**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**
**Faculty of Electrical and Computer Engineering**
**COURSE SYLLABUS**
*Microprocessors*

## 1. CODE AND NUMBER OF CREDITS

| | | |
|---|---|---|
| **CODE** | FIEC00760 | |
| **NUMBER OF CREDITS: 6** | **Theoretical: 4** | **Practical: 2** |

## 2. COURSE DESCRIPTION

This course studies the architecture and internal resources of the microprocessors and its low-level programming as a basic component of any computer. Addressing modes and memory segmentation. The instruction set and programming in assembler language. Software tools: DEBUG, CODEVIEW AND MASM. Utilities of ROMBIOS and MSDOS for management of keyboard and screen in text mode.. Interrupt system. Analysis and design of assembler language programs.

It also studies the basic 8051/8052 microcontroller and its derivatives that have improved architectures. Set of instructions and assembly-language programming. Software and hardware design of digital systems based on this kind of microprocessors using development tools given by manufacturers.

## 3. PRE-REQUISITES AND CO-REQUISITES

| | |
|---|---|
| **PRE-REQUISITES** | FIEC00299 Digital Systems I |
| **CO-REQUISITES** | None |

## 4. CORE TEXT AND OTHER REQUIRED REFERENCES FOR THE TEACHING OF THE COURSE

| | |
|---|---|
| **CORE TEXT** | **FIRST PARTIAL**<br>1. THE INTEL MICROPROCESSORS, Architecture, Programming and Interfacing. Barry B. Brey. Prentice Hall.<br>2. IBM PC ASSEMBLY LANGUAGE AND PROGRAMMING. Peter Abel. Prentice Hall.<br>**SECOND PARTIAL**<br>1. THE 8051/8052 MICROCONTROLLER. Architecture, Assembly Language, and Hardware Interfacing. Craig Steiner. Universal Publishers.<br>2. THE 8051 MICROCONTROLLER AND EMBEDDED SYSTEMS. Using Assembly and C. By Muhammad Ali Mazidi, Janice Gillispie Mazidi and Rolin D. McKinlay. |
| **REFERENCES** | 1. Lecture notes in PowerPoint.<br>2. Laboratory equipment.<br>3. Windows Virtual PC - Windows XP Mode: ROMBIOS and MSDOS functions.<br>4. Software and hardware tools for the development of digital systems based on microcontrollers.<br>5. http://www.8052.com/<br>6. http://www.mikroe.com/. |

## 5. COURSE LEARNING OUTCOMES

At the end of the course, the student will be able to:

1. Identify in detail the internal architecture, the instruction set and the operation of the Intel microprocessors.
2. Know the addressing techniques and the design of programs in Assembler Language.
3. Use the ROM-BIOS and MS-DOS functions to input data from a keyboard and display it to a text screen on a PC.
4. Identify the essential aspects of the architecture and operation of Intel microcontrollers and its derivative chips: program memory, data memory, special functions registers, timers, interrupts and programming in assembler language.
5. Develop software and hardware projects based on Intel microcontrollers

## 6. COURSE PROGRAM

**FIRST PARTIAL**

I. **Introduction: 4 hours**

- History and evolution of the microprocessors of Intel.
- Von Neumann architecture: generalities, data bus, address bus, control bus and memory.
- The 8085 as an example of a typical microprocessor of 8 bits. Internal architecture: ALU, control unit, the work registers.
- 8085 Instruction format.
- The instruction cycle. Faces of the instruction cycle.
- Technologies to improve the instruction cycle.

II. **The 8088/8086 as an example of a typical microprocessor of 16 bits: 2 hours**

- General Characteristics of 8086/8088.
- Internal Organization and memory segmentation: The interface bus unit and the execution unit.
- Pointers, Data registers and Status Register.

III. **The 80386 as an example of a 32 bits Microprocessor:** 2 hours

- Internal Architecture of the 80386DX. Major changes.
- Real mode Software Model of the 80386DX. Memory segmentation. Work registers and pointers. Flag register.
- Real Mode Memory Address space and data organization. Real Mode Input / Output address space. Real mode physical address.
- The stack. Stack instructions.
- The PC and its DEBUG program.

IV. **Data Addressing Modes:** 2 hours

- Logical address and Physical address.
- Register addressing mode.
- Immediate addressing mode.
- Direct addressing mode.
- Register indirect addressing mode.
- Based relative addressing mode.
- Based index addressing mode.
- Segment overrides.

V. **Program Memory Addressing Modes:** 2 hours

- Intrasegment direct.
- Intrasegment indirect.
- Intersegment direct.
- Intersegment indirect.

VI. **The Instruction Set and its classification:** 2 hours

- Data transfer instructions.
- Arithmetic and logic instructions.
- Jump instructions.
- String instructions.
- Miscellanea.

VII. **Assembler Language:** 4 hours

- Basic concepts.
- Assembler directives.
- Types of variables.
- Subroutines and Macros.
- Strings and data Arrays.
- Assembler memory models.
- Assembling, Linking and Executing a Program.

VIII. **Interrupts:** 2 hours

- Software interrupts.
- Hardware interrupts.
- The PC interrupt vector table.
- The MSDOS-BIOS Interface.

IX. **BIOS Functions:** 4 hours

- Use of INT 10H instruction to access BIOS functions.
- Data input from keyboard.
- Data output to video screen.
- Keyboard Input with INT 16H.
- Assembler Language Programs Examples.

X. **MSDOS Functions:** 4 hours

- Use of INT 21H instruction to access MSDOS functions.
- Data input from keyboard
- Data output to video screen.
- Assembler Language Programs Examples.

**SECOND PARTIAL**

XI. **Intel microcontroller and its derivatives:** 4 hours

- Internal Architecture: CPU, program memory and data memory.
- Basic Registers.
- Special function registers (SFRs).
- Assembler Language Fundamentals: basic directives.
- Software Tools: development environment and simulator.

XII. **Instruction Set:** 4 hours

- Data transfer instructions.
- Arithmetic and logic instructions.
- Multiply and Divide Instructions.
- Jump instructions.
- Call and return instructions.
- Miscellanea.

XIII. **Addressing Modes:** 4 hours

- Immediate addressing.
- Direct addressing.
- Indirect addressing, external direct addressing, external indirect addressing.
- Delays.
- Examples.

XIV. **I/O port programming and special resources**: 8 hours

- Ports P0, P1, P2 y P3. Port configurations. Dual role of port 0 and port 2.
- I/O port and bit addressability.
- Reading input pins vs. port latch.
- Reading Latch for output port.
- Look-up Tables.
- Seven segment displays: multiplexation of 7 segment displays.
- LCD management.
- Matrix 4x4 keypad management.
- Examples with keyboard and LCD. (6 h).

XV. **Timers:** 4 hours

- Timer 0 and Timer 1.
- TMOD register and TCON register and operating modes.
- Timer as time delay.
- Timer as event counter.
- Examples.

XVI. **The Interrupts system:** 2 hours

- Sources of Interrupts.
- Interrupt Enable Register (IE register).
- Serial Port Interrupt.
- Hardware interrupt INT0 (pin P3.2),
- Hardware interrupt INT1 (pin P3.3).
- Programming Timer Interrupts.
- Programming External Hardware Interrupts.

XVII. **Serial Communication:** 4 hours

- UART. Asynchronous serial communication.
- RS 232 standard. MAX 232.
- The serial port programming. SBUF and SCON registers.
- Baud rate (bps).
- Examples.

XVIII. **Interfacing with DAC and ADC**: 2 hours

- Interface with external world: DAC y ADC.
- Example. The ADC 0804.
- Digital / analog interfacing example.

XIX. **Input / Output Interface. Programming of PPI 8255:** 2 hours

- The 8255 Programmable Peripheral Interface (PPI): internal architecture.
- Modes of operation and programming.
- Interface with external devices via PPI 8255.
- Example.

XX. **Stepper Motors Control: 2 hours**

- Unipolar Stepper Motor Control. Programming Example.
- Bipolar Stepper Motor Control. Programming Example.

XXI. **LABORATORY PRACTICE:** 10 sessions, 20 hours

## 7. WORKLOAD: THEORY/PRACTICE

- **Theory:** 2 sessions per week – 2 hours per session.
- **Laboratory:** 1 session per week – 2 hours per session.

## 8. CONTRIBUTION OF THE COURSE TO THE EDUCATION OF THE STUDENT

This course contributes to the knowledge of engineering techniques in the design of digital systems, provides experience in the design through laboratory practice and hardware and software projects. The engineer is trained to design any digital system based on a PC or microcontrollers. The design environment and development of a digital system is based on the use of software tools combined with optimal use of hardware.

| BASIC TRAINING | PROFESSIONAL TRAINING | SOCIAL SKILLS DEVELOPMENT |
|---|---|---|
| | **X** | |

## 9. THE RELATIONSHIP BETWEEN THE LEARNING OUTCOMES OF THE COURSE AND THE LEARNING OUTCOMES OF THE DEGREE PROGRAM

| | LEARNING OUTCOMES OF THE DEGREE PROGRAM* | CONTRIBUTION (High, Medium, Low) | LEARNING OUTCOMES OF THE COURSE** | THE STUDENT MUST: |
|---|---|---|---|---|
| a) | An ability to apply knowledge of mathematics, science and engineering. | Medium | 1,2,3 | To solve the problems raised during the development of the course based on the operating specifications submitted |
| b) | An ability to design and conduct experiments, and to analyze and interpret data | Medium | 4 | Use in your designs programming languages of low level. |
| c) | An ability to design a system, component or process to satisfy realistic constraints. | High | 5 | Design their systems with specifications based on real processes. |
| d) | An ability to function on multidisciplinary teams. | ----- | | |
| e) | An ability to identify, formulate and solve engineering problems. | ----- | | Make use of the software and hardware tools. |
| f) | An understanding of ethical and professional responsibility. | ----- | | |
| g) | An ability to communicate effectively. | ----- | | |
| h) | A broad education necessary to understand the impact of engineering solutions in a social, environmental, economic and global context. | ----- | | |
| i) | A recognition of the need for, and an ability to engage in life-long learning. | ----- | | |
| j) | A knowledge of contemporary issues. | ----- | | |
| k) | An ability to use the techniques, skills, and modern tools necessary for engineering practice. | high | 5 | Acquire skills and skills in the development of microcontroller-based systems. |
| l) | Capacity to lead, manage and undertake projects. | ----- | | |

## 10. EVALUATION IN THE COURSE

| Evaluation activities | |
|---|---|
| Exams | X |
| Tests | X |
| Homework/tasks | |
| Projects | X |
| Laboratory/Experiments | X |
| Class participation | |
| Visits | |
| Other | |

## 11. PERSON RESPONSIBLE FOR THE CREATION OF THE SYLLABUS AND THE DATE OF ITS CREATION

| | |
|---|---|
| **Created by** | MSc. Hugo Villavicencio V. |
| **Date** | May / 2013 |

## 12. APPROVAL

| ACADEMIC SECRETARY OF THE ACADEMIC DEPARTMENT | DIRECTOR OF TECHNICAL ACADEMIC SECRETARY |
|---|---|
| NAME:<br>Mrs.Leonor Caicedo G. | NAME:<br>Eng.Marcos Mendoza |
| SIGNATURE: | SIGNATURE:<br>ESCUELA SUPERIOR POLITECNICA DEL LITORAL |
| Date of approval by the Directive Council:<br>2013-537 2013-10-7 | Ing. Marcos Mendoza V.<br>DIRECTOR DE LA SECRETARIA TECNICA ACADEMICA |

## 13. VALIDITY OF THE SYLLABUS

| | |
|---|---|
| RESOLUTION OF THE POLYTECHNIC BOARD: | 13-12-343 |
| DATE: | 2013-12-12 |