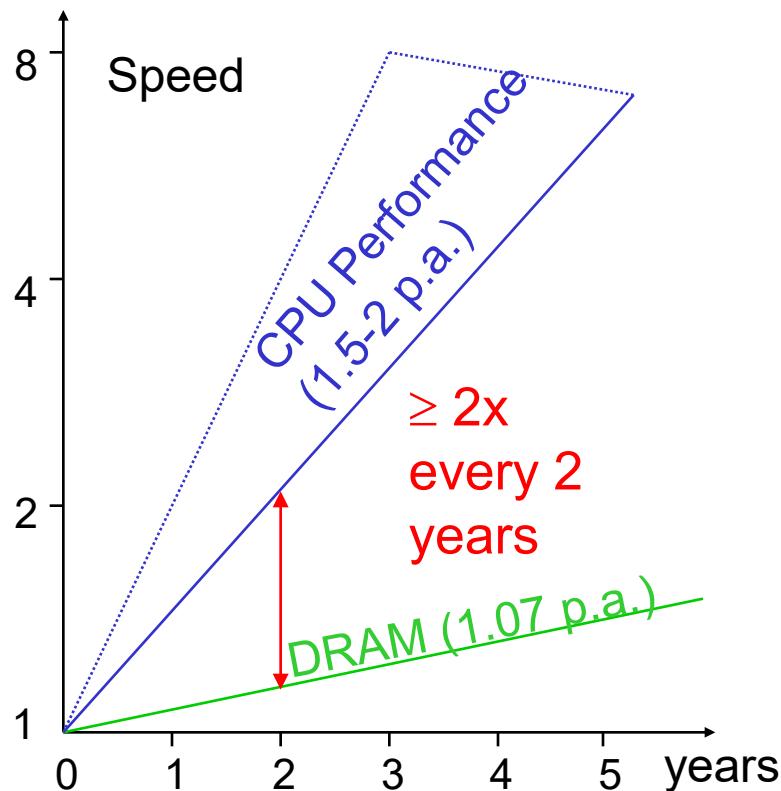


16 bit read; size in bytes;
65 nm for SRAM, 80 nm for DRAM

Source: Olivera Jovanovic,
TU Dortmund, 2011

Trends for the Speeds

Speed gap between processor and main DRAM increases



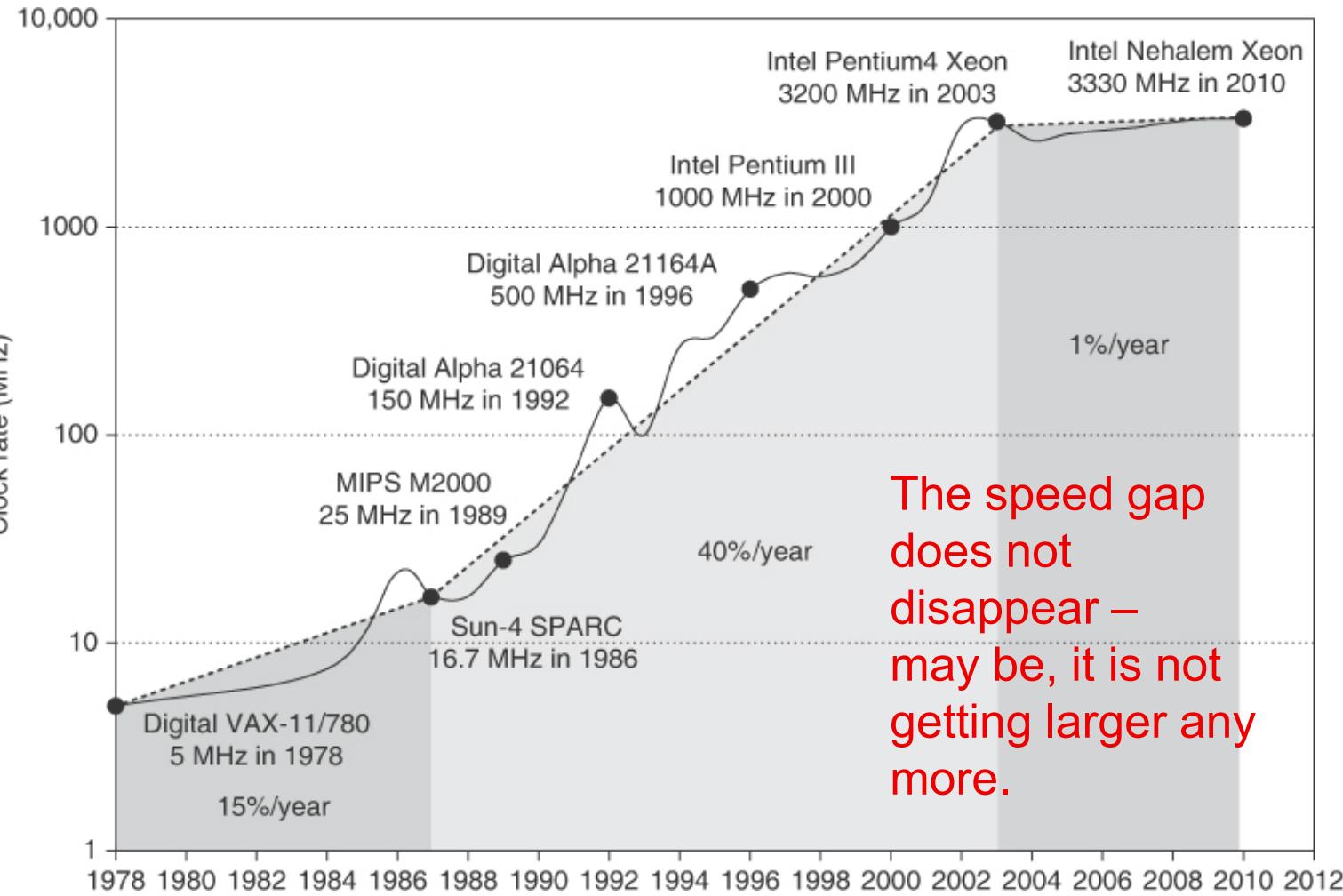
Similar problems also for embedded systems & MPSoCs

- ☞ Memory access times >> processor cycle times
- ☞ “Memory wall” problem



[P. Machanik: Approaches to Addressing the Memory Wall, TR Nov. 2002, U. Brisbane]

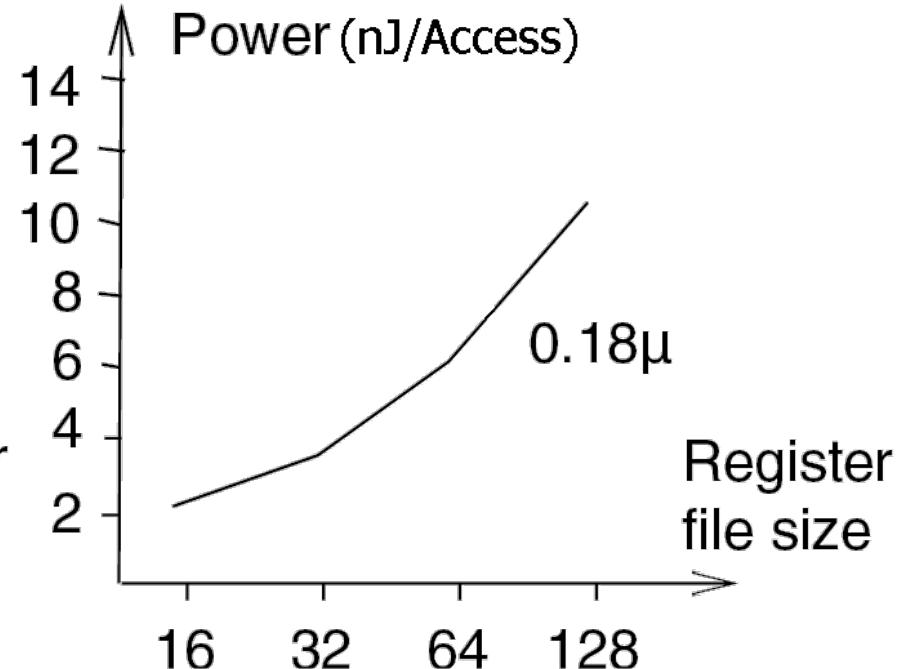
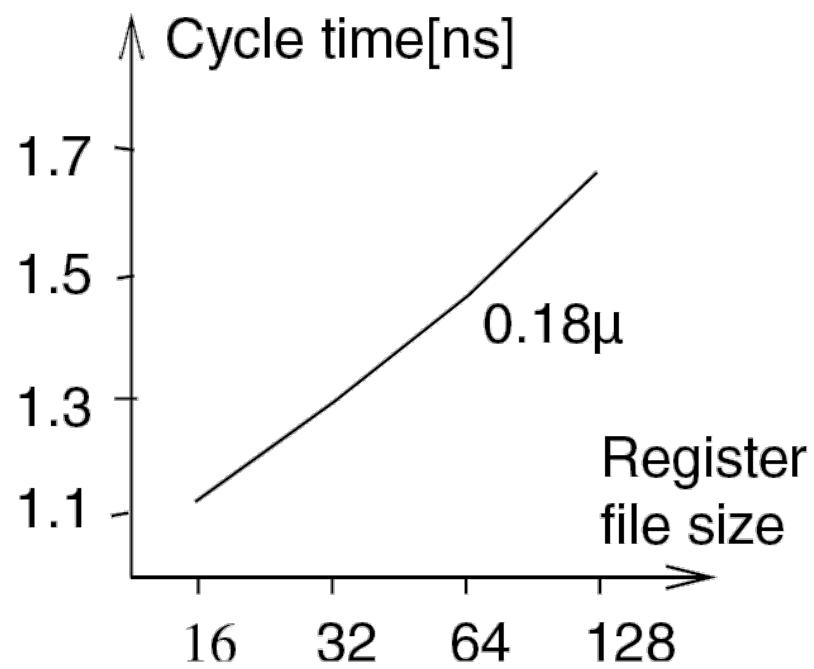
However, clock speed increases have come to a halt



Copyright © 2011, Elsevier Inc. All rights Reserved.

[Hennessy/Patterson: Computer Architecture, 5th ed., 2011]

Impact of MEM Size

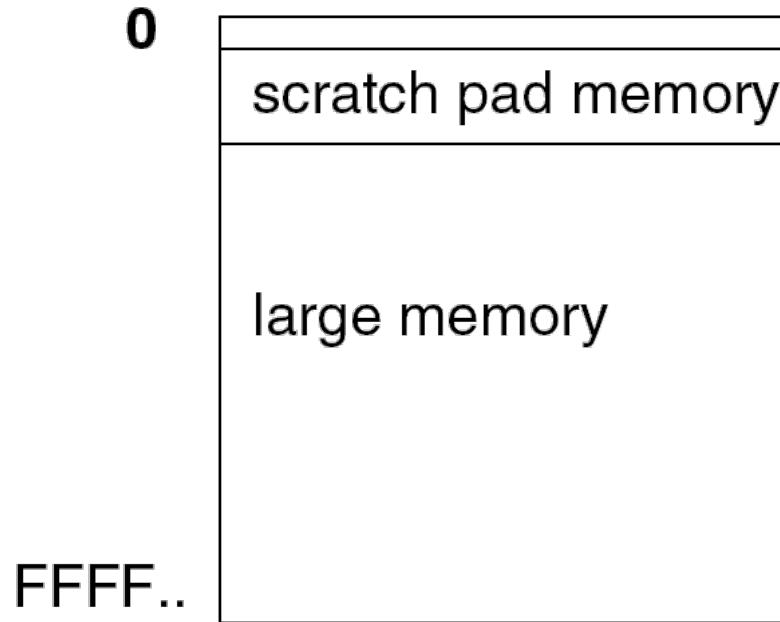


Disadvantages of Caches

- ❖ The **predictability** of the real-time performance of caches is frequently low.
 - Serious problems in scheduling

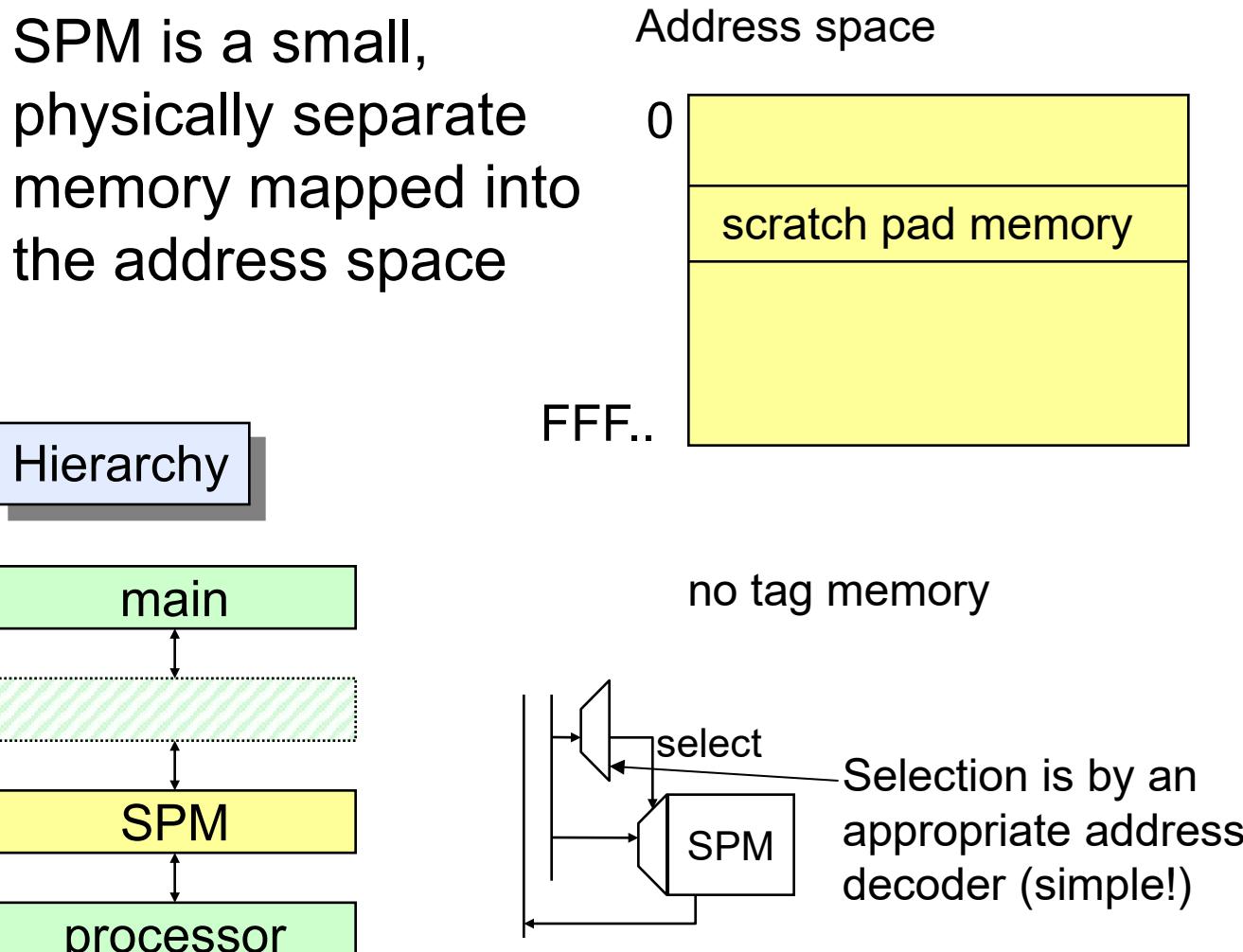
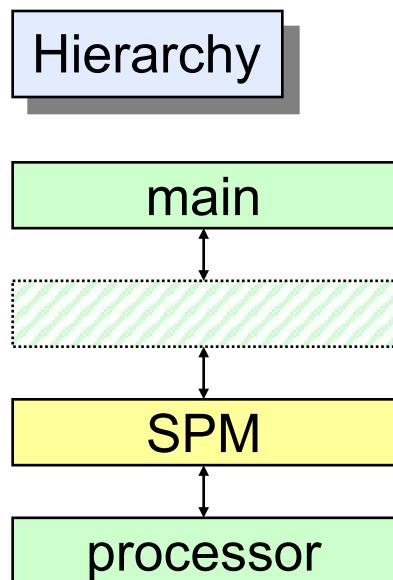
Solution: Scratch Pad Memory

- ❖ SPM vs. Cache
 - The only difference is that Caches are transparent while SPMs are not.



Hierarchical memories using scratch pad memories (SPM)

SPM is a small, physically separate memory mapped into the address space

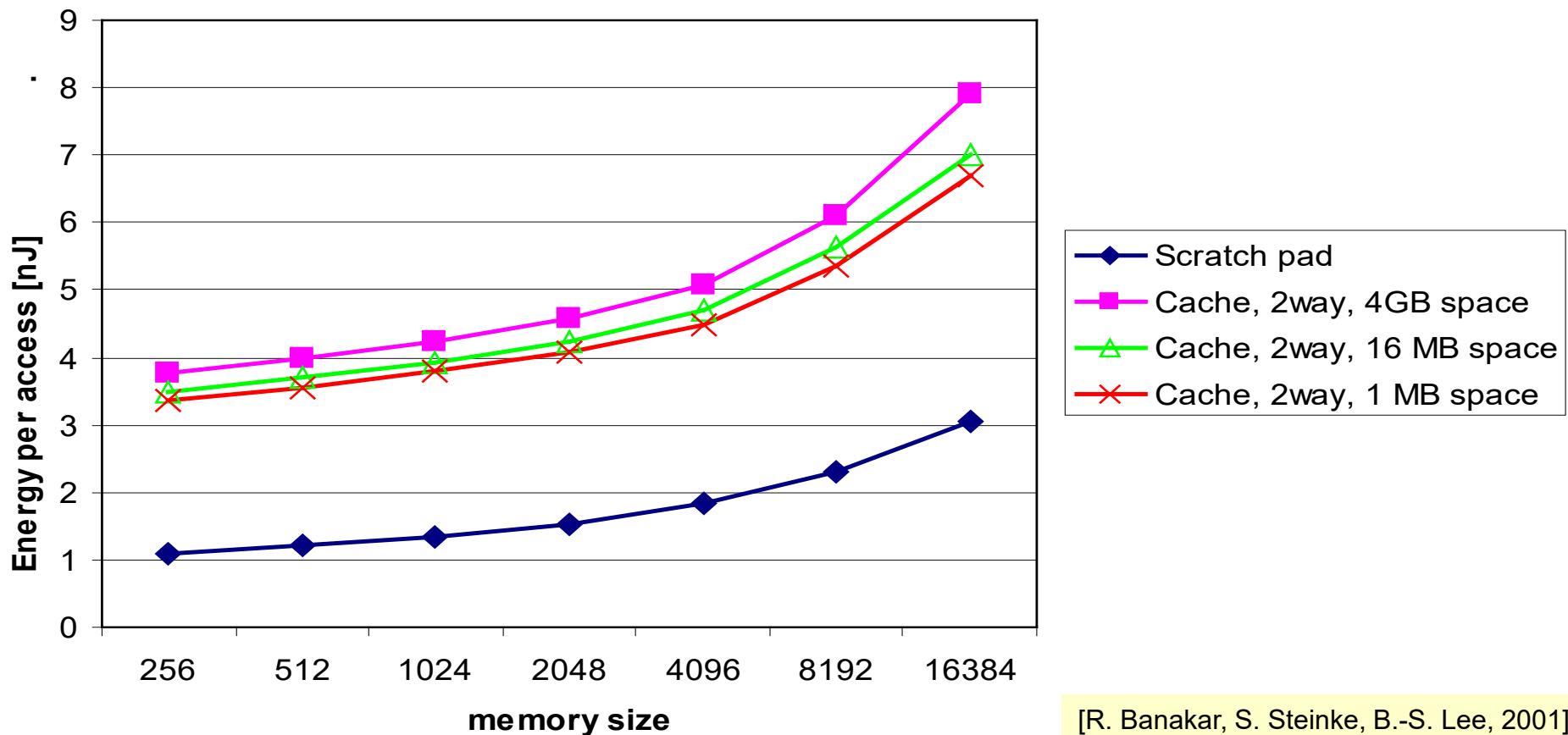


Examples:

- Most ARM cores allow tightly coupled memories
- IBM Cell
- Infineon TriCore
- Many multi-cores, due to high costs of coherent caches

Why not just use a cache? (SPM vs. Cache)

- ❖ Energy for parallel access of sets, in comparators, muxes.



[R. Banakar, S. Steinke, B.-S. Lee, 2001]

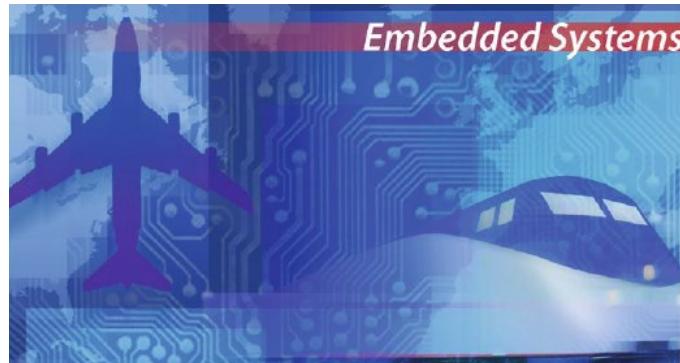
SPM vs. Cache

- ❖ SPMs are more power/energy-efficient than caches.
 - There is no cache controller hardware

- ❖ From a programming point of view:
 - The usage of SPMs is harder than caches.

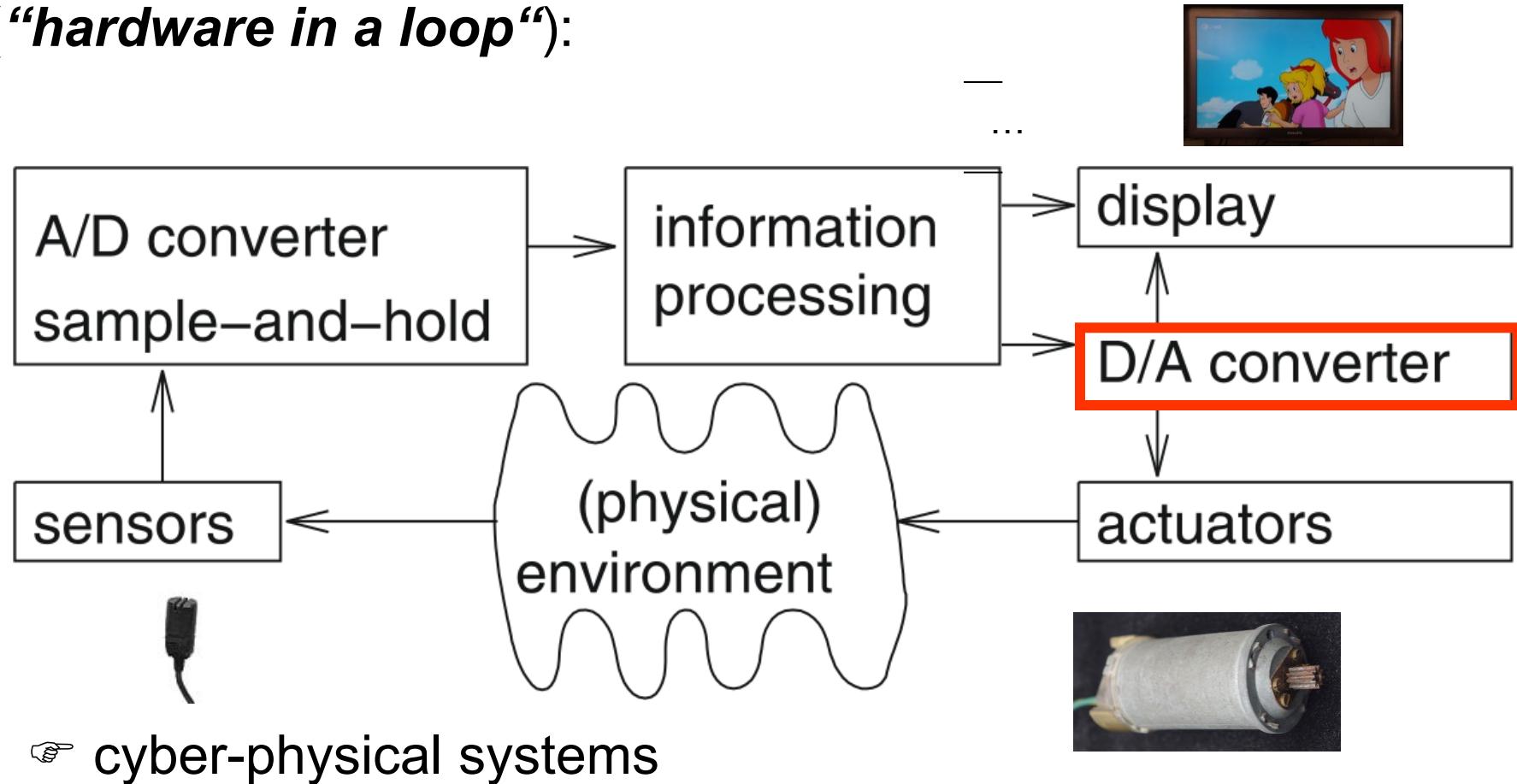
Embedded System Hardware

- D/A Converter-

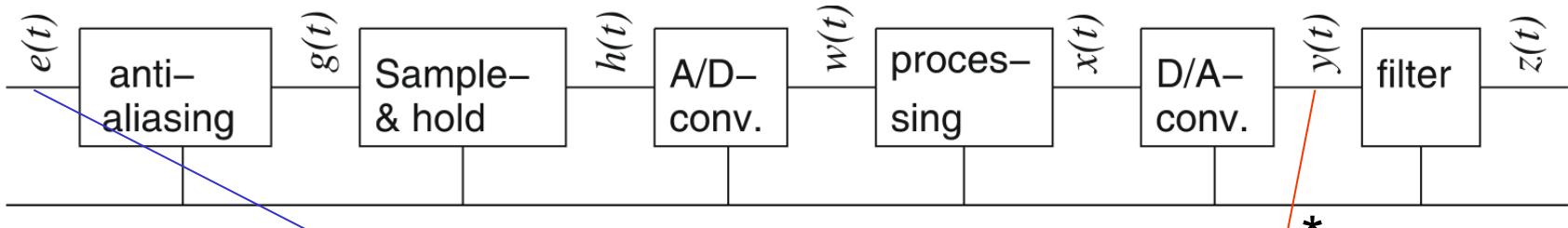


Embedded System Hardware

Embedded system hardware is frequently used in a loop
("hardware *in a loop*"): ...

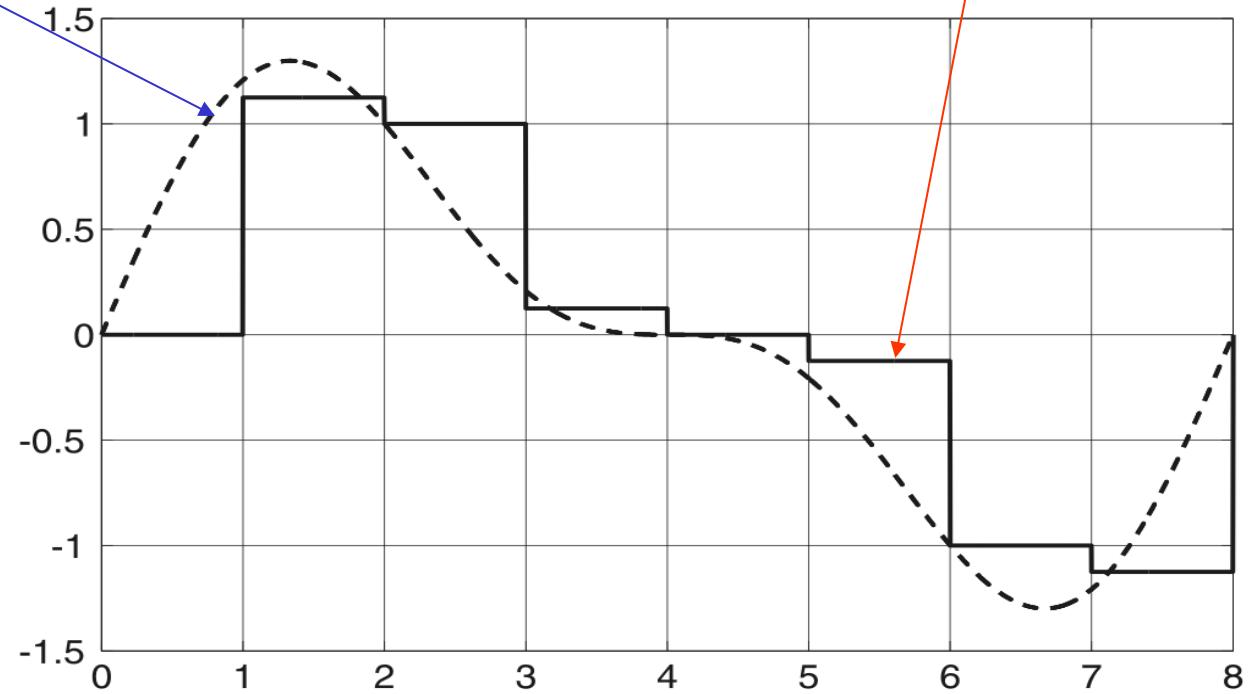


Output generated from signal $e_3(t)$



* Assuming
“zero-order
hold”

Possible to
reconstruct
input
signal?



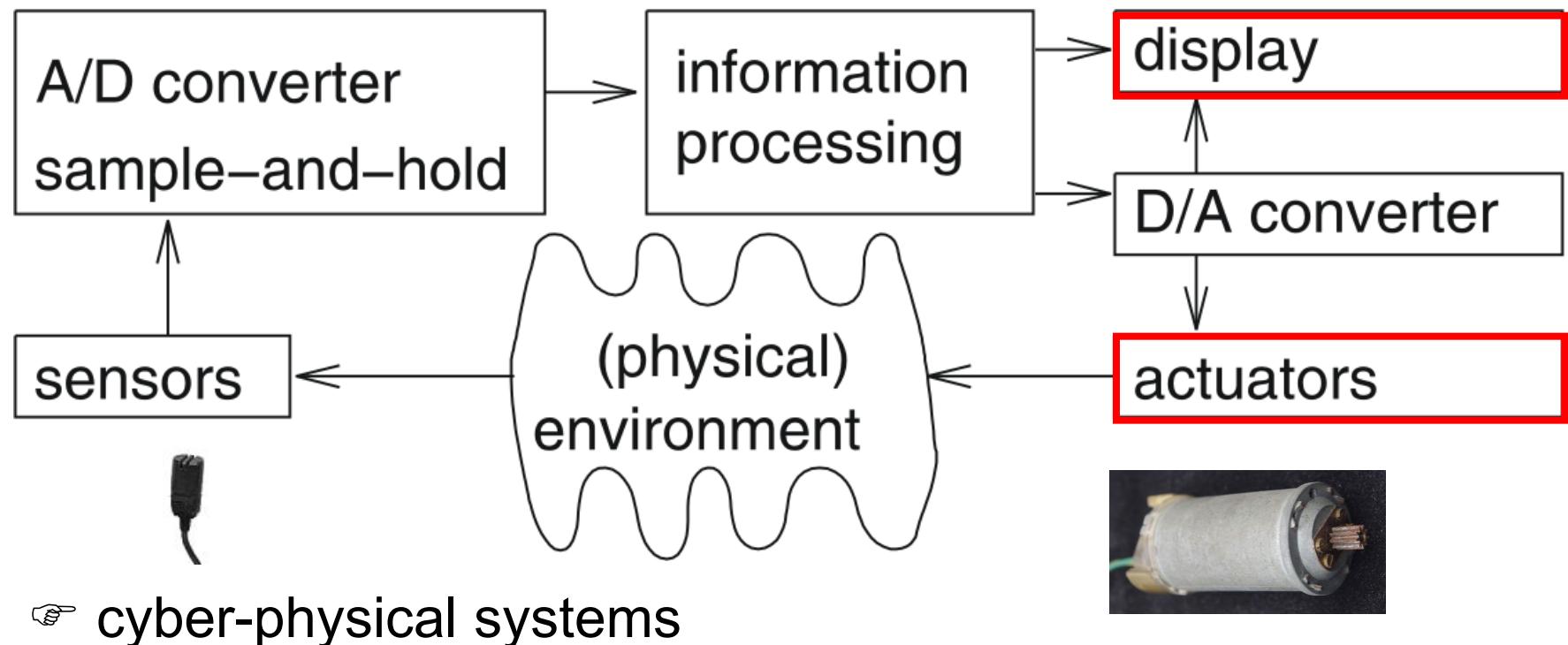
Embedded System Hardware

- Actuators and Displays-



Embedded System Hardware

Embedded system hardware is frequently used in a loop (“**hardware in a loop**”):



Displays

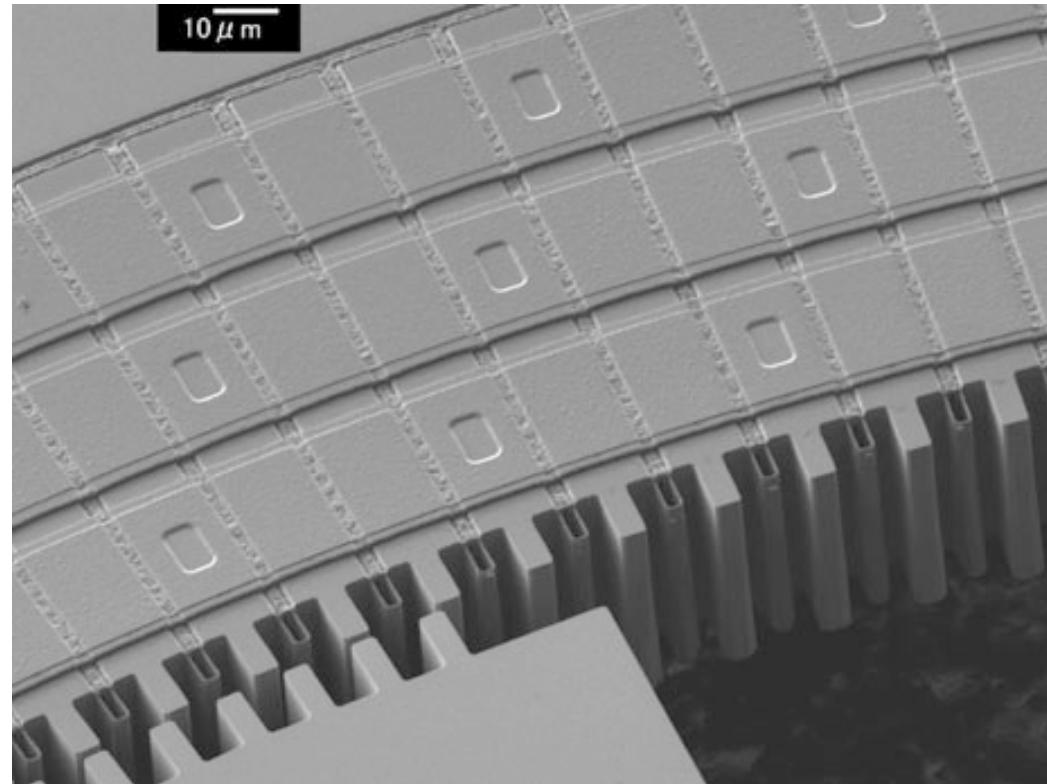
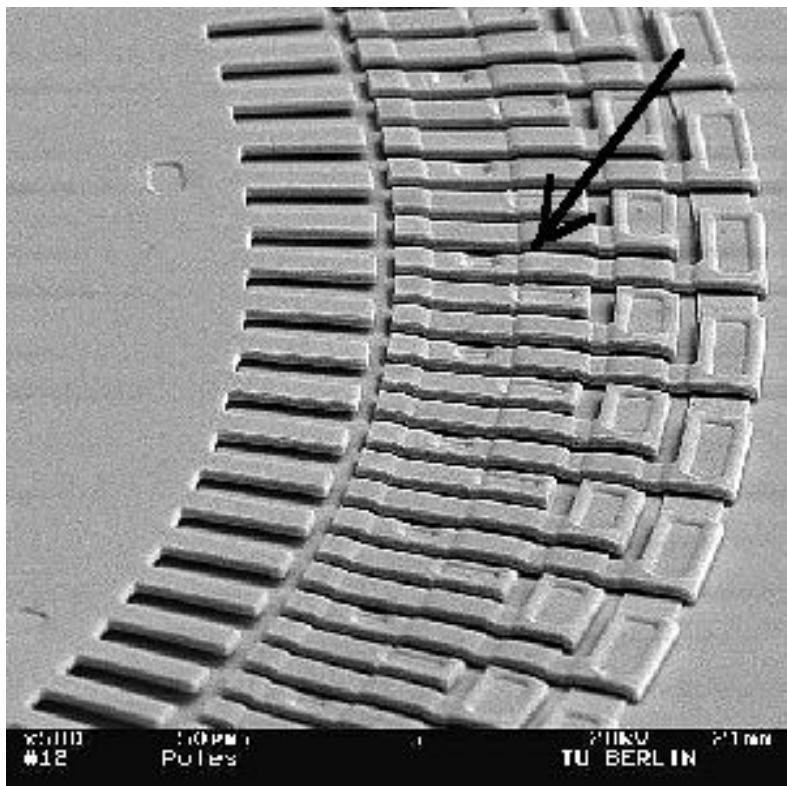
- ❖ Display technology is an area which is extremely important.
- ❖ A large amount of information exists on this technology.
- ❖ Major research and development efforts lead to:
 - New display technology such as organic displays.
 - Organic displays are emitting light and can be fabricated with very high densities. In contrast to LCDs, they do not need backlight and polarizing filters. Major changes are therefore expected in these markets.

Actuators

- ❖ Huge variety of actuators and output devices, impossible to present all of them.
 - Motor as an example:



Actuators (2)



Courtesy and ©: E. Obermeier,
MAT, TU Berlin

<http://www.piezomotor.se/pages/PWtechnology.html>

http://www.elliptec.com/fileadmin/elliptec/User/Produkte/Elliptec_Motor/Elliptecmotor_How_it_works.h

Summary

❖ Embedded Systems Hardware

- Sensors
- Information Processing
 - Processing
 - Memory
- A/D and D/A Convertor
- Actuators and Displays