270061 - DSBM - Design of Microcomputer-Based Systems

Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 707 - ESAII - Department of Automatic Control
Academic year: 2018
Degree: BACHELOR’S DEGREE IN INFORMATICS ENGINEERING (Syllabus 2010). (Teaching unit Optional)
ECTS credits: 6
Teaching languages: Catalan

Teaching staff

Coordinator: - Enric X. Martin Rull (enric.xavier.martin@upc.edu)
Others: - Daniel Garcia Solà (daniel.garcia.sola@upc.edu)

Prior skills


Requirements

- Prerequisite CI

Degree competences to which the subject contributes

Specific:

CEC1.1. To design a system based on microprocessor/microcontroller.

CEC2.2. To program taking into account the hardware architecture, using assembly language as well as high-level programming languages.

CEC2.3. To develop and analyse software for systems based on microprocessors and its interfaces with users and other devices.

CEC3.1. To analyse, evaluate and select the most adequate hardware and software platform to support embedded and real-time applications.

CEC3.2. To develop specific processors and embedded systems; to develop and optimize the software of these systems.

CT7.2. To evaluate hardware/software systems in function of a determined criteria of quality.

General:

G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.
270061 - DSBM - Design of Microcomputer-Based Systems

Teaching methodology

There will be a complementation between classes of theory and problems, lectures are reinforced with examples showing the possible alternatives and solutions to common problems. Some topics will be proposed for self-assessment exercises that students can be aware of your progress, and may ask teacher support in case it detects any deficiency. The practical sessions will take place in situ in the laboratory teaching department in the FIB. There will be two practices that require large cumulative work of students in the preparation of a project.

Learning objectives of the subject

1. Given an application, determine for each task: its duration, the maximum waiting time, time critical care, and plan the right strategy to meet these requirements.
2. Identify the critical regions of a program and schedule properly due to bugs in open source data compartment.
3. Determine the parts of the code needed to be programmed in assembler, and can be programmed in high level language.
4. Sizing the timing of the watchdog correctly, and placing the 'kicks' the watchdog properly within the code.
5. To determine the most appropriate software architecture for a particular application, based on the number of processes, the computational requirements of these and immediate response.
6. Determine the best serial communication interface for communication between two integrated circuits (processors, processor-interface).
7. Build routines minimum hardware abstraction interface for all communications.
8. Place adequate protections to a microprocessor system for its connectivity to the outside.
9. Protecting against noise reception and a microprocessor system.
10. Sizing correctly sampling frequency of one or more signals according to their nature and the computational load of micro.
11. Setting a reference voltage AD converter with appropriate values from the signal dynamic range and resolution required.
12. Interpret the representation of a periodic signal in a Bode diagram and its decomposition in Fourier series.
14. To extract the correct values of the electrical characteristics published by the manufacturer in the technical reference manuals.
15. Select the appropriate device (transistor, relay, triac, optocoupler) for connecting external loads to the processor pins. Sizing the components necessary to successfully make the connection.
17. Defender design hardware or software-based microprocessors in a presentation in front peers (target practice).
18. Draw and correctly interpret time charts.
19. Learning to use laboratory components: Oscilloscopes, logic analyzer, etc. .. (Target practice)
20. Schedule a system with different sensors and interfaces to fulfill requirements of execution. (Target practice)
21. Design and implement a microcontroller solution for a given problem. (Target practice)

Study load

<table>
<thead>
<tr>
<th>Study load</th>
<th>Total learning time: 150h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes:</td>
<td>18h</td>
</tr>
<tr>
<td>Practical classes:</td>
<td>12h</td>
</tr>
<tr>
<td>Laboratory classes:</td>
<td>30h</td>
</tr>
<tr>
<td>Guided activities:</td>
<td>6h</td>
</tr>
<tr>
<td>Self study:</td>
<td>84h</td>
</tr>
</tbody>
</table>
## Content

### Introduction.

**Degree competences to which the content contributes:**

**Description:**
Characteristics of microprocessor based systems. Type, common solutions to market.

### Laboratory tools.

**Degree competences to which the content contributes:**

**Description:**
Oscilloscopes, logic analyzer, emulator, simulator, ICD, debuggers.

### Hardware Issues

**Degree competences to which the content contributes:**

**Description:**
Hardware for microcomputer-based systems. Circuits, components, time diagrams, schematic design.

### I / O and analog impulse.

**Degree competences to which the content contributes:**

**Description:**
How to connect sensors to a microprocessor interface. Bridges, resolution, precision. Check the impulse. Limits, often.

### Software Aspects of microprocessor based systems.

**Degree competences to which the content contributes:**

**Description:**
Optimization and code inspection. Startup, linkatge, location code, bootloaders. Concurrence, context switching, task management, interrupts, RTOS.

### Interfaces between systems.

**Degree competences to which the content contributes:**

**Description:**
Description Interface I2C, SPI, CAN, Ethernet, Bluetooth, RF.
### Supervision and safety.

**Degree competences to which the content contributes:**

**Description:**
Watchdog, NMI, redundancy, monitoring.

### Protections, compatibility.

**Degree competences to which the content contributes:**

**Description:**
Robust to noise, the system microprocessor protection, electromagnetic compatibility.
## Planning of activities

### Introduction

**Hours:** 6h  
- Theory classes: 2h  
- Practical classes: 0h  
- Laboratory classes: 0h  
- Guided activities: 0h  
- Self study: 4h

**Description:**  
View the types of solutions available. Evaluate the most appropriate for a problem. To understand the representation and the associated figures.  

**Specific objectives:**  
17, 19

### Practice 1

**Hours:** 6h  
- Theory classes: 0h  
- Practical classes: 0h  
- Laboratory classes: 2h  
- Guided activities: 0h  
- Self study: 4h

**Description:**  
Learn to use the tools build in ASM and C Language Understanding the architecture of the microcomputer.  

**Specific objectives:**  
20

### Hardware for micro-computer boards.

**Hours:** 10h  
- Theory classes: 2h  
- Practical classes: 2h  
- Laboratory classes: 0h  
- Guided activities: 0h  
- Self study: 6h

**Description:**  
Circuits, components, basic diagrams, time diagrams, schematic design tools ...

**Specific objectives:**  
12, 13, 14, 16, 17

### Practice 2

**Hours:** 4h  
- Theory classes: 0h  
- Practical classes: 0h  
- Laboratory classes: 2h  
- Guided activities: 0h  
- Self study: 2h
### Practice 3

**Description:**
Laboratory tools: analyzer, Oscilloscopes, emulator and simulator. ICD, debug.

**Specific objectives:**
- 20

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 0h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 2h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 0h</td>
</tr>
</tbody>
</table>

### CT1. Control Theory 1

**Description:**
There will be an exercise in basic circuits in assembling a microcomputer

**Specific objectives:**
- 10, 11, 12, 13, 14, 16, 17, 19

<table>
<thead>
<tr>
<th>Hours</th>
<th>Guided activities: 1h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Self study: 4h</td>
</tr>
</tbody>
</table>

### I / O and analog impulse.

**Description:**
Concepts / O, connecting sensors, bridges, resolution, accuracy. Concepts related to frequency, sampling frequency dividers.

**Specific objectives:**
- 10, 11, 12

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 3h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 4h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 10h</td>
</tr>
</tbody>
</table>
# Practice 4

**Hours:** 10h  
Theory classes: 0h  
Practical classes: 0h  
Laboratory classes: 6h  
Guided activities: 0h  
Self study: 4h  

**Description:**  
Practical I / O We developed an exercise to work on a system microcontroller interfaces.  

**Specific objectives:**  
21

---

# PL1. Lab Test 1

**Hours:** 6h  
Guided activities: 2h  
Self study: 4h  

**Description:**  
There will be an exercise spot on the hardware aspects of the microcomputer system.  

**Specific objectives:**  
20, 21

---

## Noise protection, support MS.

**Hours:** 6h  
Theory classes: 1h  
Practical classes: 1h  
Laboratory classes: 0h  
Guided activities: 0h  
Self study: 4h  

**Description:**  
Study of protection against electrical noise and EM radiation. Ensure compatibility of systems designed with the other existing systems.  

**Specific objectives:**  
8, 9

---

# Practice 5

**Hours:** 16h  
Theory classes: 0h  
Practical classes: 0h  
Laboratory classes: 10h  
Guided activities: 0h  
Self study: 6h  

**Description:**  
Practice miniprojecte the system expandable USB
## Specific objectives:
17, 18, 19, 22

### CT2. Control Theory 2

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 3h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 1h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 4h</td>
</tr>
</tbody>
</table>

**Description:**
We will exercise all the interconnection of a microcontroller on a board and its I/O

**Specific objectives:**
6, 7, 10, 17, 19

### Interfaces between systems

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 3h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 1h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 4h</td>
</tr>
</tbody>
</table>

**Description:**
Explained the basic concepts of communication interfaces I2C, SPI, CAN, Ethernet, Bluetooth, RF ... to be worked in practice.

**Specific objectives:**
6, 7

### Monitoring Systems

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 1h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 1h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 4h</td>
</tr>
</tbody>
</table>

**Description:**
Study of the functioning of monitoring systems microcontrollers: Watchdog, NMI, redundancy ...

**Specific objectives:**
4

### Software for Microcontrollers

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 3h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 3h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 14h</td>
</tr>
</tbody>
</table>

**Description:**
...
### Description:
Optimization and code inspection, startup, linkatge, location code, bootloaders. Concurrency, context switches, interrupt management, tasks and RTOS.

### Specific objectives:
1, 2, 3, 5

### CT3. Control Theory 3

**Hours:** 7h  
Guided activities: 1h  
Self study: 6h

### Specific objectives:
1, 2, 3, 4, 5

### Preparing presentations and practical work

**Hours:** 3h  
Theory classes: 0h  
Practical classes: 0h  
Laboratory classes: 0h  
Guided activities: 3h  
Self study: 0h

**Description:**
To help and guide students to prepare their practical work displayed at the end of the course.

**Specific objectives:**
18

### PTL. Presentation of Laboratory work

**Hours:** 7h  
Guided activities: 3h  
Self study: 4h

**Description:**
It will be presented to Professor practical laboratory work and will be check its operation.

**Specific objectives:**
8, 9, 18, 22
The grade for this course will be obtained from the weighted average of the marks of theory (40%), laboratory practices (40%) and final work (20%).

\[
NF = 0.1 \times CT1 + 0.1 \times CT2 + 0.1 \times CT3 + 0.1 \times CT4 + 0.40 \times PL + 0.2 \times TF \quad (CT = \text{control theory}, \ PL = \text{Laboratory}, \ TF = \text{Final Work})
\]

The laboratory mark \((PL)\) will be obtained while doing practical work in the lab. There will be at least two partial deliveries of the practices.

In addition students must deliver a design of an embedded system based on a real problem. This work will be presented at the end of the course and will produce the mark \((TF)\). The docs for this design will be written in English. The quality of design, the selection of components and clarity of the presentation will be evaluated.

With the report in English and the presentation of the final work a grade for the Cross-Competence (G3.2) will be obtained.

### Bibliography

**Basic:**


**Complementary:**