JOINT INSTITUTE
交大密西根学院

Standard Undergraduate Course Profile

<table>
<tr><td>COURSE NUMBER: Ve373</td><td>COURSE TITLE: Microprocessor Based System Design</td></tr>
<tr><td>CREDIT: 4</td><td>PREREQUISITES: Ve270 and Ve370 or instructor permission</td></tr>
<tr><td>TEXTBOOKS/REQUIRED MATERIAL:
None</td><td>PREPARED BY: Gang Zheng
LAST UPDATED: October 25, 2020
DATE OF DISCIPLINE GROUP APPROVAL:
DATE OF UC APPROVAL:</td></tr>
<tr><td>CATALOG DESCRIPTION:
This course is designed to cover different aspects of microprocessor-based system, and consists of both lecture and laboratory sessions. Topics include microprocessor memory map, software development, simulation, debugging and testing; hardware and software interfacing; principles of interrupts; peripheral configuration and control for Timers, UART, ADC, PWM, I2C, and other external devices. Experiments with specially designed laboratory facilities will help students to develop skills for embedded software development using assembly and C programming languages.</td><td>COURSE TOPICS:
1. Introduction to embedded systems
2. PIC MCU architecture, registers, memory organization,
3. PIC32 instruction set, MIPS assembly programming
4. Embedded C Programming
5. I/O ports and operations, intro to MPLAB IDE
6. Timers and applications
7. Interrupts
8. Interaction with external devices, LCD controller
9. Power-saving operations
10. Input capture, output compare, PWM
11. Analog to digital converter
12. Serial communications, UART
13. SPI and I2C
14. Introduction to DMA and CAN</td></tr>
<tr><td colspan="2">COURSE STRUCTURE/SCHEDULE: Lecture: two 90-minute lectures each week; Laboratory: 1 per week, 2.5 hrs</td></tr>
<tr><td>COURSE OUTCOMES [Student Outcomes in brackets]</td><td>After completing Ve373, students should:
1. Be able to implement a complete microprocessor-based system using the components provided in the lab, including input and output peripheral devices, and simple but complete control software. [1,2,6]
2. Given a digital device and its reference manual, be able to write control software in assembly or C programming languages to initialize and perform operations on the device. [7]
3. Be able to interface external controllers, such as an LCD controller through, I/O ports
4. Be able to handle timed signals using internal timers and timing management system
5. Be able to handle multiple peripherals using interrupts
6. Understand the importance of, and be able to apply power saving operations in microprocessor-based system designs
7. Understand basic communication techniques used in modern embedded systems, and be able to establish communications between embedded devices using UART, SPI, or I2C</td></tr>
<tr><td>COURSE OBJECTIVES [Course Outcomes in brackets]</td><td>1. To teach students PIC microprocessor architecture and various peripherals [1,2,3,4,5,6,7]
2. To teach students how the hardware and software components of a microprocessor-based system work together to implement system level features. [1,2,3,4,5,6,7]
3. To teach students the operating principles of, and provide hands-on experience with, common microprocessor peripherals such as UARTs, timers, and ADC. [3,4,5,7]
4. To provide practical experience in applied digital logic design and embedded C programming.[1,2]
5. To expose students to the tools and techniques used by practicing engineers to design, implement, and debug microprocessor-based systems. [1,2,3,4,5,6,7]</td></tr>
<tr><td>ASSESSMENT TOOLS [Course Outcomes in brackets]</td><td>Homework [2,3,4,5,6,7]
Midterm Exam [2,4,5]
Final Exam [3,4,5,6,7]
Lab experiment [2,3,4,5,6,7]
Course project [1]</td></tr>
</table>

Rev 1: June 2015; Rev 2: Oct. 2020