Pokhara University
Faculty of Science and Technology

Course Code.: ELX 320
Course title: **Embedded System (2-1-2)**
Nature of the course: Theory & Practice

Level: Bachelor

Full marks: 100
Pass marks: 45
Time per period: 1 hour
Total periods: 45
Program: BE

1. Course Description

The course typically focuses on the design, implementation, and analysis of computer systems that are integrated into larger devices or systems to perform dedicated functions. These systems often operate in real-time environments and are optimized for specific tasks. The key topics of the course are: Microcontrollers, Programming Embedded Systems, Real-time Operating Systems (RTOS), Input/Output (I/O) Devices, Interrupts and Timers, Power Management, Optimization and Embedded Software Development Tools and IoT based embedded systems.

2. General Objectives

The general objectives of this course are:

- To equip students with the knowledge and skills to design, develop, and implement efficient hardware-software systems for specific, real-time applications
- To acquaint students with microcontrollers, interface devices, manage real-time operations, and optimize system performance for various embedded applications like IoT, automotive, and robotics

3. Contents in Detail

Specific Objectives	Contents
• Define components, and list the characteristics of embedded systems • Differentiate between embedded systems and general-purpose computing systems • Learn the various applications and use cases of embedded systems (e.g., automotive, IoT, consumer electronics) • Learn the embedded system design process. • Gain insight into the internal structure of memory, and explore the utilization of counters, timers, and I/O ports	**Unit I: Introduction to Embedded Systems, Microcontrollers and Processors (5 hrs.)** 1.1 Definition and Overview of Embedded Systems 1.2 Embedded Systems vs. General-Purpose Systems 1.3 Applications and Domains (Automotive, Healthcare, Industrial Control, IoT, etc.) 1.4 Key Components: Microcontrollers, Sensors, Actuators, Peripherals 1.5 Overview of Popular Embedded Architectures (ARM, AVR, PIC) 1.6 Memory Architecture: RAM, ROM, Flash, EEPROM 1.7 Input/Output (I/O) Ports, Timers, Counters **1.8 Embedded Design Process: Requirement Analysis, System Design, Testing**

for communication with external devices	
• Provide in-depth knowledge of programming languages commonly used in embedded systems, such as Embedded C and Assembly language • Design, write, test, and debug software for embedded systems that meet specific functional requirements	**Unit II: Programming for Embedded Systems (5 hrs.)** 3.1 Embedded C Programming **for Microcontroller**: 3.1.1 Introduction to C for Embedded Systems 3.1.2 Data Types, Control Structures, and Pointers 3.1.3 Memory Management and Interrupt Handling 3.2 Assembly Language: 3.2.1 Basics of Assembly for Microcontrollers 3.2.2 Instruction Set Architecture (ISA) 3.2.3 Register-level Programming 3.2.4 Embedded Python for advanced boards like Raspberry Pi
• Emphasize the development of both practical and theoretical understanding of real-time systems • Gain proficiency in utilizing a leading Real-Time Operating System (RTOS) such as VxWorks • Evaluate the advantages and disadvantages of various concepts, including task scheduling, context switching, task synchronization, interrupt handling, resource sharing, deadlock, priority inversion, multithreading, and multitasking	**Unit III: Real-Time Operating Systems (RTOS)(5 hrs.)** 4.1 Concepts of Real-Time Systems 4.2 Introduction to RTOS (e.g., FreeRTOS, VxWorks) 4.3 Task Scheduling, Context Switching, Task Synchronization 4.4 Interrupt Handling in RTOS 4.5 Resource Sharing, Deadlock, and Priority Inversion 4.6 Multithreading and Multi-tasking in RTOS
• Gain insight about VHDL Programming, different modeling styles, data types, sub program and packages, test benches. • Learn to program in VHDL for combinational and sequential cicuits	**Unit IV: Embedded System Design using VHDL (5 hrs.)** 5.1 Introduction to VHDL 5.2 Different Modelling styles in VHDL for combinational and sequential circuits 5.3 Data types in VHDL 5.4 Sub program and Packages 5.5 VHDL realization for combinational and sequential circuits

• Understand the distinction between serial and parallel communication and when to use each in embedded applications. • Explore the different types of communication (e.g., device-to-device, device-to-network, and device-to-cloud) that are essential for embedded systems. • Understand Wireless Communication Protocols such as Bluetooth and BLE (Bluetooth Low Energy), Wi-Fi, ZigBee and LoRa	**Unit V: Communication Protocols (3 hrs.)** 5.1Serial Communication: 5.1.1UART, SPI, I2C 5.1.2Basics of CAN, LIN for Automotive Systems 5.2Wireless Communication: 5.2.1Bluetooth, ZigBee, Wi-Fi, LoRa, GSM/GPRS 5.3Networking: 5.3.1TCP/IP Basics in Embedded Systems 5.3.2Ethernet for Embedded Devices 5.3.3USB and HDMI Standards for Embedded Devices
• Learn the types of peripherals commonly used in embedded systems, including input devices (e.g., sensors, switches), output devices (e.g., displays, actuators), and communication devices (e.g., serial interfaces). • Explore Analog-to-Digital Converters (ADC) and Digital-to-Analog Converters (DAC) for handling analog sensor inputs and actuator control.	**Unit VI: Peripherals and Interfacing (4 hrs.)** 6.1Sensor Interfacing: Analog and Digital Sensors (e.g., temperature, humidity, motion) 6.1.1ADC/DAC Converters 6.2Actuator Interfacing: 6.2.1Motor Control (DC, Stepper, Servo) 6.2.2PWM for controlling brightness, speed, etc. 6.3Display Interfacing: 6.3.1 LCD, OLED, and Seven Segment Displays 6.4Memory Interfacing: 6.4.1 EEPROM, SD Cards
• Understand the various communication protocols used in IoT, such as MQTT, CoAP • Understand how to interface embedded systems with various sensors and actuators for IoT applications. • Familiarize with IoT Standards and Industry Protocols	**Unit VII: Internet of Things (IoT) and Embedded Systems (3 hrs.)** 7.1 Introduction to IoT Concepts 7.2Embedded System's Role in IoT 7.3IoT Communication Protocols (MQTT, CoAP) 7.4IoT Platforms (NodeMCU, ESP32) 7.5Cloud Integration for Embedded Systems (AWS IoT, Google Cloud)

Note: The figures in the parentheses indicate the approximate periods for the respective units.

10 Methods of Instruction

Lecture, tutorials, lab works, projects.

5. List of Tutorials

The following tutorial activities of 15 hours per group of maximum 24 students should be conducted to cover all the required contents of this course.

S.N.	Tutorials

1	Beginner-Level Projects: Home Automation System, Smart Door Lock System, Temperature and Humidity Monitoring, Obstacle Avoidance Robot, Digital Thermometer
2	Intermediate-Level Projects: Weather Station, Smart Irrigation System, Bluetooth-Controlled Car, Home Security System, Pulse Oximeter
3	Advanced-Level Projects: IoT-Based Energy Meter, Drone Control System, Smart Traffic Light System, Heart Rate Monitor using ECG, Wearable Health Monitor

6. Practical Works

S.N.	Practical works
1	Introduction of VHDL for simulation of digital logic circuits and its components
2	Design and simulate basic logic gates (AND, OR, NOT, NAND, NOR, XOR, XNOR) using VHDL
3	Implement and test combinational circuits such as multiplexers, demultiplexers, encoders, and decoders
4	Design and simulate basic arithmetic circuits, including half adders, full adders, and subtractors
5	Create and analyze sequential circuits like flip-flops (D, JK, T), counters (binary, decade), and shift registers
6	Develop and simulate finite state machines, including both Mealy and Moore models, for various applications

7. Evaluation system and Students' Responsibilities

Evaluation System

In addition to the formal exam(s) conducted by the Office of the Controller of Examination of Pokhara University, the internal evaluation of a student may consist of class attendance, class participation, quizzes, assignments, presentations, written exams, etc. The tabular presentation of the evaluation system is as follows.

External Evaluation	Marks	Internal Evaluation	Marks
Semester-End Examination	50	Class attendance and participation	5
		Tutorials and projects works	5+5
		Quizzes/assignments and presentations	10
		Internal Term Exam	25
Total External	50	Total Internal	50
Full Marks 50+50 = 100			

Students' Responsibilities:

Each student must secure at least 45% marks in the internal evaluation with 80% attendance in the class to appear in the Semester End Examination. Failing to obtain such score will be given NOT QUALIFIED (NQ) and the student will not be eligible to appear in the End-Term examinations. Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during the period. If a student fails to attend a formal exam, quiz, test, etc. there won't be any provision for a re-exam.

8. Prescribed Books and References

Text Book

1. Simon, D. E. (1999). *An embedded software primer* (Vol. 1). Addison-Wesley Professional.
2. Mazidi, M. A., Naimi, S., & Naimi, S. (2010). *AVR Microcontroller and Embedded Systems The* (p. 364). Pearson India.
3. Perry, D. L. (2002). *VHDL: programming by example* (Vol. 4). New York: McGraw-Hill.
4. Bahga, A., & Madisetti, V. (2014). *Internet of Things: A hands-on approach*. Vpt.

Reference Books

1. Peckol, J. (2019). *Embedded systems: A contemporary design tool* (2nd ed.). Wiley.
2. Vahid, F., & Givargis, T. (2010). *Embedded system design: A unified hardware/software introduction* (3rd ed.). Wiley.