

ELEC 54C Course Outline as of Fall 2018**CATALOG INFORMATION**

Dept and Nbr: ELEC 54C Title: MICROCONTROLLERS

Full Title: Microcontrollers and Embedded Systems

Last Reviewed: 8/14/2023

Units		Course Hours per Week		Nbr of Weeks	Course Hours Total	
Maximum	3.00	Lecture Scheduled	2.00	17.5	Lecture Scheduled	35.00
Minimum	3.00	Lab Scheduled	3.00	8	Lab Scheduled	52.50
		Contact DHR	0		Contact DHR	0
		Contact Total	5.00		Contact Total	87.50
		Non-contact DHR	0		Non-contact DHR	0

Total Out of Class Hours: 70.00

Total Student Learning Hours: 157.50

Title 5 Category: AA Degree Applicable

Grading: Grade Only

Repeatability: 00 - Two Repeats if Grade was D, F, NC, or NP

Also Listed As:

Formerly:

Catalog Description:

An introduction to computer programming concepts using microcontrollers to program computer circuits and hardware. Microcontrollers such as the Arduino and PicAxe will be used to interface with circuits constructed on prototyping boards. Course content includes both software and hardware troubleshooting.

Prerequisites/Corequisites:

Course Completion of ELEC 54B

Recommended Preparation:**Limits on Enrollment:****Schedule of Classes Information:**

Description: An introduction to computer programming concepts using microcontrollers to program computer circuits and hardware. Microcontrollers such as the Arduino and PicAxe will be used to interface with circuits constructed on prototyping boards. Course content includes both software and hardware troubleshooting. (Grade Only)

Prerequisites/Corequisites: Course Completion of ELEC 54B

Recommended:
Limits on Enrollment:
Transfer Credit: CSU;
Repeatability: Two Repeats if Grade was D, F, NC, or NP

ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:

AS Degree:	Area	Effective:	Inactive:
CSU GE:	Transfer Area	Effective:	Inactive:
IGETC:	Transfer Area	Effective:	Inactive:
CSU Transfer:	Transferable	Effective: Fall 2018	Inactive:
UC Transfer:		Effective:	Inactive:

CID:

Certificate/Major Applicable:
Not Certificate/Major Applicable

COURSE CONTENT

Student Learning Outcomes:

At the conclusion of this course, the student should be able to:

1. Write programs in a high-level programming language such as the C++ derived Arduino language to control two different microcontrollers.
2. Use a microcontroller to detect inputs from sensors.
3. Control LEDs, servo motors, speakers, and other devices in response to inputs and programming.

Objectives:

During this course students will:

1. Convert between number systems (binary and hexadecimal).
2. Develop truth tables for logic gates.
3. Download and run a program to a microcontroller.
4. Use editors to compose programming code and compilers to produce executable software.
5. Describe basic hardware interfaces at the conceptual level.
6. Program an Atmega32 to control a DC motor.
7. Build an Atmega32 microcontroller circuit on a prototyping board.

Topics and Scope:

- I. Computers and microcontrollers
 - A. Algorithms and problem solving
 - B. Setting up the microcontroller and programming environment
- II. Logic Gates
- III. Microcontrollers
 - A. Setting up the programming environment
 - B. Memory Concepts
 - C. Declarations and variables

- D. Assignment and initialization
- IV. Digital inputs and outputs
- V. Analog sensors
- VI. Selection (conditional) statements
- VII. Reading digital input pins
- VIII. Making decisions with if, if-else
 - A. Comparison operators and relational expressions
 - B. Boolean expressions
- IX. Repeating with loops
- X. Numbers and arithmetic
 - A. Analog measurements
 - B. Numerical types
 - C. Numerical operators
 - D. Arithmetic expressions
 - E. Mathematical functions
- XI. Functions
 - A. Introduction to functions
 - B. Defining a function to repeat an action
- XII. Arrays
 - A. Seven-segment LED displays
 - B. Using lists for data
 - C. Reading and writing array data
- XIII. Data file basics with Serial I/O
 - A. Reading from files
 - B. Writing to files
- XIV. Working with third-party libraries
- XV. Communications and Serial I/O
- XVI. Common hardware interfaces
 - A. UART
 - B. I2C
 - C. SPI
 - D. Interrupt driven I/O
- XVII. Common microcontrollers
 - A. Arduino
 - B. PicAxe
 - C. Intel
- XVIII. Constructing an Atmega328 microcontroller
 - A. Reading a Schematic
 - B. Constructing the circuit
 - C. Wiring inputs and outputs

Assignment:

Lecture-Related Assignments:

1. Textbook readings, 20-30 pages per week.
2. Homework assignments (10-15)
3. Quizzes (2-6), final exam

Lab-Related Assignments:

1. Lab assignments (8-14)
2. Program Documentation (8-14)

Methods of Evaluation/Basis of Grade:

Writing: Assessment tools that demonstrate writing skills and/or require students to select, organize and explain ideas in writing.

Program Documentation

Writing
20 - 30%

Problem Solving: Assessment tools, other than exams, that demonstrate competence in computational or non-computational problem solving skills.

Homework problems, lab assignments

Problem solving
30 - 40%

Skill Demonstrations: All skill-based and physical demonstrations used for assessment purposes including skill performance exams.

None

Skill Demonstrations
0 - 0%

Exams: All forms of formal testing, other than skill performance exams.

Quizzes and final exam

Exams
30 - 40%

Other: Includes any assessment tools that do not logically fit into the above categories.

None

Other Category
0 - 0%

Representative Textbooks and Materials:

Fundamentals of Digital Logic and Microcontrollers. 6th ed. Rafiquzzaman, M. Wiley. 2014