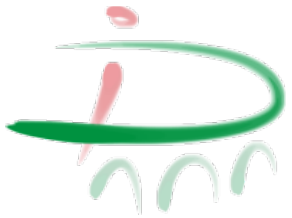
ESD Electronic Systems Design
Synthesis Verification Testing Power Communication

Progettazione di Sistemi Embedded
embbedded systems design

Franco Fummi
University of Verona
Department of Computer Science
Italy



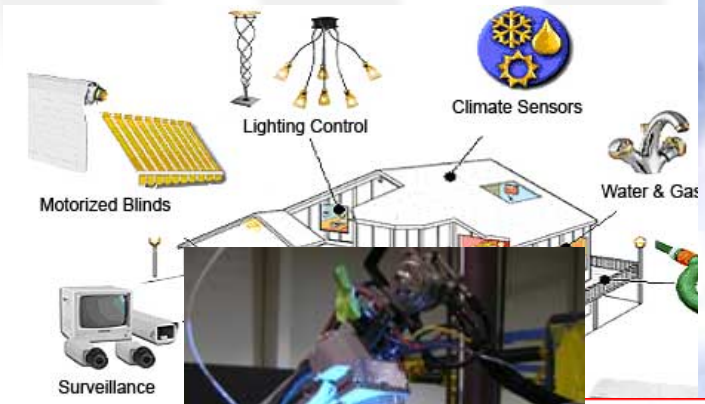
EDALab
Networked Embedded Systems

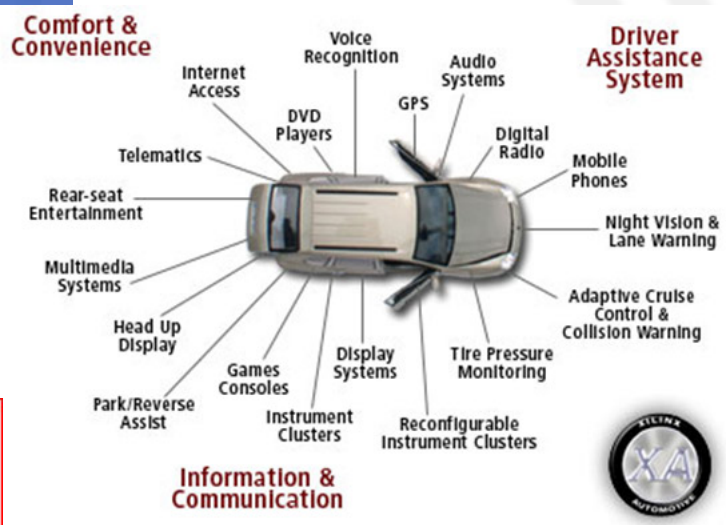
Laurea Magistrale in Ingegneria e Scienze Informatiche
Embedded Systems Design Course

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

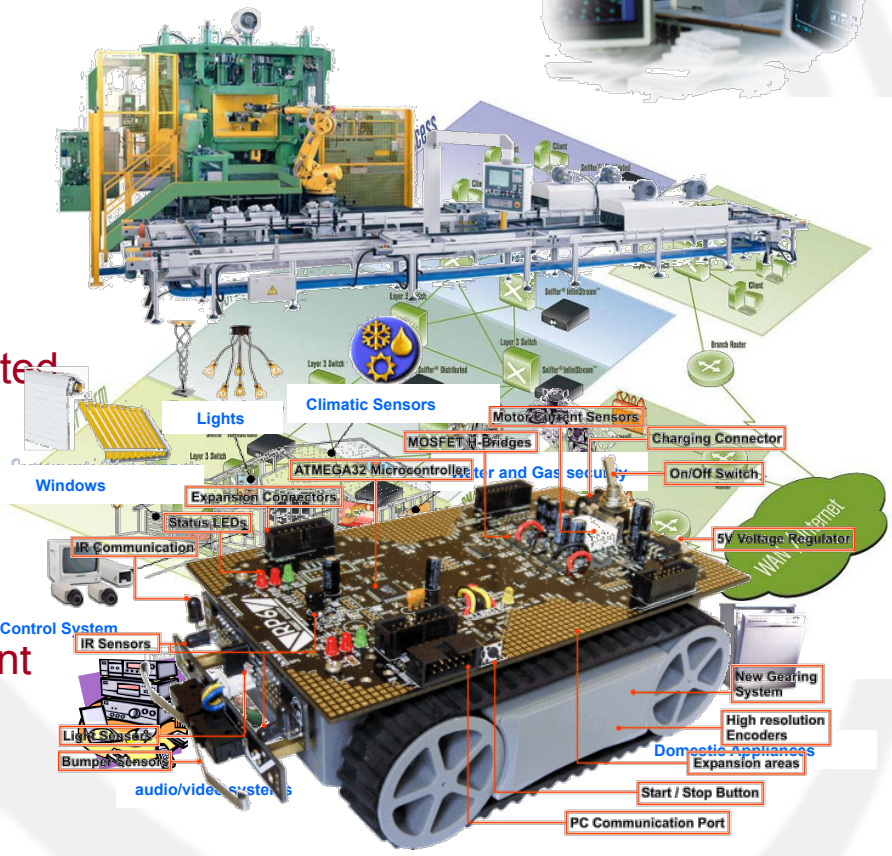
Embedded Systems: Where?





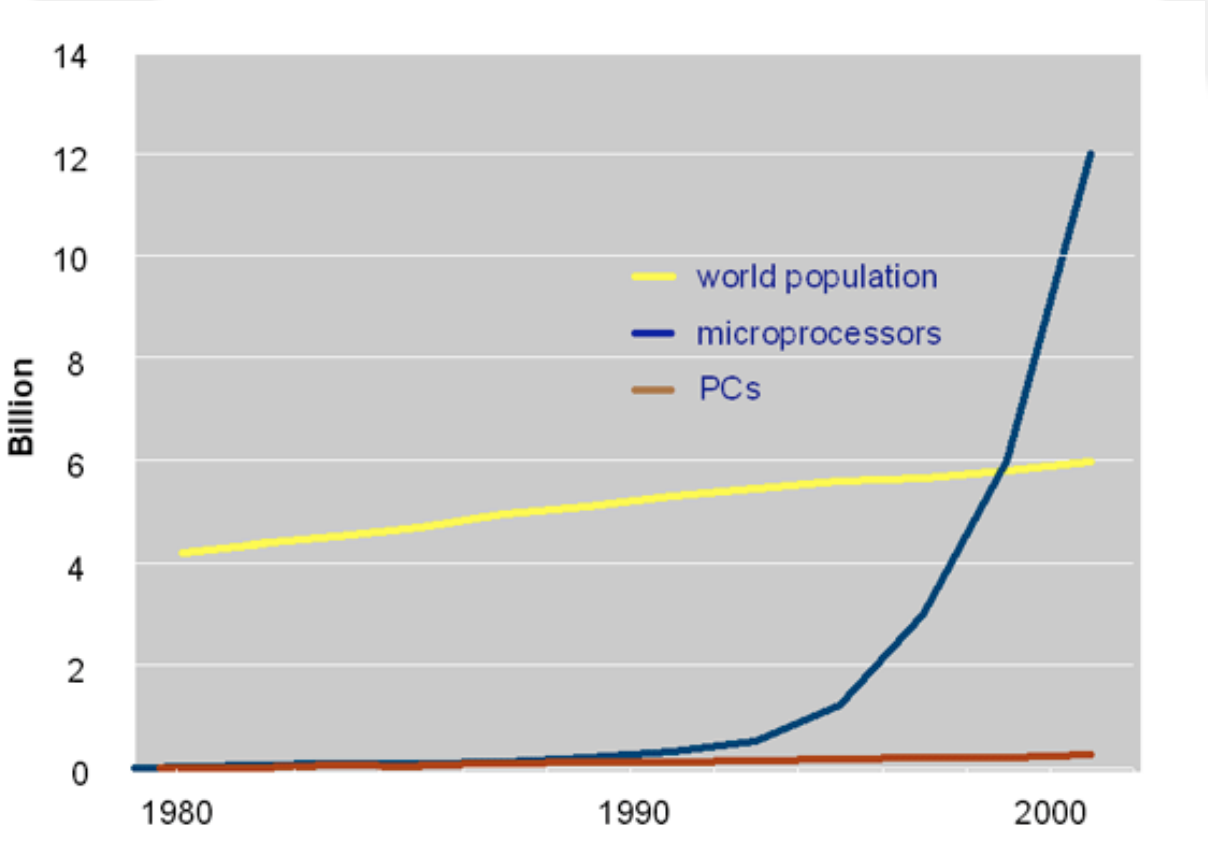
ES: Historical perspective

- 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 nor computing objects and in the environment
- To cyber-physical systems (’10-):
  - embedded systems integrated with physical processes

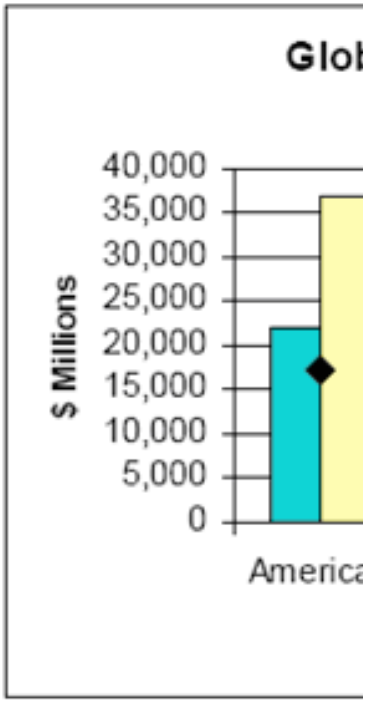


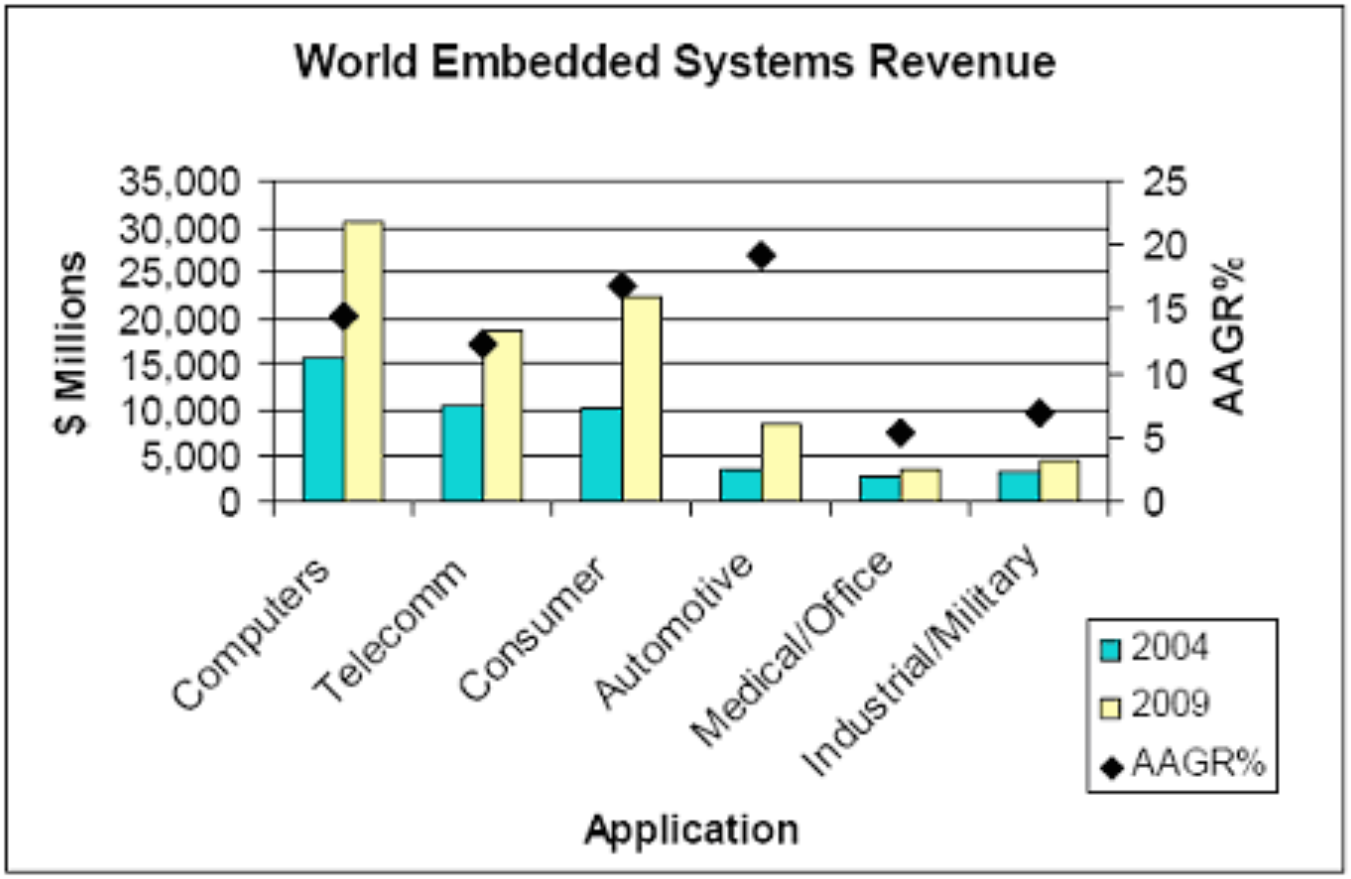
ES: History

- 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



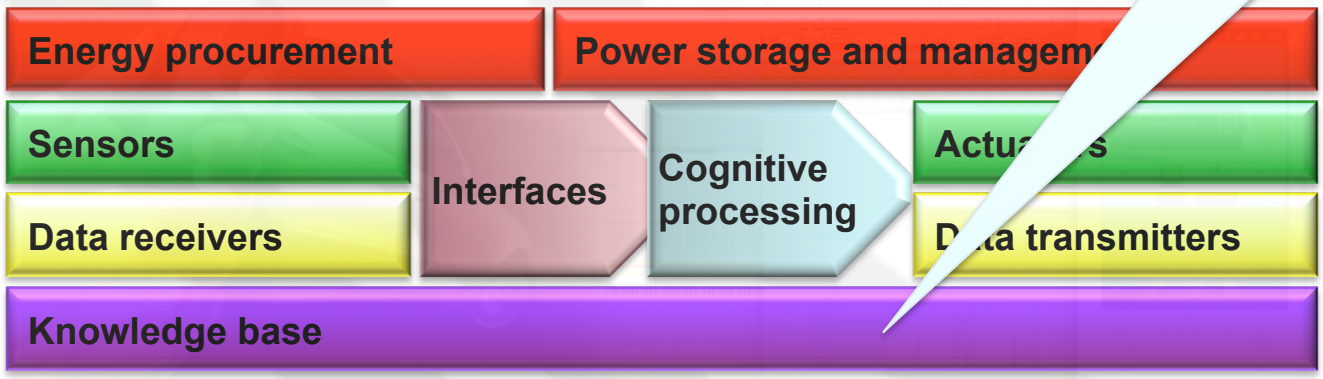


Embedded software			
Embedded IC			
Embedded boards			
Total Revenues	45,873	88,144	14.0

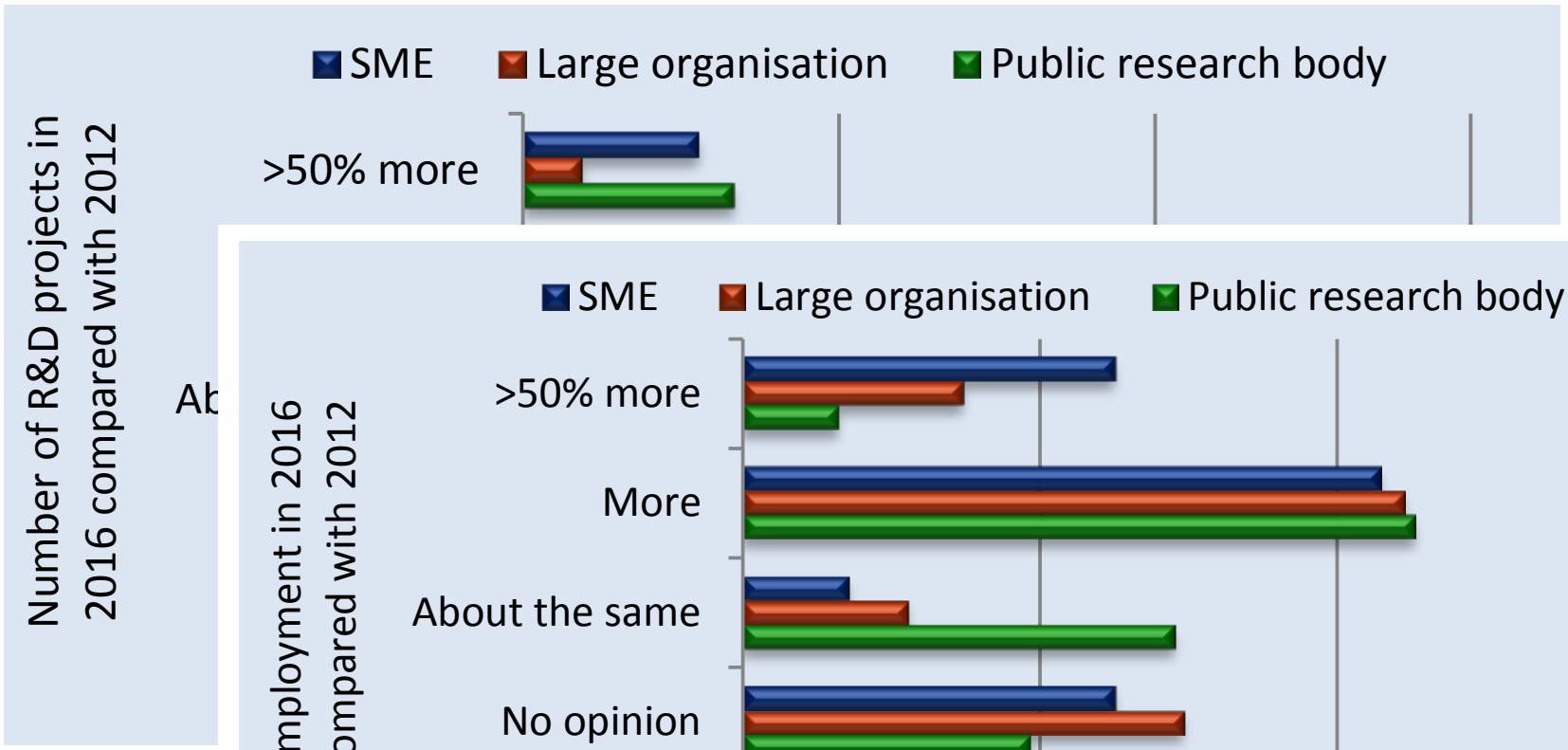
From ES to Smart Systems

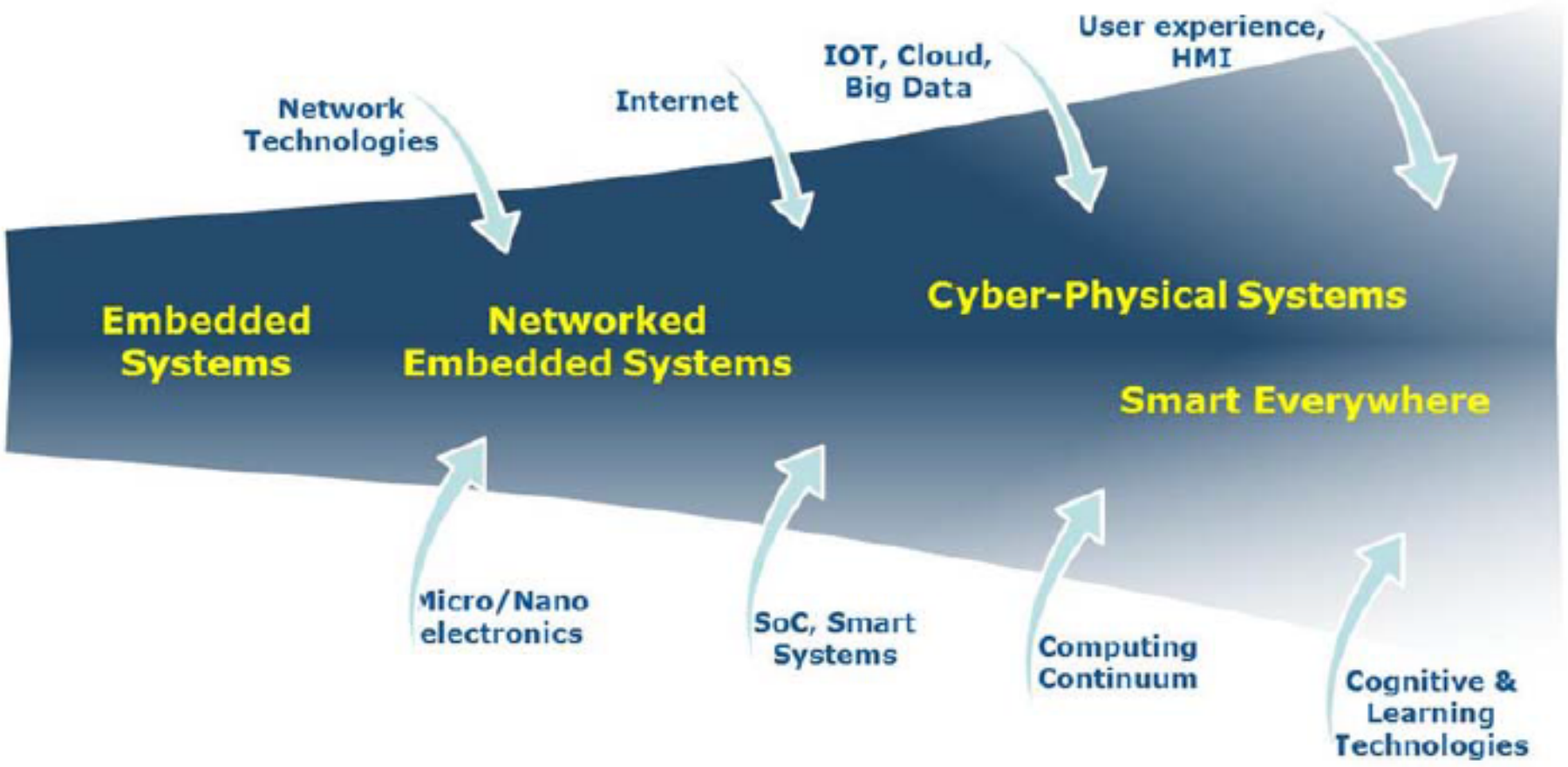
- Miniaturized self-sufficient device that
  - Incorporates **functions** of sensing, actuation, and control
  - To describe and analyze a situation, an **decisions** based on the available data
  - In a **predictive** or adaptive manner (sm
  - Energy-**autonomous** and ubiquitously

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:
  - relazione 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

EDALAB Networked Embedded Systems

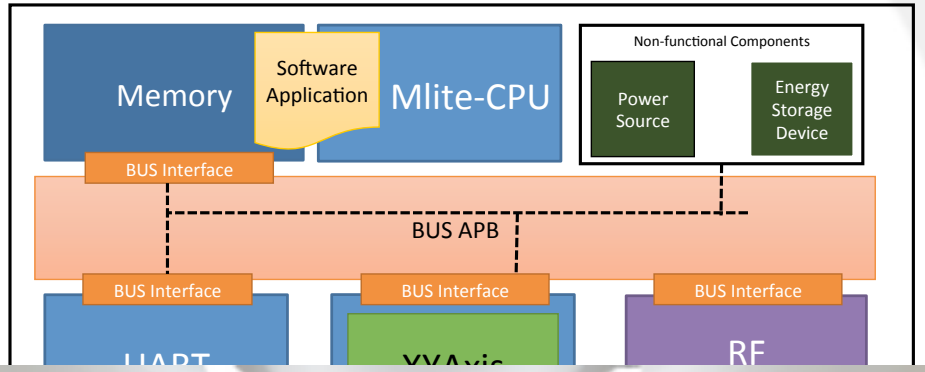
COMPUTER SCIENCE PARK

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…

Benchmark and Labs.

- Smart devices:
  - The Open Source Test Case (SMAC project)
- Laboratorio Ciberfisico:
  - Secondo piano CV
- Lab. NES/Parco

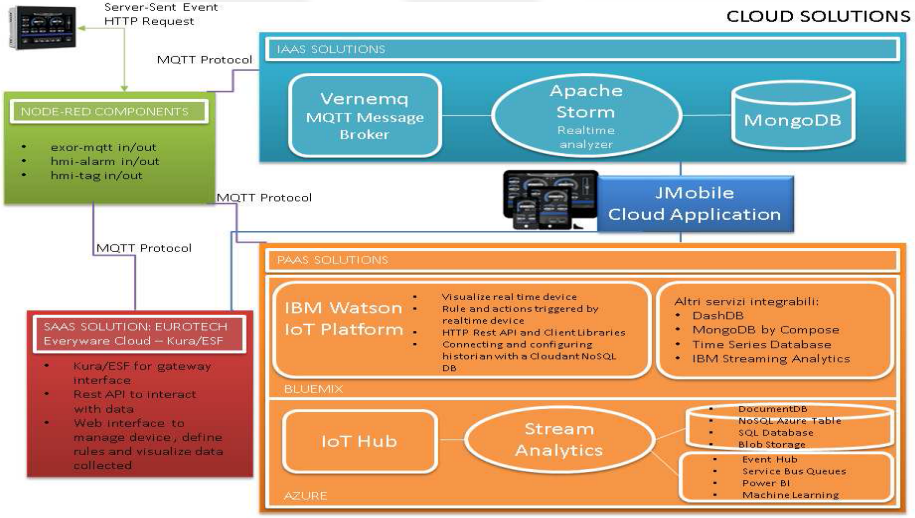


2016 News

Semiformal Assertion Based Verification of Hardware/Software Systems in a ModelDriven Design Framework
Pravadelli, G., Quaglia, D., Vinco, S., Fummi, F.

MATLAB SIMULINK

fmi FUNCTIONAL MOCK-UP INTERFACE



ICEAA - IEEE APWC
September 11-15, 2017, Verona, Italy

FDL Forum on specification & Design Languages
September 18-20 | Verona, Italy

Detailed Program

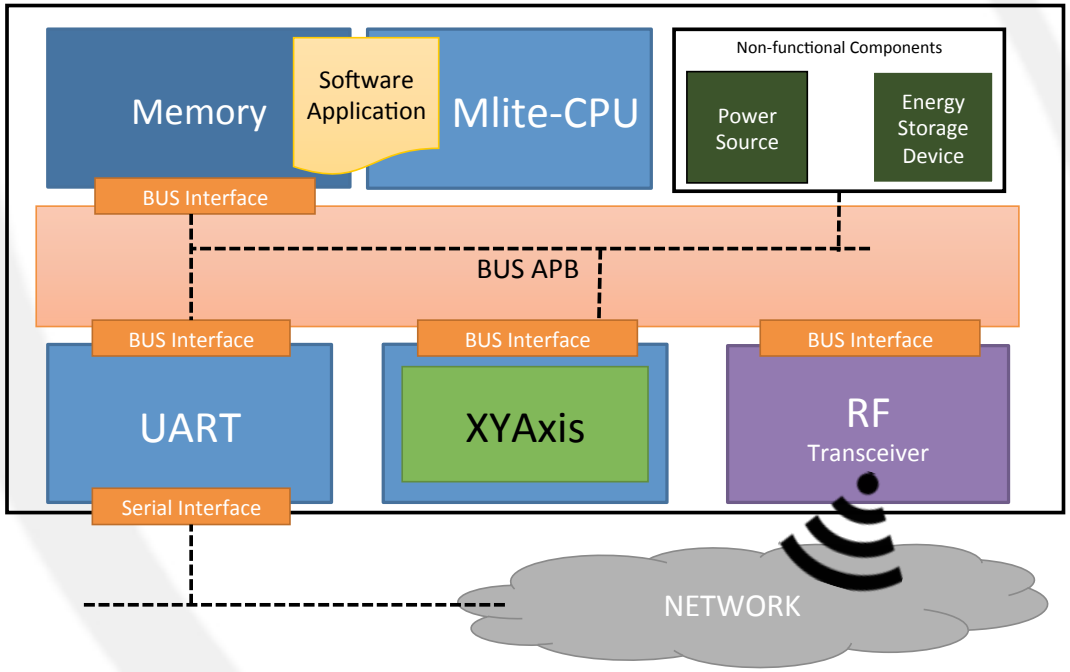
week	data	day	lecture	lab.	topic
1	5-Oct	Wed.	2		Course introduction; Embedded systems modeling
1	7-Oct	Fri.	3		Embedded systems modeling II; SystemC-based design
2	12-Oct	Wed.	2		SystemC-based design II; SystemC-based design III
2	14-Oct	Fri.	3		Platform-based design; Transactional-based design; TLM 2.0 standard
3	19-Oct	Wed.		2	SystemC compilation/execution/debugging
3	21-Oct	Fri.	3		TLM 2.0 standard II; SystemC/AMS support
4	26-Oct	Wed.		2	SystemC modeling at RTL
4	28-Oct	Fri.		2	SystemC modeling at TLM
5	2-Nov	Wed.		2	SystemC timing evolution
5	4-Nov	Fri.		2	SystemC/AMS
6	9-Nov	Wed.		2	Mixed RTL/TLM/AMS SystemC
6	11-Nov	Fri.	3		High-level synthesis (HLS): scheduling; High-level synthesis: allocation
7	16-Nov	Wed.		2	Platform, testbench and device driver (OSTC)
7	18-Nov	Fri.	3		Software embedded synthesis; Model-based design (MBD) of embedded software; IoT and Cloud
8	23-Nov	Wed.			Cyber-physical systems: models of computations
8	25-Nov	Fri.			**intermediate exam**
9	30-Nov	Wed.		2	Model-based design: Matlab/Simulink/FMI
9	2-Dec	Fri.	3		VHDL introduction; VHDL syntax
10	7-Dec	Wed.		2	Embedded software design
10	9-Dec	Fri.	3		VHDL modeling; VHDL timing simulation
11	14-Dec	Wed.		2	VHDL modeling at RTL
11	16-Dec	Fri.	2		VHDL timing simulation II; VHDL synthesis
12	21-Dec	Wed.		2	VHDL timing simulation
12	23-Dec	Fri.	3		Networked embedded systems (NES); Smart systems
13	11-Jan	Wed.		2	Automatic synthesis from TLM and RTL
13	13-Jan	Fri.	2		Introduction to embedded systems verification; Introduction to embedded systems testing
14	18-Jan	Wed.			final report preparation
14	20-Jan	Fri.			IoT and Cloud architectures; GPGPU: design problems and opportunities
15	25-Jan	Wed.			final report preparation
15	27-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
  - AMS modeling
  - VHDL modeling, 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
  - Model-based design
  - **IoT and Cloud**
- Verification & testing:
  - Introduction to verification
  - Introduction to testing
  - VHDL timing simulation
  - **FMI/FMU simulink**

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
  - Embedded software design
  - FMI/FMU cosimulation



MATLAB SIMULINK

fmi FUNCTIONAL MOCK-UP INTERFACE

Teaching supports (I)

- Course web page
  - Detailed program
  - Complete program
- E-learning web page
  - Slides
  - Laboratory instructions
  - Questions/answers
- Book
  - Ongoing
- 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

DIPARTIMENTO DI
Informatica

HOME ATENEO | HOME DIPARTIMENTO

INFORMAZIONI GENERALI | RICERCA | DIDATTICA | PERSONE | SEMINARI | PRIMO PIANO DIDATTICA | PRIMO PIANO | AVVISI | STRUTTURE

DIDATTICA
- Corsi di laurea
- Corsi di laurea magistrale
  - Laurea magistrale in Ingegneria e scienze informatiche
    - Modalità iscrizioni
    - Insegnamenti
    - Calendario didattico
    - Orario lezioni
    - Piani didattici
    - Calendario esami
    - Avvisi del corso di studio e degli insegnamenti
    - Proposte tesi e stage
    - Organi collegiali e di governo
    - Docenti
  - Laurea magistrale in Mathematics
  - Laurea Magistrale in Medical bioinformatics
- Percorsi Abilitanti Speciali
- Tirocini Formativi Attivi
- Corsi di laurea magistrale / specialistica (a esaurimento / disattivati)
- Dottorati di ricerca
- Master
- Dottorati di Ricerca Interateneo
- Scuole di Dottorato collegate

Didattica > Corsi di laurea magistrale > Laurea magistrale in Ingegneria e scienze informatiche > Insegnamenti

Laurea magistrale in Ingegneria e scienze informatiche

Progettazione di sistemi embedded (2016/2017)

PAGINE COLLEGATE
Avvisi relativi al corso

CODICE INSEGNAMENTO 4S02911
DOCENTE Franco Fummi
CREDITI 6
SETTORE DISCIPLINARE ING-INF/05 - SISTEMI DI ELABORAZIONE DELLE INFORMAZIONI
LINGUA DI EROGAZIONE Italiano
PERIODO I sem. dal 3-ott-2016 al 31-gen-2017.

Orario lezioni

I SEM.

GIORNO	ORA	TIPO	LUOGO	NOTE
mercoledì	13.30 - 15.30	laboratorio	Laboratorio didattico Laboratorio Ciberfisico	dal 10-ott-2016 al 31-gen-2017
venerdì	8.30 - 11.30	lezione	Aula I	

Obiettivi formativi

Tecniche per la progettazione automatica di sistemi embedded a partire dalla loro specifica per passare attraverso la verifica, la sintesi automatica e il collaudo. Il corso presenta i principali linguaggi per affrontare questo progetto e i più avanzati strumenti automatici per la loro manipolazione.

Programma

Introduzione ai sistemi embedded: definizione dei campi di applicazione, caratteristiche generali, caratteriste comuni.

Modellazione di sistemi embedded: problematiche generali della modellazione dei sistemi embedded, linguaggi per la descrizione dei sistemi embedded

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