

UNIVERSITÀ DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Sistemi Embedded

2526-3-E3101Q124

Aims

Understanding of the major issues in real-time embedded systems. Ability to develop simple real-time embedded applications on micro-controllers and SBC (Single Board Computer) in languages near to the hardware, e.g., C, with and without OS.

Contents

Embedded systems: features and requirements. The structure of embedded systems: micro-controllers, DSP, FPGA, memories and their organization, communication systems. Peripherals, sensors and actuators. Real-time scheduling theory. Software architectures and libraries for real-time, fail-safe and safety-critical programming. Real-time operating systems. Laboratory activity with micro-controller programming in C. Laboratory activity with PC-like machines in C under Linux with scheduling classes RT and deadline.

Detailed program

- 1. Embedded systems: features and requirements
 - 1. General features of embedded systems; application domains; market value and diffusion.
 - 2. Basic requirements: timing, reliability, efficiency.
 - 3. Design choices vs requirements.
- 2. The structure of embedded systems

- 1. Small recap of digital and analog electronics.
- 2. Scale levels: IC. PCB. network.
- 3. Processing units: CPU, microcontrollers, DSP, GPU, ASIC. Programmable logic: FPGA.
- 4. Memories: SRAM and DRAM, non-volatile memories; interactions between processor and memory: Von Neumann and Harvard architectures, memory hierarchies.
- 5. Communication systems: GPIO, Pulse Width Modulation, RS-232, USB, I2C, SPI, CAN bus, JTAG.
- 6. Systems with a high degree of integration: SoC e NoC.
- 7. Examples of micro-controllers.

3. Peripherals, sensors and actuators

- 1. Timers.
- 2. DMA.
- 3. Fundamentals of sampling theory: Nyquist theorem, aliasing, quantization noise; Comparators and A/D and D/A converters.
- 4. Models of sensors and actuators: Affine models, saturation, harmonic distortion, dynamic range.
- 5. Sensors: Accelerometers and gyroscopes.
- 6. Actuators: Linear solenoids and DC motors.

4. Real-time scheduling theory

- 1. Basic definitions: periodic, aperiodic e sporadic tasks, utilization, valid and feasible schedules, optimality.
- 2. Cyclic schedule.
- 3. Static priority scheduling: rate-monotonic and deadline-monotonic schedule, schedulability analysis.
- 4. Dynamic priority scheduling: earliest-deadline-first and least-slack-time-first schedule.
- 5. Scheduling for aperiodic and sporadic jobs.
- 6. Blocking time analysis.
- 7. Critical sections, scheduling anomalies (priority inversion and deadlock), priority inheritance and priority ceiling protocols.

5. Software architectures and libraries

- 1. Main architectures: Round-robin, round-robin with interrupts, function-queue-scheduling.
- 2. (hints) POSIX.4, Ada Real-Time and Ravenscar, Real-Time and High-Integrity Java.
- 3. (hints) real-time operating systems.

6. Laboratory activity on micro-controllers programming.

- 1. Software development toolchain and IDE.
- 2. Assembly programming and development of some basic programs.
- 3. C programming.
- 4. Team project development.

Prerequisites

- 1. Basic knowledge of computer architecture.
- 2. Basic skills in C programming.

3. Basic knowledge of operating systems and concurrent programming.

Teaching form

- · erogative classes;
- laboratory interactive practices, with development of projects in small groups; such activities will base on HW STM from the Nucleo family as well as compatible shields, and on the small robot Coderbot (Raspberry Pi).

Textbook and teaching resource

relevant

• G. C. Buttazzo. Hard Real-Time Computing Systems, Predictable Scheduling Algorithms and Applications, 3rd Edition. Springer, 2011

less relevant

- J. W. S. Liu. Real-Time Systems. Prentice-Hall, 2000.
- E. A. Lee, S.A. Seshia. Introduction to Embedded Systems: A Cyber-Physical Approach. Second Edition, MIT Press, 2017.
- D. E. Simon. An Embedded Software Primer. Addison Wesley, 1999.
- C. Brandolese, W. Fornaciari. Sistemi Embedded: Sviluppo Hardware e Software per Sistemi Dedicati. Pearson, 2007.

Semester

Second semester

Assessment method

- Written examination with exercises and/or open questions on topics not covered by the laboratory activities;
- · written report on the laboratory activities;
- possible oral examination about the laboratory report.

Office hours

Send email for an appointment.

Sustainable Development Goals

INDUSTRY, INNOVATION AND INFRASTRUCTURE