COMMON POOL OF GENERIC ELECTIVES (GE) COURSES OFFERED BY THE DEPARTMENT

GENERIC ELECTIVES (GE-1)

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
Microcontroller Systems	**4**	**3**	**-**	**1**	**Class XII passed with Physics + Mathematics/Applied Mathematics + Chemistry OR Physics + Mathematics/Applied Mathematics + Computer Science/Informatics Practices**	**Basic C language program-ming**

Learning Objectives

The Learning Objectives of this course are as follows:

- Understand architecture of Microcontroller.
- Write assembly language / C programs for the microcontroller.
- Apply knowledge and demonstrate proficiency of designing hardware interfaces for memory and I/O.

Learning outcomes

The Learning Outcomes of this course are as follows:

- Explain the concepts related to architecture of microcontrollers
- Demonstrate knowledge of the development tools for a microcontroller, and write assembly language code according to specifications
- Design systems for common applications like general I/O, counters, data acquisition etc.
- Interfacing the external devices to the controller according to the user requirements to create novel products and solutions for the real - time problems.

SYLLABUS OF ELGE-6A **Total Hours- Theory: 45 Hours, Practicals: 30 Hours**

UNIT – I (11 Hours)

Introduction to microcontroller: Introduction to Microcontroller based system, Difference between Microprocessor and Microcontroller, Classification of microcontrollers based on architecture and Instruction Set (Overview of Harvard architecture and Von Neumann architecture, RISC and CISC microcontrollers), Microcontroller Features - Brown out Detector, Watch Dog Timer.

UNIT – II (12 Hours)

Architectural Overview of AVR Microcontroller: Block diagram description of ATMEGA32, Pin Description of ATMEGA32, AVR status register, General Purpose Register File, X, Y & Z registers, Stack Pointer, System Clock and Clock Options in AVR, System Control and Reset, Sleep Modes, AVR ATmega32 Memories: Flash Program Memory, SRAM Data Memory, EEPROM Data Memory & I/O Memory.

UNIT – III (11 Hours)

Instruction set of AVR Microcontroller: Addressing modes, Instruction set of AVR microcontroller, Data transfer, Arithmetic, Logic and Compare, Rotate and Shift, Branch and Call instructions, Bit manipulation instructions, MCU Control Instructions, Simple programs in Assembly Language / C Language

UNIT – IV (11 Hours)

AVR on-chip peripherals: General purpose I/O Ports, AVR I/O Port Programming, Introduction to interrupts, External interrupts, 8 and 16-bit Timers, Timer programming.

Practical component (if any) – Microprocessor System
(Hardware and AVR Studio/ Other suitable IDE)

Learning outcomes

The Learning Outcomes of this course are as follows:

- Be proficient in use of IDE's for assembly/ C programming for the microcontroller.
- Interface various I/O devices to provide solutions to real-world problems.
- Prepare the technical report on the experiments carried.

LIST OF PRACTICALS (Total Practical Hours – 30 Hours)

1. Program to transfer a block of data.

2. Program to find the sum/subtraction of two 8-bit numbers.
3. Program to find the sum of N 8-bit numbers.
4. Program to find multiplication/ Division of two 8-bit numbers.
5. Program to find smallest of N numbers
6. Program to find the square root of 8-bit number.
7. Program to sort the numbers in ascending/ descending order.
8. Flash LED at observable rate.
9. Interface Input Switches and output LEDs.
10. Interface 7 segment display.

Note: Students shall sincerely work towards completing all the above listed practicals for this course. In any circumstance, the completed number of practicals shall not be less than nine.

Essential/recommended readings

1. Programming and Customizing the AVR Microcontroller, By Dhananjay Gadre, McGraw Hill Education

Suggestive readings

1. The AVR Microcontroller and Embedded Systems Using Assembly and C, By Muhammad Ali Mazidi, Sarmad Naimi and Sepehr Naimi, Pearson Education.
2. AVR ATmega32 data sheet- ATMEL Corporation

Note: Examination scheme and mode shall be as prescribed by the Examination Branch, University of Delhi, from time to time.

GENERIC ELECTIVES (GE-2)

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
Arduino/ Rpi App Development	4	2	-	2	Class XII passed with Maths/Applied Maths	Basic C language programming

Learning Objectives

This course introduces the student to the fundamental understanding of Arduino/Rpi processors. After completion of this course students should be well versed in programming the microcontroller. They should be able to use various sensors and make microcontroller respond to the external environment.

Learning outcomes

The Learning Outcomes of this course are as follows:

- Understand the basic concepts of Arduino Uno / Raspberry Pi and the programming environments.
- Understand digital and analog ports of a microcontroller and their usage.
- Understand the working of various sensors and their application in robotics.
- Design different circuits and display their outputs using LCD and other indicator

SYLLABUS OF ELGE-6B Total Hours- Theory: 30 Hours, Practicals: 60 Hours

UNIT – I (8 Hours)

Basic functionality of the Arduino/ Rpi board and its processor, Setting and configuring the board: Pin diagram of Arduino/Rpi development board, Integrated Development Environment (IDE), IDEs like AVR Studio, WIN AVR, ARM 11, Installing and configuring for Robot programming, In System Programmer (ISP), loading programs on Robot, Differentiating Arduino board from Rpi board.

UNIT – II (8 Hours)

Introduction of Embedded C Programming and programming concepts for Arduino/ LINUX for Rpi, Digital Ports: Data Read and Write, Interfacing LEDs, Buzzer, Switches, 7 segment displays, LED dot matrix, Traffic lights, Introduction to 2 x 16 Characters LCD, Basic LCD control, Displaying message on LCD.

UNIT – III (6 Hours)
Sensors: IR range sensor of different range, Analog IR proximity sensors, Ultrasound scanner, LDR, Gyroscope and Accelerometer, Magnetometer, GPS receiver.

UNIT – IV (8 Hours)
Communication with Arduino/ Raspberry Pi : Wired RS232 (serial) Communication, Wireless ZigBee Communication, USB Communication, Simplex infrared Communication (IR remote to robot), Reading and writing to SD card.

Practical component (if any) – Arduino/Rpi App Development
(Supporting IDE)

Learning outcomes

The Learning Outcomes of this course are as follows:

- Familiarize with the Arduino/Rpi microcontroller development boards.
- Understand interfacing of various display devices viz. 7-segment display, LED dot matrix, LCD.
- Understand various sensors, their applications and designing control experiments using

LIST OF PRACTICALS (Total Practical Hours- 60 Hours)

1. To blink an LED/interface a Buzzer using a digital pin of the processor.
2. To display binary count on LEDs using digital port of the processor.
3. To display decimal count on a 7-segment display.
4. To read data from a digital port of the processor and then display it on other digital port.
5. To print a message on LCD.
6. To display different patterns on LED dot matrix.
7. To read the voltage of a potentiometer using analog port of the processor and depict the variation on LEDs/LCD.
8. To interface IR proximity sensor to determine if some obstacle is nearby.
9. To interface Ultrasonic sensor to determine if some object is in the facing direction.
10. To interface LDR and display if its dark or bright on 7 segment/LCD.
11. To design a Traffic Light System
12. To design a Voice Control Home Automation
13. To design a PWM based variable system
14. To design a wireless appliance controlling system.

Note: Students shall sincerely work towards completing all the above listed practicals for this course. In any circumstance, the completed number of practicals shall not be less than twelve.

Essential/recommended readings

1. Michal Mc Roberts “Beginning Arduino” Second Edition, Technology in Action
2. Massimo Banzi, “Getting started with Arduino” 2nd Edition, Orelly 2011
3. Richard Blum, “ Arduino Programming in 24 Hours”, Pearson Education, 1st edition, 2015.

Suggestive readings

1. Simon Monk, “Raspberry Pi Cookbook: Software and Hardware Problems and Solutions”, O'Reilly Reprints; Second edition 2016

Note: Examination scheme and mode shall be as prescribed by the Examination Branch, University of Delhi, from time to time.