RETIS Lab investigates *embedded computing systems* with particular emphasis in

- Real-time methodologies
- Operating systems
- Design tools
People
Currently, the RETIS Lab comprises 35 people, including:

- 2 Full professors
- 2 Associate professors
- 4 Assistant professors
- 12 Research associates
- 15 PhD students

What is an embedded system?
⇒ It is a computing systems hidden in an object to control its functions, enhance its performance, manage the available resources and simplify the interaction with the user.
Embedded computing systems are becoming pervasive in our society:

- Flight control systems
- Plant control
- Automotive
- Railways switching systems
- Robotics
- Defense systems
- Patient monitoring systems

... and many others
### What’s special in Embedded Systems?

<table>
<thead>
<tr>
<th>FEATURES</th>
<th>REQUIREMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scarce resources (space, weight, time, memory, energy)</td>
<td>High efficiency in resource management</td>
</tr>
<tr>
<td>High concurrency and resource sharing (high task interference)</td>
<td>Limit interference by temporal isolation</td>
</tr>
<tr>
<td>Tight environment interaction</td>
<td>High predictability in the response time</td>
</tr>
<tr>
<td>High variability on workload and resource demand</td>
<td>Robustness (Overload handling and adaptivity)</td>
</tr>
</tbody>
</table>

### Predictability + efficiency

The RETIS group has contributed to increase predictability and efficiency of real-time systems for flight control and automotive applications.

In collaboration with **Airbus and Bosch**
Linux has been extended to handle real-time software and support the rapid growth of complexity in next-generation cell phones.

In collaboration with Ericsson

ARM11 (4 cores)

Linux 2.6.x

The FLEX Board

In collaboration with Microchip
Embedded Control Board

Main features:
- Power management
- Remote programming

Expandibility:
- Connection with other special-purpose boards

Architecture

Software

Application (C)

RTOS

ERIKA

Hardware

Microchip
dsPIC 30F601x
Educational control kits

ASCOLTA

Home assistance for heart failure through an advanced wearable monitoring systems

Objectives

- **Reduce hospital costs** through continuous monitoring at home.

- **Simplify health monitoring** through a wireless wearable monitoring system

- Automatic **detection of critical conditions**, asking for medical intervention.
ASCOLTA

Blood pressure
Breath signal
Body motion

Wireless Sensor Networks

Controller → HW → actuators
Controller → HW → sensors
Controller → HW → wireless

Typical measures
- sound
- light
- gas
- temperature
- video

© Scuola
Sample applications

- Surveillance
- Intrusion detection and tracking
- Defence systems
- Environmental monitoring
- Exploration
- Rescuing
International Collaborations

- **Carnegie Mellon University**
  - Prof. Rajkumar / Lehoczky (feasibility analysis)

- **Univ. of Illinois** at Urbana Champaign
  - Prof. Lui Sha / Marco Caccamo (RT control and scheduling)

- **Univ. of North Carolina** at Chapel Hill
  - Prof. Sanjoy Baruah (Multiprocessor scheduling)

- **Univ. of California at Berkeley**
  - Prof. Sangiovanni Vincentelli (Design methodologies and tools)

- **Florida State University**
  - Prof. Ted Baker (Multiprocessor scheduling)

- **Univ. of Indianapolis**
  - Prof. Yao Liang (Sensor Networks)
Industrial Collaborations

- Philips - QoS management for multimedia systems
- Ericsson - RT systems for telecommunications
- Airbus - kernels for avionic applications
- Bosch - kernels for automotive applications
- Microchip - RTOS for embedded control applications
- Magneti Marelli - Schedulability tools for automotive systems
- Ansaldo - Predictable kernel mechanisms
- ST Microelectronics - RTOS for multicore architectures
### Courses on Embedded Systems

<table>
<thead>
<tr>
<th>Course</th>
<th>Teacher</th>
<th>NOTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real-Time Systems</td>
<td>Giorgio Buttazzo</td>
<td>Methodologies for increasing software predictability. Task scheduling and feasibility analysis; protocols for accessing shared resources; interrupt handling.</td>
</tr>
<tr>
<td>Introduction to Neural Networks</td>
<td>Giorgio Buttazzo</td>
<td>Basic networks models and learning algorithms for pattern recognition, classification, prediction, and control.</td>
</tr>
<tr>
<td>Object Oriented Software Design in C/C++</td>
<td>Giuseppe Lipari</td>
<td>Software design methodologies in C++; Code design patterns; Templates; Concurrency in C++; the new C++11 standard.</td>
</tr>
<tr>
<td>Laboratory of Real-Time Systems</td>
<td>Mauro Marinoni, Gianluca Franchino</td>
<td>Programming microcontrollers, I/O devices, sensor acquisition and processing, servomotor control, wireless communication protocol. Use of a real-time kernel to develop real-time control applications.</td>
</tr>
</tbody>
</table>
### Courses on Embedded Systems

<table>
<thead>
<tr>
<th>Course</th>
<th>Teacher</th>
<th>NOTES</th>
</tr>
</thead>
</table>
| Combinatorial Optimization                  | Michela Milano, Andrea Roli, Enrico Malaguti, Michele Lombardi | Integer Linear Programming  
Simulated Annealing  
Genetic Programming  
Hybrid methods |
| Android Framework                           | Alberto Panizzo                  | Mobile device features, Application environment, User interface, Data management  
Complex services and security |
Forward and backward error recovery. Static and dynamic redundancy. Transactions and checkpoints |
| Modeling of concurrent programming          | Sverre Hendseth                  | Formal methods for expressing concurrent programs and verifying their correctness |
| How to do research                          | Giorgio Buttazzo, Gerhard Fohler  | How to write scientific papers  
Common English mistakes  
How to make slides  
How to make presentations  
How to review papers |

[http://retis.sssup.it/phd-engineering/](http://retis.sssup.it/phd-engineering/)