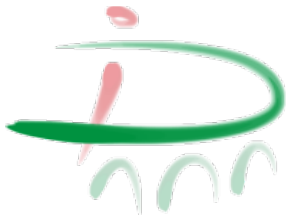
ESD Electronic Systems Design
Synthesis Verification Testing Power Communication

Progettazione di Sistemi Embedded
embbedded systems design

Franco Fummi
University of Verona
Department of Computer Science
Italy



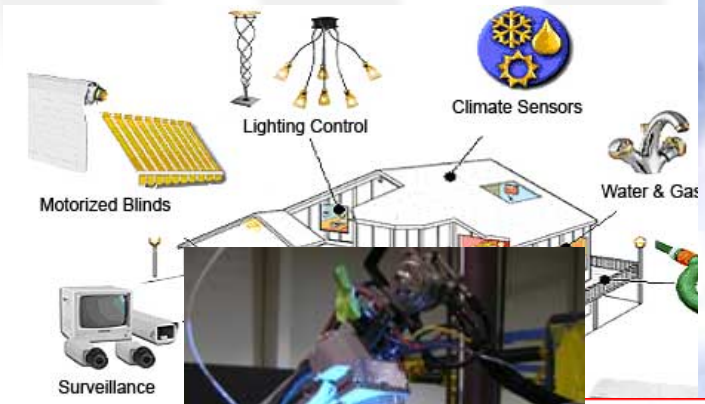
EDALab
Networked Embedded Systems

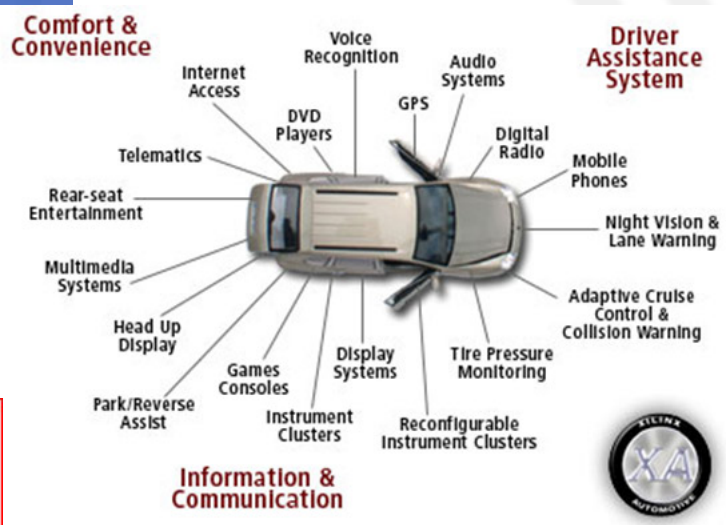
Laurea Magistrale in Ingegneria e Scienze Informatiche
Embedded Systems Design Course

Goals

- Techniques for the automatic design of embedded systems:
  - starting from their specification throughout:
    - validation / verification
    - automatic synthesis
    - testing
- This lecture is focused on:
  - most important design languages
  - most evolved tools for their manipulation

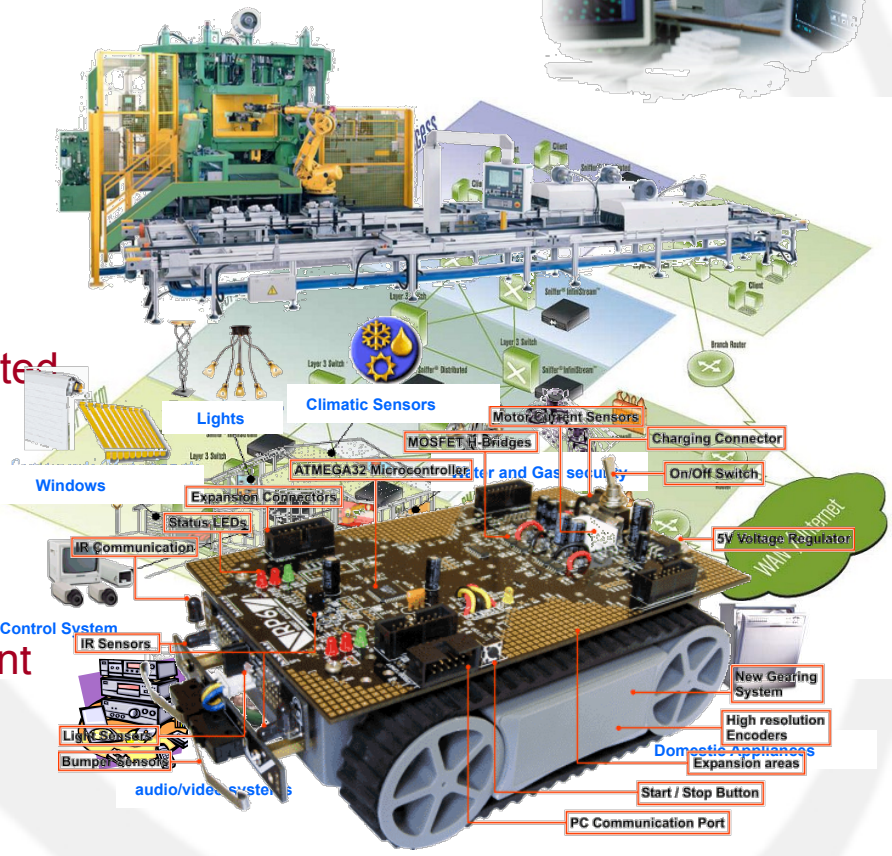
Embedded Systems: Where?





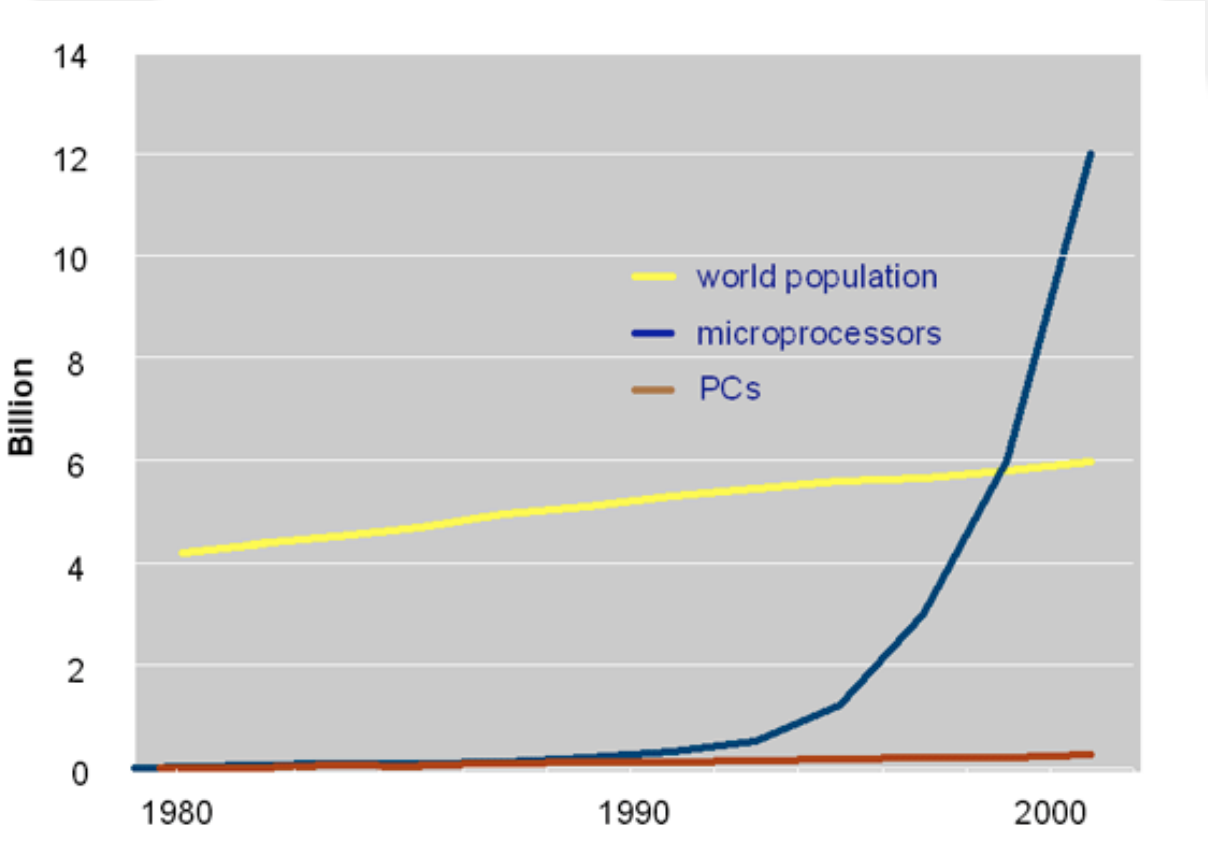
ES: Historical perspective

- From computer (’60-’80):
  - General purpose systems for solution of general problems
- To digital control systems (’80-’90):
  - Systems dedicated to control and automation
- To distributed systems (’90-’00):
  - General purpose systems and/or dedicated systems cooperating through the network
- To embedded systems (’00-):
  - Distributed systems integrated in non computing objects and in the environment
- To cyber-physical systems (’10-):
  - embedded systems integrated with physical processes

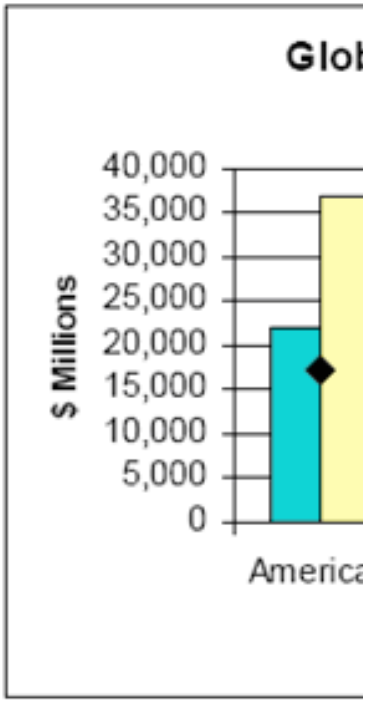


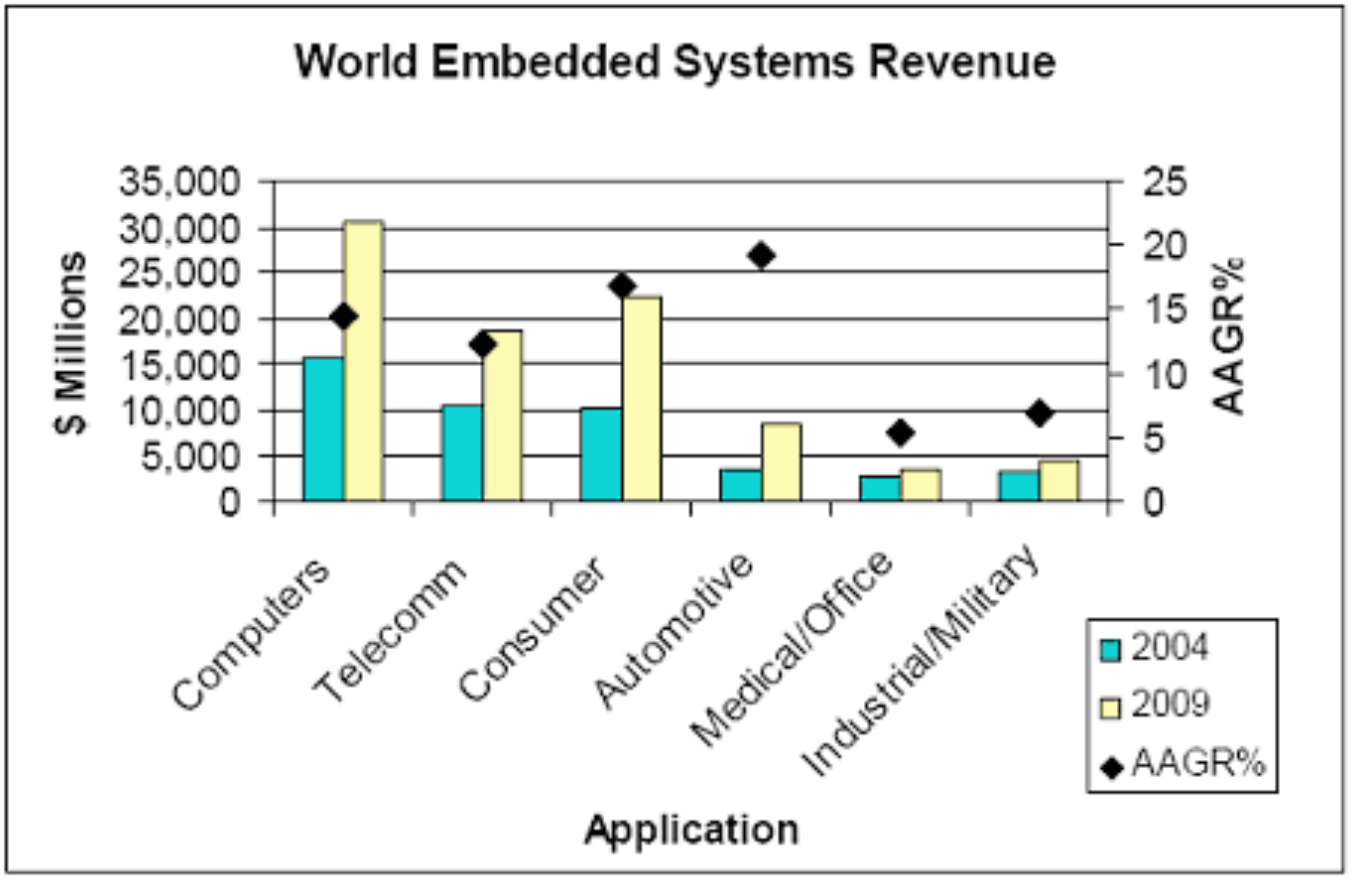
ES: History

- First comp
  Systems:
  - not show
    to the pa
    compute
- The Apollo
  the world's
  - small siz
    devoted
- Mass prod
  - 1961 witl
- No stop…



ES Market

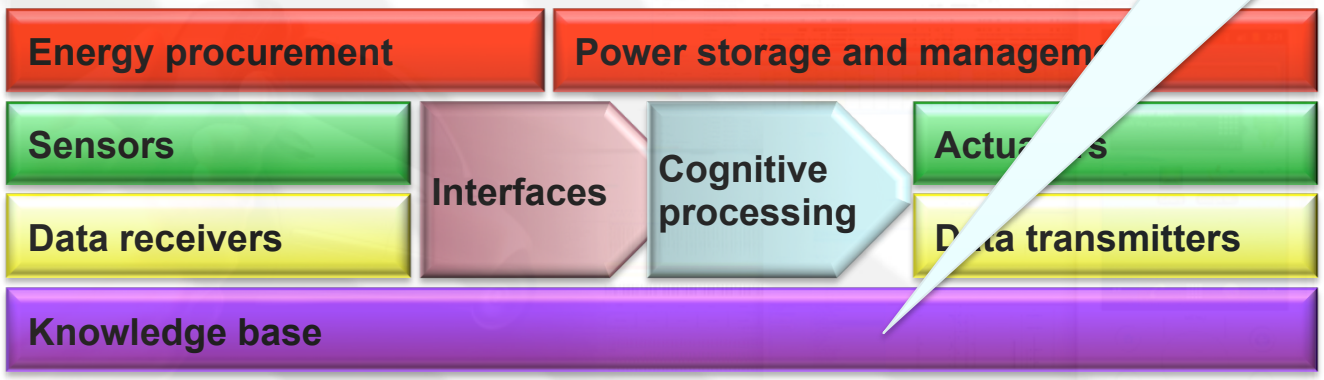




Embedded software			
Embedded IC			
Embedded boards			
Total Revenues	45,873	88,144	14.0

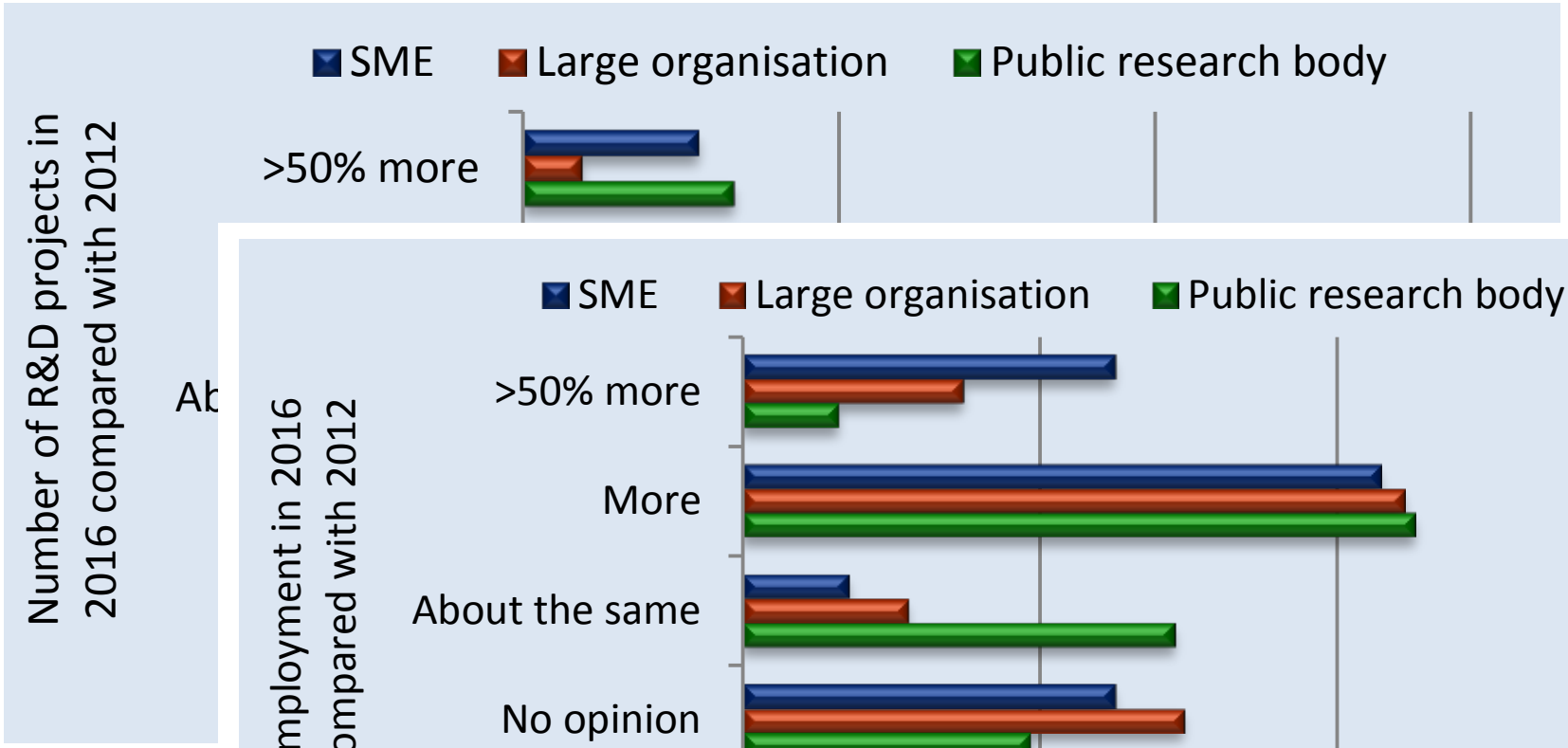
From ES to Smart Systems

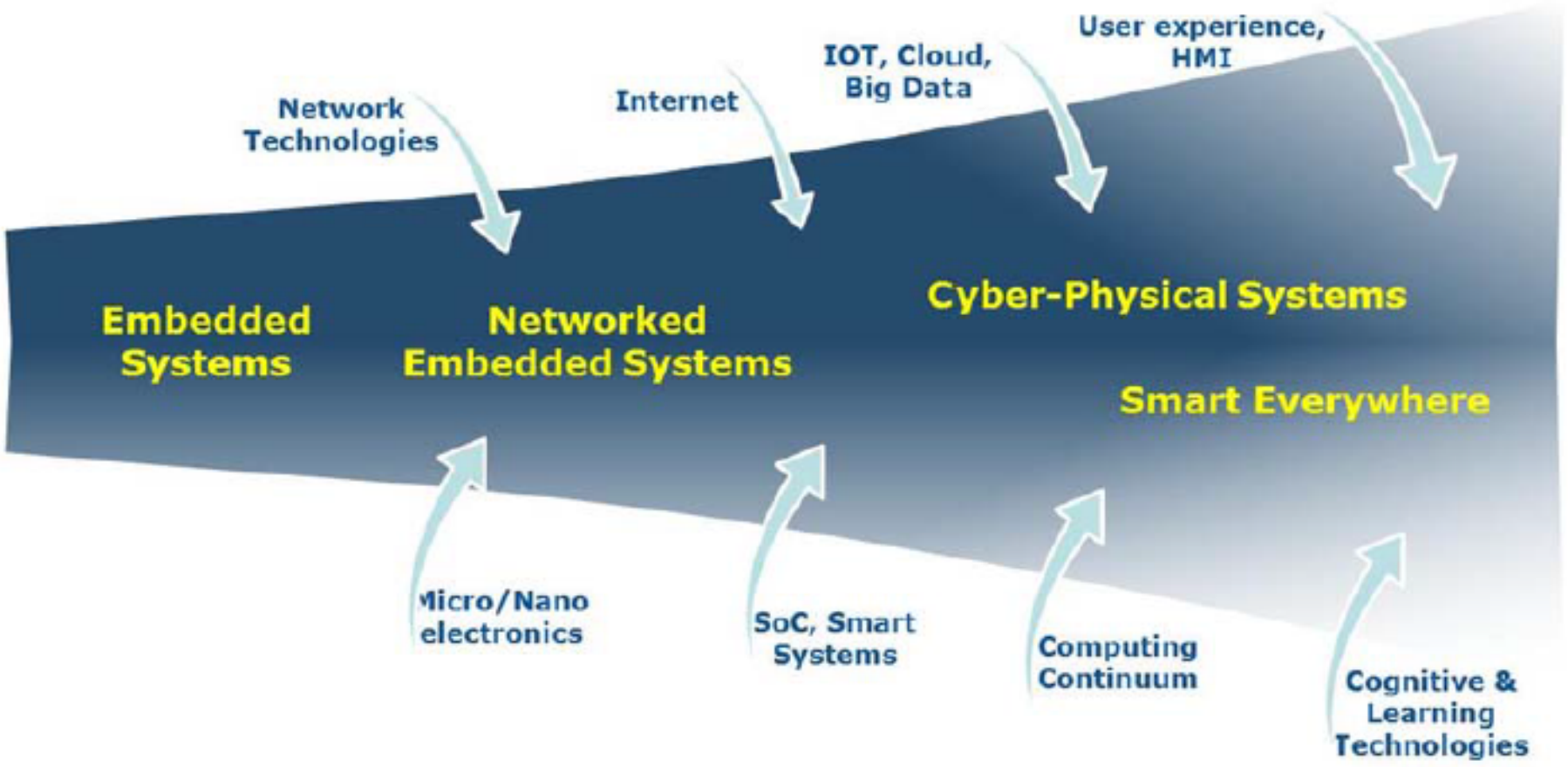
- Miniaturized self-sufficient device that
  - Incorporates **functions** of sensing, actuation, and control
  - To describe and analyze a situation, an
    decisions based on the available data
  - In a **predictive** or adaptive manner (sm
  - Energy-**autonomous** and ubiquitously



Knowledge base separates smart systems from systems which, although they may be automated, remain purely reactive

Grow in smart system R&D




Network Technologies
Internet
IOT, Cloud, Big Data
User experience, HMI
Embedded Systems
Networked Embedded Systems
Cyber-Physical Systems
Smart Everywhere
Micro/Nano electronics
SoC, Smart Systems
Computing Continuum
Cognitive & Learning Technologies

How Relevenat (I)

How Relevant (II)

- & progetti europei completati e attivi:
  - Angel, Vertigo, Coconut, C4C, Complex, SMAC, Contrex
- 2 progetti europei in FP6
  - ANGEL (mobile gateway for sensors network)
  - VERTIGO (HW formal verification)
- 5 progetti europei in FP7
  - COCONUT (embedded systems design and verification)
    - best evaluation of the overall embedded systems track
  - C4C (control for coordination of distributed systems)
  - COMPLEX (platform-based design space exploration)
  - SMAC (smart systems design)
  - CONTREX (mixed-criticality systems)

ES: How to design?

- We cannot design embedded systems like general purpose systems
  - Different design constraints, different goals
  - Embedded design is about the system, not about the computer
- E.g.
  - In general purpose computing, design often focuses on building the fastest CPU
  - In embedded systems the CPU simply exists as a way to implement control algorithms communicating with sensors and actuators

ES: Design constraints

- Size and weight
  - Hand-held electronics
  - Weight costs money in transportation
  - Human body cannot eat desktops
- Power
  - Buttery power instead of AC
- Harsh environment
  - Power fluctuation, RF interferences, heat, vibration, water, …
- Safety critical and real time operations
- Low costs

ES: Designer knowledge

- HW architecture alternatives
  - for a correct HW/SW trade-off
- SW design skills
  - lots of languages continuously extending
- HW/SW interaction mechanisms
  - O.S., MW, HdS for efficient SW development
- Network infrastructure
  - all ES are now networked embedded systems
- Computation effort estimation
  - theory is important when used in practice
- Join 3C: computation, control & communication

Course Structure

- 34 lectures:
  - 32 theory hours
    - 22 lectures
  - 24 practical hours
    - 12 lectures
- People:
  - Franco Fummi (theory)
  - Michele Lora (laboratory class)
  - … for practical elaborations

Modalità di Esame (I)

- Teoria + lab. + opzioni:
  - teoria
    - scritto con votazione /30
  - relazione laboratorio
    - +3 punti max
  - on demand
    - elaborato 0 +∞
    - (orale) +3 -∞
- Regole generali:
  - elaborato dura 1 anno accademico
  - consegna in date stabilite

Modalità di Esame (II)

- Alternative:
  - Elaborato personale
    - stage aziendale
    - tesi
  - Teoria
    - no way :-)
- Design&Reuse:
    - tesi
    - stage pre-tesi

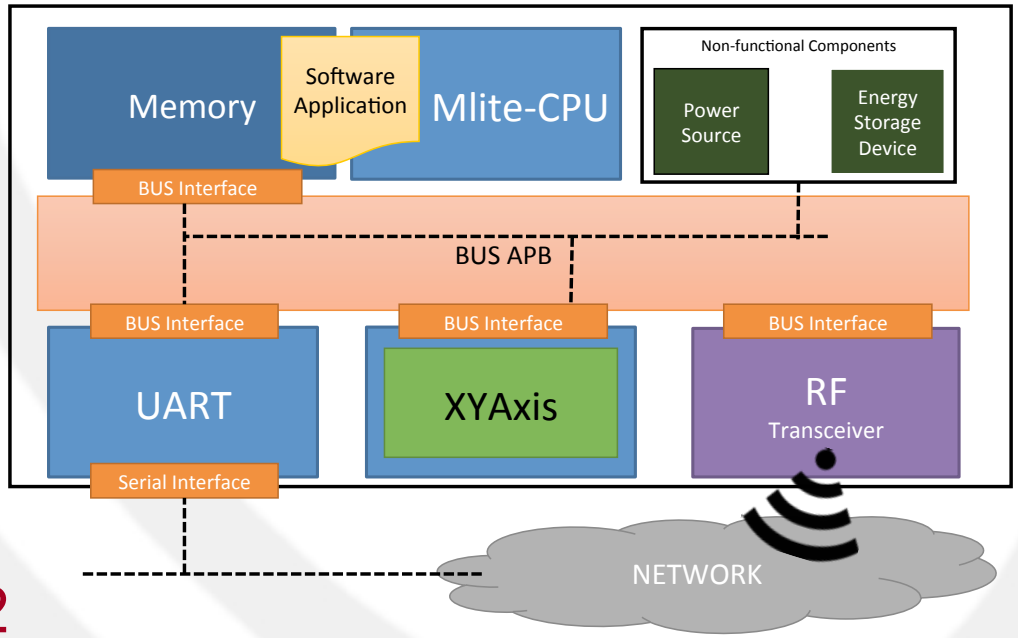
COMPUTER SCIENCE PARK

Pre/post Condizioni

- Precedenze Indispensabili:
  - Architettura degli Elaboratori
  - Programmazione
  - Linguaggi ...
  - Sistemi (Metodi di specifica)
- Fondamentale per
  - Curriculum sistemi embedded (magistrale in Ingegneria)
    - Sistemi operativi avanzati, Architetture avanzate, Software per Sistemi Embedded, Sistemi Embedded Multimediali, Sistemi Embedded di Rete…

A Challenging Benchmark

- Smart devices:
  - The Open Source Test Case (SMAC project)
- Laboratorio Ciberfisico:
  - Secondo piano CV2



Detailed Program

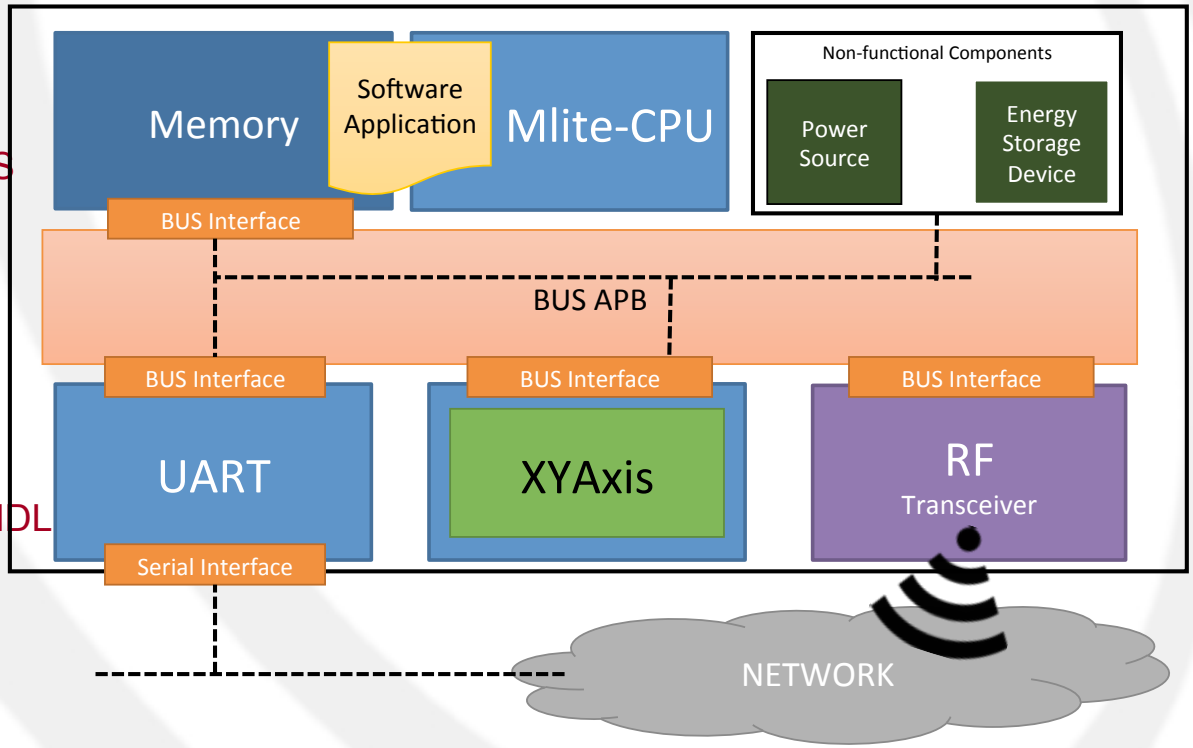
week	data	day	lecture	lab.	topic
1	2-Oct	Fri.	3		Course introduction; Embedded systems modeling
2	7-Oct	Wed.	2		Embedded systems modeling II; SystemC-based design
2	9-Oct	Fri.	3		SystemC-based design II; SystemC-based design III
3	14-Oct	Wed.		2	SystemC modeling at RTL
3	16-Oct	Fri.	3		Platform-based design; Transactional-based design; TLM 2.0 standard
4	21-Oct	Wed.		2	SystemC compilation/execution/debugging
4	23-Oct	Fri.	3		TLM 2.0 standard II; SystemC/AMS support
5	28-Oct	Wed.		2	SystemC timing evolution
5	30-Oct	Fri.			An embedded architecture for cars managing
6	4-Nov	Wed.		2	SystemC modeling at TLM
6	6-Nov	Fri.	3		High-level synthesis (HLS): scheduling; High-level synthesis: allocation
7	11-Nov	Wed.		2	SystemC/AMS
7	13-Nov	Fri.	3		Software embedded synthesis; Model-based design (MBD) of embedded software; HMI design
8	18-Nov	Wed.		2	SystemC/AMS and IP-Xact
8	20-Nov	Fri.			**intermediate exam**
9	25-Nov	Wed.		2	Mixed RTL/TLM/AMS SystemC
9	27-Nov	Fri.			Cyber-physical systems: models of computations
10	2-Dec	Wed.		2	Platform, testbench and device driver (OSTC)
10	4-Dec	Fri.	3		VHDL introduction; VHDL syntax
11	9-Dec	Wed.		2	Embedded software design
11	11-Dec	Fri.	3		VHDL modeling; VHDL timing simulation
12	16-Dec	Wed.		2	VHDL modeling at RTL
12	18-Dec	Fri.	2		VHDL timing simulation II; VHDL synthesis
13	8-Jan	Fri.			GPGPU: design problems and opportunities
14	13-Jan	Wed.		2	VHDL timing simulation
14	15-Jan	Fri.	2		Networked embedded systems (NES); Middleware for embedded systems
15	20-Jan	Wed.		2	Automatic synthesis from RTL and TLM
15	22-Jan	Fri.	2		Introduction to embedded systems verification; Introduction to embedded systems testing
16	27-Jan	Wed.			NO
16	29-Jan	Fri.			**final exam**
	hours	56	**32**	**24**	
	credits	6,0	**4,0**	**2,0**	

Topics (theory)

- Specification:
  - Embedded systems modeling
  - SystemC-based design
  - TLM design introduction
  - TLM 2.0 standard
  - VHDL modeling
  - VHDL syntax
  - Networked ES (NES)
  - Smart systems
- HW synthesis:
  - Introduction to TLM design
  - High-level synthesis
  - Automatic VHDL synthesis
- SW synthesis:
  - Embedded software generation
  - Automatic device driver generation
  - Middleware for embedded systems
  - Model-based design
  - HMI deisgn
- Verification & testing:
  - Introduction to verification
  - Introduction to testing
  - VHDL timing simulation
  - RTL-TLM mixed simulation
  - Embedded software verification

Topics (lab.)

- Specification:
  - Compiling / executing /debugging SystemC
  - Modeling SystemC TLM
  - Modeling SystemC RTL
  - Timing evolution in SystemC
  - Analog modeling in SystemC/AMS
  - Platforms and IP-Xact
  - Mixed modeling RTL/TLM/AMS
  - Timing modeling in VHDL
- Hardware synthesis:
  - Automatic synthesis from TLM
  - VHDL modeling at RT
  - Automatic synthesis from RTL VHDL
- Software synthesis:
  - Testbench and device driver
  - Model based design: radCASE
  - Embedded software design



Teaching supports (I)

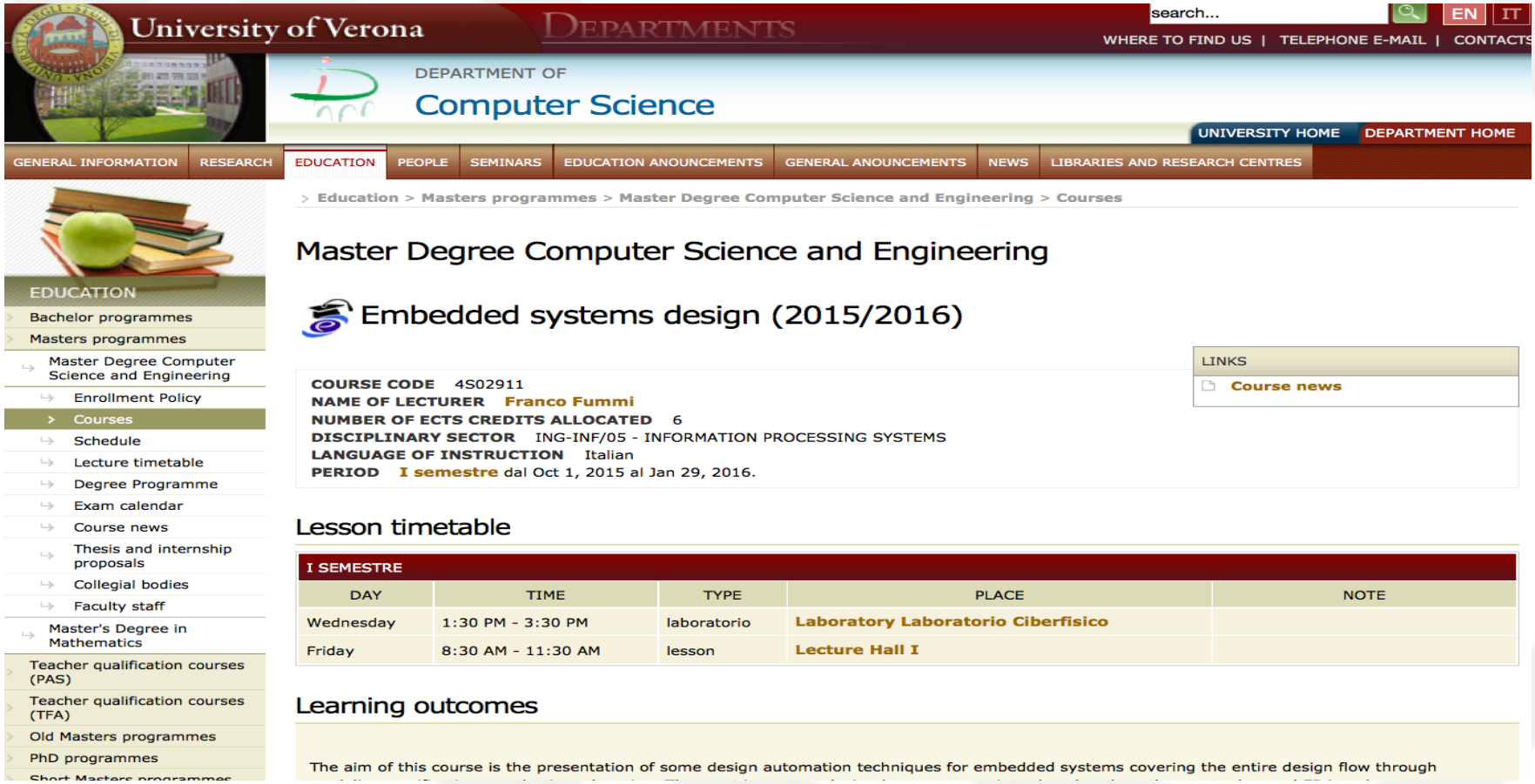
- Course web page
  - Detailed program
  - Complete program
- E-learning web page
  - Slides
  - Laboratory instructions
  - Questions/answers
- Seminars
  - Indications during the course

Teaching supports (II)

- Theory slides:
  - 0.CourseIntroduction
  - 1.EmbeddedSystemsModeling
  - 2.SystemCBasedDesignFlow
  - 3.PlatformBasedDesign
  - 4.TLMBasedDesign
  - 5. SystemC/AMS
  - 6.HighLevelSynthesis
  - 7.EmbeddedSoftware
  - 8.ModelBasedDesign
- Theory slides:
  - 9.VHDLDesignIntroduction
  - 10.VHDLSyntax
  - 11.VHDLSpecification
  - 12.VHDLSimulation
  - 13.VHDLSynthesis
  - 14.NESDesign
  - 15.SmartSystems
  - 16.VerificationAndTesting

More information

http://www.di.univr.it/~fummi



For the stronger ...

7994

Tuesday
8:30 – 10:30

In the
corridors...
running

franco.fummi@univr.it

For the strongest…

7048

Monday
10.00 – 11.00

On the e-learning

michele.lora@univr.it