Politecnico di Milano

Embedded Systems 1 - Ms
Advanced Operating Systems (AOS) - Ms

Energy aware design of computing systems and applications (PhD course)

Anno Accademico 2015-2016

Lecturer:

Prof. William Fornaciari

Politecnico di Milano, DEIB

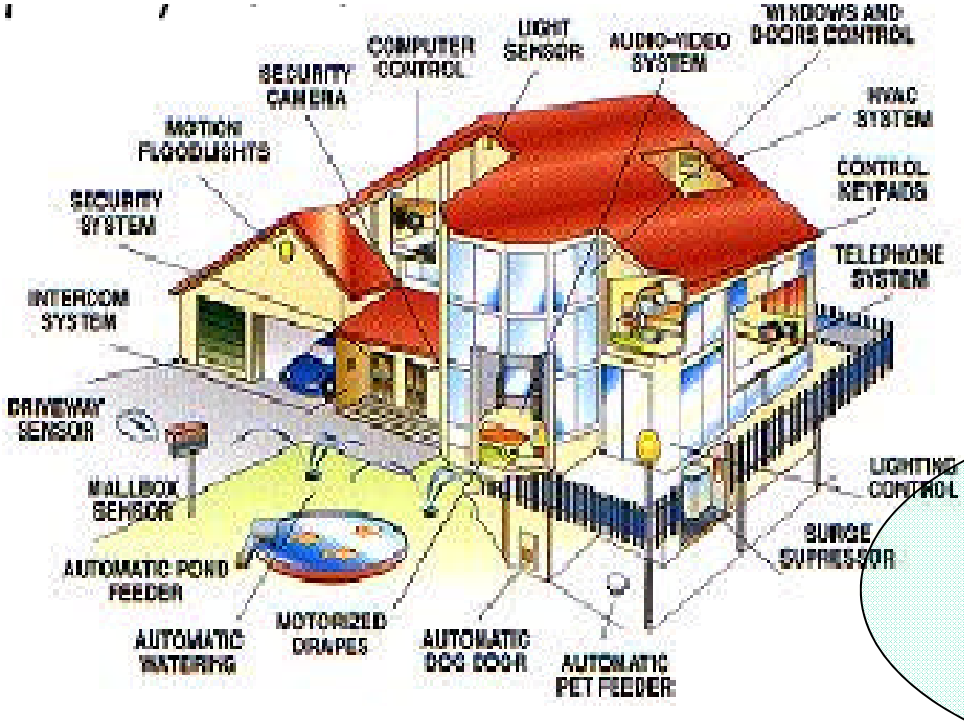
william.fornaciari@polimi.it

home.deib.polimi.it/fornacia

Embedded Systems Everywhere

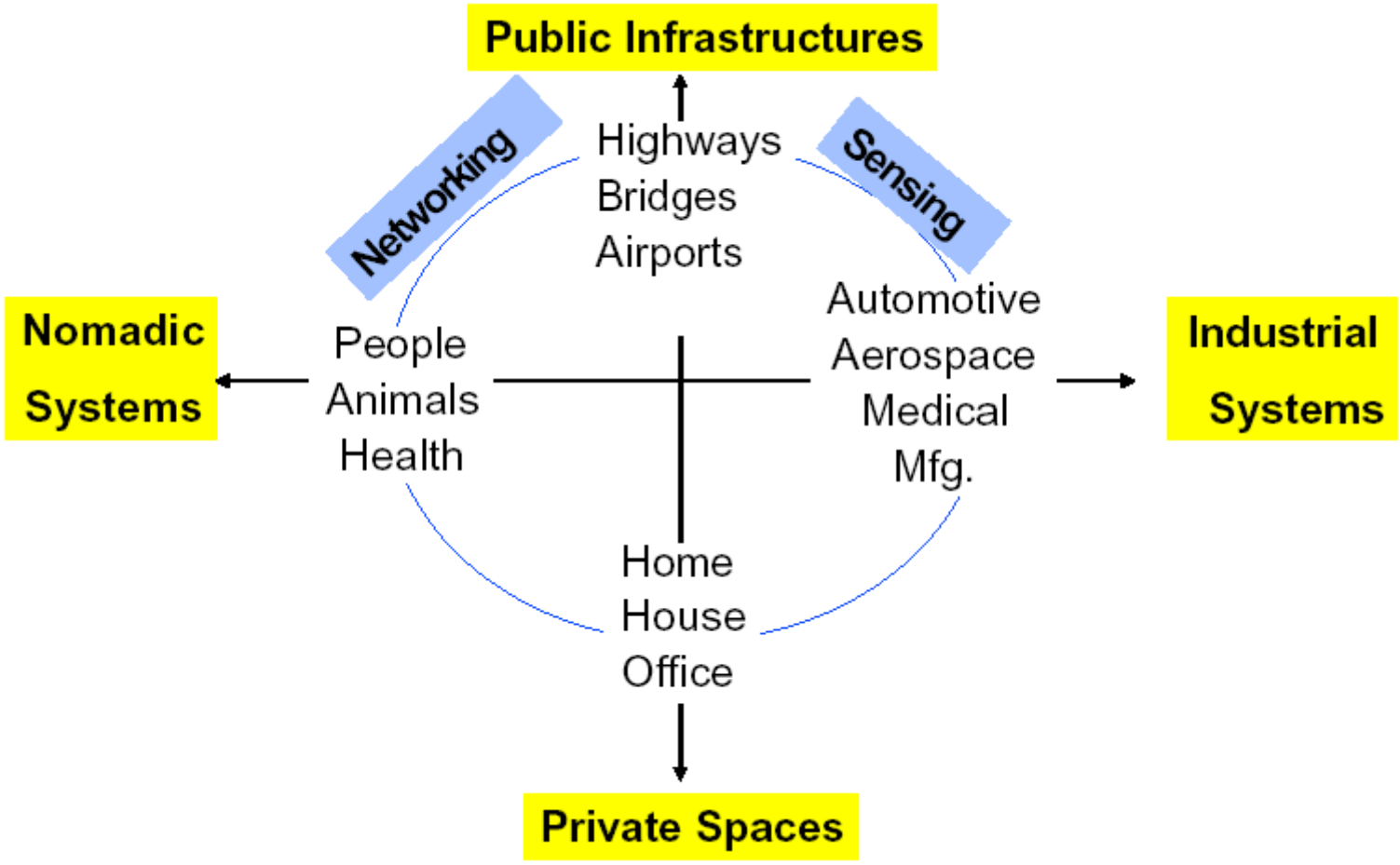
Networked Embedded Intelligence

- Enabling transportation, infrastructure industries
- Leading to revolutions like the digital home
- Turning ambient dreams into reality
- Enabling sensor networks improving our quality of life



Ubiquitous
Low Power
High performance
Interconnected

Four main application contexts



Embedded Systems 10 years from now

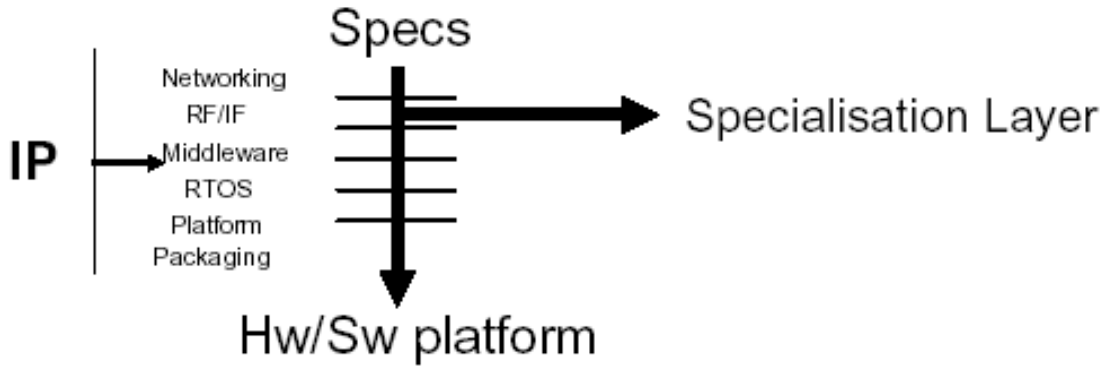
- Networked: from working in isolation towards communicating, networked, distributed solutions
- Secure: threatened by enormous security issues, challenging its technical and economical viability
- Complex:
  - Giga-complexity enabled by nano-technology
  - Complex through heterogeneity
  - Transducer devices
    - Sensors: Biosensors, MEMS, NEMS
    - Actuators/Interactive Screens/Displays
    - Speech input device/Handwriting input devices
  - Computing devices: application domain specific, reconfigurable; more software than hardware?
  - Communication: protocols, standards, RF
- Low power: scavenging power
- Thermal/power control: Run-time management of resources

Embedded Systems Design: not Business as usual…

- Embedded Systems require a holistic approach to design, manufacturing, and skill creation in a distributed industrial context (eco-alliances)
- Embedded systems are complex by nature:
  - GLOBAL TECHNICAL SYSTEMS: networked-sensing intelligence
  - (HW/SW)-actuation = MULTI-DISCIPLINARY
  - OPERATING on an EMBEDDING ENVIRONMENT IN AN APPLICATION DOMAIN with its own requirements and expertise

Embedded System Characteristics

- Multi-disciplinary by nature: EE + CS + DOMAIN
- Hard constraints: real-time, low cost, low energy yet complex software on dedicated distributed platforms, short time-to-market, security, usability
- Requires global system approach based on application domain expertise
- Products result from eco-alliances in domain Traverses many layers of abstraction (vertical)



Overall goal of the courses

- **Providing the perspective of a real engineer**
  - **How to move from toy-grade demo applications to real products**
  - **One out of four engineers in EU will be ES designers**
- Link between available technologies and product requirements
- Which trade-offs?
- How to deal with TTM
- How to fight power consumption
- Management of real-world constraints (roadmap, size, certification, ….)
- Link with State-of-the-art (SoA) research

ES1 goals

- Describe the main technologies and architectures to design embedded systems
- Create an industrial perspective on the problems and trade-offs, concerning
  - Analysis of the design alternatives
  - Project management
  - Available and future technologies
- Provide “hooks” for in-depth researches
- Show the cross links existing among several engineering fields

ES1 – short list of topics (1)

- **Introduction and background**
  General characteristics, constraints, trends; Energy and power optimization
- **Hardware platforms for Embedded Systems**
  Software executors for ES (DSP, Network processors, Microcontrollers, Multi-Many cores…).
  Communication architectures: standard buses, Network-on-Chip (NoC), field buses, wireless interfaces, examples.
  Hardware executors: characteristics of the application specific IC and related design flow, systems for fast prototyping, reconfigurable and updatable systems, examples of off-the-shelf (PCB-based) designs and wireless sensor networks
- **Software architectures for Embedded Systems**
  Toolchains for the realization of the embedded software, analysis/profiling of code, energy optimization
  Introduction to the real-time operating systems: general features, scheduler, configuration, dynamic management of the resources, soft real time features, energy optimization and resource management

ES1 – short list of topics (2)

- **Analysis and optimization of Hw/Sw Systems**
  Analysis of the project constraints and selection of the best implementing architecture with particular emphasis on realization time and cost.
  Problems related to the estimation of Power/Performance at the different abstraction levels
  Optimization of power and performance at the different abstraction levels
  Analysis and modeling of thermal problems with particular attention to the multi-core architectures with NoC-based communication
  Management and planning of embedded systems: models for the development flow, design for resuse and estimation of hw and sw development time and cost, project management
- **Case studies**

Hands-on

- ES1- Analysis of design alternatives, trade-offs?
- AOS – development of code for micro/multi-core based systems
- ES1 + AOS– Hands-on con STMicroelectronics/ARM boards
- Development of small-scale projects under the supervision of my teaching assistants (3 lessons)

Advanced Operating Systems (AOS)

Part 1: User-space system development

- **Theory**
  - General characteristics and roadmaps, peculiarities of applications and system software
  - Example of showstoppers: energy of SW, system-level power management, memory footprint, reactivity
  - Scheduling of CPU for single and multi-many core architectures, frameworks for the run-time management of resources
- **Practice**
  - C/C++ Application development in Linux
  - Code versioning with GIT
  - Inter-Process Communication (IPC)
  - Concurrent programming in C++11 and management of deadlocks and starvation

Advanced Operating Systems (AOS) Part 2: Hardware level programming

- **Theory**
  - RT operating systems: general characteristics, limits of traditional schedulers, configuration, user control
  - Peripherals and Interrupt management
- **Practice**
  - Examples with STM32 microcontrollers
  - CPU startup: C runtime initialization, boot loaders
  - Mastering MCU data sheets: device driver programmer's model
  - Interfacing external devices (e.g. GPIO, UART),
  - Exception vector and Interrupts

Advanced Operating Systems (AOS) Part 3: OS kernel level

- **Theory**
  - Introduction on kernel architecture
  - Loadable modules, memory management, I/O memory and ports, char devices and file operations
  - Processes and scheduling, Sleeping and Context switching
- **Practice**
  - Boot with operating system (Miosix)
  - System calls, file-system and peripheral drivers
  - Context switching
  - Time management

Energy aware design of computing systems and applications (PhD) – Group DOT

- The course covers topics in energy aware computing, from the architecture, application design methodology and system software points of view. It provides an holistic view of energy efficiency across the computing continuum, from ultra-low power embedded systems to low-power servers and green computing. Popular tools and flows will be also presented.
- **Syllabus**: (1) Introduction: motivations and topics overview; (2) Architectures: power/energy/thermal viewpoint; (3) Monitors and knobs, policies; (4) Software development; (5) Design Methodology; (6) RunTime adaptive management of resources; (7) Application case studies

Exam

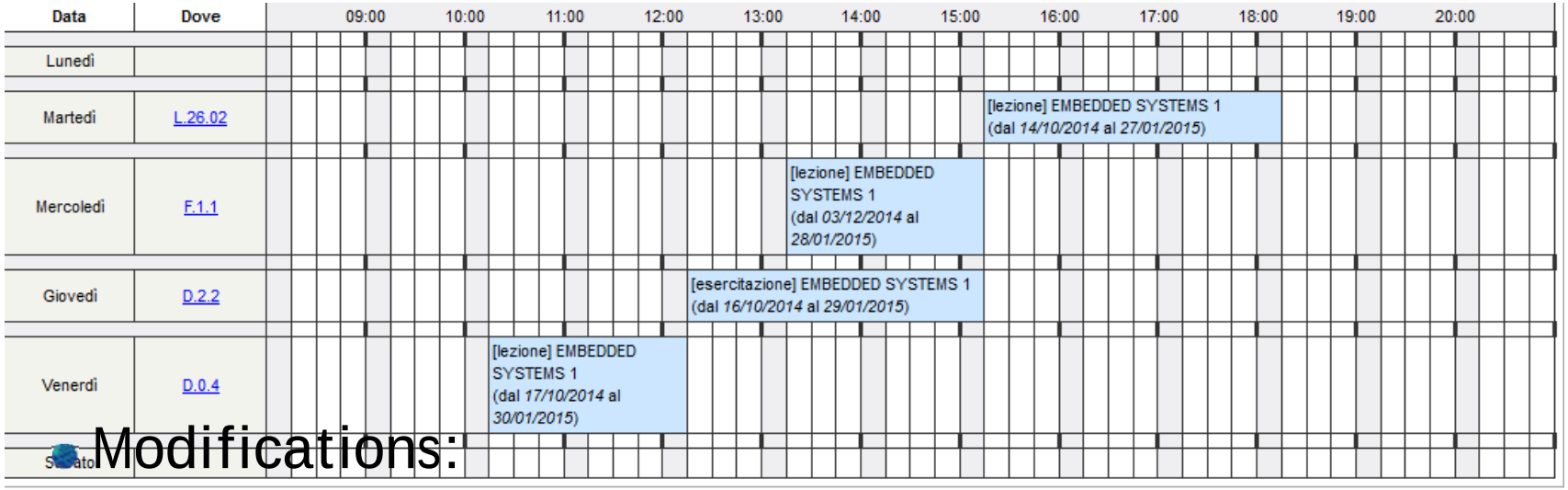
- Standard
  - Written test
  - Small literature review or project work (<30% weight)
- Note
  - Possibility to have a unique project work for both courses (ES+AOS) or in conjunction with other courses (e.g., computer architecture, Advanced User Interfaces), “Energy Aware design of computing systems and applications” or thesis
  - Availability of stages with industries
  - Be part of international research projects
  - Selection of candidates for PhD program
- Project topics
  - See web sites of Zoni, Massari, Terraneo, Libutti for a full list of topics

Possible Cooperations - research projects

- International projects (2015-2019)
  - HARPA – Low power / thermal control
  - CONTREX – Embedded Systems, IoT, mixed criticality
  - MANGO – High Performance Computing (HPC), security
  - M2DC – HPC, virtualization, data analytics, IoT, security
  - P3S – Kidtronics, wellness
- Possibility to develop master thesis and PhD in an international research environment
- Use of commecial development boards
  - STM, Cypress, Texas Instruments, Panda board, PoliNode, Zynq, Virtex, Odroid, …

When and where?

- Small adjustments to fit also other courses running in parallel



- Modifications:
  - Mercoledì – from November 18, 2015 on
  - Giovedì – start al 12:15 – end at 14:30
  - Friday – end at 11:30 (not really used extensively)
  - Martedi cercare di chiudere alle 16:30 - Rinaldi

Schedule of lessons

- Week 1 – unified ES + AOS: general introduction to the ES
- Week 2 – see calendar on the web, first split with different distributions
- Material: see my web page + pages of the teaching assistants
- Main goal
  - written test before Christmas break for AOS and part of ES1
  - January for project review and additional seminars

Schedule of lessons week 1 and week 2

	Shared	AOS	ES	*Suggested to all*						
					Note	**ES**	**AOS**	**lez**	**Es**	**Topic**
h	**room**	**time**	**day**							
	Oct-16									
3	L26.02	15:00-18:00	10/6/2015	Tuesday		3		3		Course presentation
	F11	13:00-15:00								
3	D22	12:00-15:00	10/8/2015	Thursday		3		3		Introduction to Embedded Systems
2	D04	10:00-12:00	10/9/2015	Friday			2	2		CPU & Scheduling
3	L26.02	15:00-18:00	10/13/2015	Tuesday		3	3		3	Linux Application Development Toolkit + Code versioning with GIT (part 1)
	F11	13:00-15:00								
3	D22	12:00-15:00	10/15/2015	Thursday		3		3		ES Architecture
2	D04	10:00-12:00	10/16/2015	Friday		2		2		ES Architecture & Sensors

Contact

- Prof. William FORNACIARI
  Politecnico di Milano - DEIB
  http://home.deib.polimi.it/fornacia/
  william.fornaciari@polimi.it
  02 2399 3504

HiPEAC Lab (Building 21)
High Performance Embedded and Computing Lab
Room for hosting of Ms and PhD students

