Microprocessor: Course Materials

This page contains **Syst** Microprocessor course materials and relevant information. Toggle the tabs below to view all the content of this course area.

To view specific Microprocessor course materials, select the "Materials" tab

- **Views**

- **Materials**
- **SLO**
- **Topics**
- **Evaluation**
- **Course**
- **Print**
- **Resources**

Traditional View

Every AAS degree program in electronics technology has both Digital and Microcomputer fundamentals courses. The Digital course teaches the binary number system and coding, logic gates and flip flops, and basic combinational and sequential logic circuits. Some courses also include an introduction to A/D and D/A conversion. Others provide an introduction to microcontrollers. Usually the Digital course is the prerequisite for a more comprehensive microcontroller's course.

Today, virtually all digital circuits are made with a microcontroller or a programmable logic device (PLD) like an FPGA. Most colleges have gotten around to adding PLD coverage to some extent. Some also introduce microcontrollers in a digital course.

The importance of a microcomputer course that emphasizes embedded controllers is an essential part of any electronics technology curriculum. The goal of this course is to introduce a popular embedded controller and show how it is used to implement almost any digital function. That involves programming the microcomputer and using the interfaces that connect it to those devices it will monitor and control.

Most technician work is at a higher systems level working with computers, boards, equipment, modules and in some cases ICs. The work is more related to testing and troubleshooting than to design. Techs trace signals and make measurements to insure compliance to some specification or standard. They use oscilloscopes, pulse generators, and logic analyzers to perform these jobs. Much of the work involves buses and interfaces between boards, modules and equipment. Typically most courses do not include coverage of these critical connections.

New Systems View

Virtually every electronic product has at its heart a microcontroller. In fact, for larger pieces of equipment and systems, two or more microcontrollers are very common. Technician work involves testing and troubleshooting and that will typically involve the microcontroller. To work at the systems level in electronics means having a grasp of how the microcontroller connects to the rest of the equipment and how it is programmed to perform the desired system functions. The Microcontroller course describe here serves to teach those fundamentals and to show how the microcontroller is part of a larger system.

Microprocessor: Course Materials

This page contains e**Syst** Microprocessor course materials and relevant information. Toggle the tabs below to view all the content of this course area.

To view specific Microprocessor course materials, select the "Materials" tab

- **Views**
- **Materials**
- **SLO**
- **Topics**
- **Evaluation**
- **Course**
- **Print**
- **Resources**

Microprocessor Instructional Materials List
Last Update: June 27, 2013

Click on the desired materials to download. By downloading, you agree to the terms of use of eSyst.

1. Instructor Presentation: Serial Digital Interface
   Basic electronics systems operations all contain the generation, transfer, and reception of electronic data. One of the most common but not usually covered interfaces is Serial Input and Output (I/O). This presentation covers the standard ways of getting data in and out of digital circuits, equipment, and computer systems.

2. Instructor Guide: Microprocessor Beer Module
   In an automation and process control system, a microcontroller is used to read system interface inputs and provide a desired output. The objective of this lab is to use C language to program Freescale microcontroller to do Pasteurization process control.

3. Instructor and Student Resource: Microprocessor Block Diagram Animation
   Ever wonder how a computer works? Using the top-down approach, this block diagram takes you on a journey through the various stages of a microprocessor's architecture. At the system level, the details of each component are concealed, all emphasis being placed on the task that each device performs. For starters, every microprocessor has an arithmetic logic unit (ALU). As its name suggests, this component allows a computer to perform both arithmetic and logic operations. Therefore, it is important to take time to understand how this circuit works by following the animation.

4. Student Guide: Operation of Full Adder and Subtractor Using Xilinx ISE 9.2i Project Navigator and Spartan 3E FPGA Development Board with Schematic and VHDL
   The goal of this lab is to lean the use of Spartan 3E FPGA development board from Xilinx; how to program in VHDL and create the hardware connections between the development board and your PC.

5. Instructor Guide: Security Monitor and Control System 68HC12 Development Board
   The Security Monitor and Control lab is designed to provide a quick start for student to learn Assembly language using Feescale Code Warrior development system. The onboard switches are used to simulate arm switch and security sensors; built-in LEDs are used to indicate system outputs. A template is provided that is ready to upload to

the Freescale controller board. System functional requirements are provided. Students are tasked to verify that system is operated as described. Change of I/O assignment is made for students to practice changing I/O statements using Assembly language.

* Instructors should contactmike.lesiecki@domail.maricopa.edu , eSyst Director, for an answer key.

Microprocessor: Course Materials

This page contains **Syst** Microprocessor course materials and relevant information. Toggle the tabs below to view all the content of this course area.

To view specific Microprocessor course materials, select the "Materials" tab

- **Views**
- **Materials**
- **SLO**
- **Topics**
- **Evaluation**
- **Course**
- **Print**
- **Resources**

Student Learning Outcomes

Toggle the arrows to view the SLOs. New student learning outcomes (SLOs) are listed in red.

Microprocessor

1. Represent quantities in binary codes; convert between the decimal and binary number systems. Convert between the hex and binary codes.

2. Represent characters with and perform conversions with the ASCII code.

3. Perform basic math operations such as add, subtract, multiply and divide in binary.

4. Name the basic solid state memory types such as RAM and ROM, identify the various types like SRAM, DRAM, PROM, EPROM, EEPROM, flash, explain how each works, give the basic specifications, and name an application for each.

5. Name and explain the specifications for the following serial interfaces used in digital and computer systems: RS-232/UART, RS-485/422/423, USB, SPI, SCI, CAN, I2C, LIN, Flex, MOST, Ethernet 10/100/1000.

6. Relate data rate to transmission path bandwidth.

7. Explain the concept of sampling and the function of analog to digital and digital to analog converters including the importance of sampling rate, Nyquist criterion and resolution.

8. Name the basic sections and components of a digital computer.

9. Explain the stored program concept of a computer showing data and address flows using a block diagram of a CPU.

10. Distinguish between and define the terms microprocessor, CPU, microcomputer, embedded controller, microcontroller, and PLC.

11. Learn the basic commands and data formats for a common computer language such as BASIC and write simple programs to duplicate logic and math operations, control sequences, and I/O functions.

12. Describe the features and specifications of a common 8/16/32-bit microcomputer (PIC, 68HC11/12 or equivalent, 8051 derivative, etc.).

13. Identify the most popular 8, 16 and 32-bit microcontrollers.

14. Write simple data manipulation, math, and I/O routines in assembly language and C.

15. Interface a common embedded controller to some basic I/O circuits such as switches, LEDs, relays, serial I/O, display, keyboard, Analog-to-digital converter or other common device and write simple code to control/monitor it.

16. Use a typical integrated development environment (IDE) software package to program the learned microcomputer in assembler and C including software subroutine library.

17. Define digital signal processing (DSP), show a block diagram of how it is implemented, and list at least 3 common applications.

Laboratory

1. Program a basic microcontroller using a higher level language like BASIC. Implement basic logic functions including interfaces for I/O operations.

2. Generate a serial data signal and translate the pulses into data and determine the data rate.

3. Build a complete embedded controller project including I/O and programs.

4. Troubleshoot and test digital circuits with an oscilloscope and, if available, a logic analyzer.

5. Debug a program using the debugger section of a development system.

Microprocessor: Course Materials

This page contains **Syst** Microprocessor course materials and relevant information. Toggle the tabs below to view all the content of this course area.

To view specific Microprocessor course materials, select the "Materials" tab

- **Views**
- **Materials**
- **SLO**

Detailed Course Outline

Toggle the arrows to view the outlines. New topics are listed below in red.

Microcomputer Course

1. Basic digital computer architecture
   a. CPU
   b. Memory
   c. Input/output

2. Types of digital computers
   a. Supercomputers
   b. Servers
   c. Personal computers
   d. Microcomputers, microprocessors, embedded controllers

3. Memory circuits (If not covered in Digital course)
   a. Static memory basics
   b. Dynamic memory basics
   c. Memory specifications
   d. RAM chips organization and operation. IC decoding, storage and retrieval, refresh fundamentals
   e. ROM basics
   f. ROM types
   g. PROM
   h. EPROM
   i. EEPROM
   j. Flash, NAND and NOR types

4. Parallel Buses (If not covered in Digital course)
   a. Concepts and data transmission limitations
   b. Common PC buses: PCI, PCI-X, PCI Express, CML, LVDS

5. Serial Buses (If not covered in Digital course)
   a. Serial bus basics, data rates
   b. RS-232, RS-422/423, RS-485
   c. USB
   d. SPI
   e. I^2C
   f. CAN, LIN and Flex
   g. Ethernet 10/100/1G/10G
   h. Optical fiber

6. Architecture, organization, and operation of a microcomputer
   a. Registers
   b. ALU/math
   c. I/O
   d. Instruction fetch/execute

7. Introduction to a popular 8-bit (16 or 32-bit ok) embedded controller
   a. Organization and operation
   b. Survey of instruction set
   c. Special features like interrupts stack

8. Survey of popular microcontrollers
   a. 8-bit - HC11/12, 8051, PIC, Atmel, PSOC, Rabbit, Zilog, etc.
   b. 16-bit - TI MPS430, PIC
   c. 32-bit - ARM, MIPS, Power PC

9. Programming an embedded controller
   a. Survey of software (languages, operating systems, other)
   b. Use of integrated development system environment
   c. Programming basic functions in assembly language
   d. Programming basic functions in the C language
   e. Subroutines

10. Interfacing
    a. Use of parallel ports
    b. Use of serial interfaces
    c. Use of counters/timers
    d. Use of PWM
    e. Use of AD and DA

11. Development of a simple system
    a. Design
    b. Interface
    c. Software
    d. Testing and debugging

12. Introduction of digital signal processing (DSP)
    a. Concepts
    b. AD/DA review
    c. Common applications (filters, modulation/demodulation, frequency analysis)
    d. Programmable DSP
    e. DSP implementations in FPGA