B. Tech with

MINOR

in

EMBEDDED SYSTEMS

Academic Regulations, Course Structure and Syllabus

Effective from 2023-24 admitted batches

Offered by

Department of Electronics and Communication Engineering

KSRM College of Engineering (A) – Kadapa

(Approved by AICTE, Accredited by NAAC with A+ Grade and NBA and Affiliated to JNTUA, Anantapuramu)

ELIGIBILITY / REGISTRATION / AWARD OF MINOR

The primary objective of a minor degree is to provide students with a secondary area of study to broaden their knowledge, enhance their skill set, and potentially improve their career prospects. It allows students to explore interests beyond their major, potentially leading to a more well-rounded and competitive profile.

i) Minor degree is introduced by the respective departments offering B. Tech. programs and is applicable to all B.Tech (Regular and Lateral Entry) students admitted in Engineering & Technology.

ii) There is NO minimum CGPA requirement to register for Minor degree program. However, the student shouldn't have any course backlog **up to III Semester in the major degree.**

iii) A student shall earn **additional 18 credits for award of Minor degree** offered by the department other than parent department. This is in addition to 163 credits by a regular student and 123 Credits by a Lateral Entry student for the award of Major degree.

iv) A student is permitted to register for Minor in IV Semester after the results of III Semester are declared. Students shall register and pass in all the courses prescribed and being offered from V semester under the respective Minor degree.

v) Students have to attend classwork for courses under Minor degree beyond regular academic hours meant for major degree. Students can also undergo the courses under Minor through any proctored online platforms with the prior approval of the BoS Chairman and the HoD of the respective department offering Minor degree.

vi) The attendance for the registered courses under Minor and regular courses offered for Major degree in a Semester will be considered separately.

vii) A student shall have an aggregate of 75% attendance in all courses registered under Minor in that particular semester to become eligible for attending Semester-End examinations.

viii) The registration for the Minor will be cancelled, if the student is detained due to lack of attendance in Major.

ix) The registration for the Minor will be cancelled, if the student fails in any course of either Minor / Major in any semester from V to VIII Semester.

x) A student registered for Minor shall pass in all subjects that constitute the requirement for the Minor degree program. No class/division (i.e., second class, first class and distinction, etc.) will be awarded for Minor degree program.

xi) A separate grade sheet will be issued for the Minor degree courses semester-wise..

xii) If a student drops or is terminated from the Minor program, the additional credits so far earned cannot be converted into open or core electives; they will remain extra.

xiii) The Minor will be mentioned in the degree certificate as Bachelor of Technology in Civil Engineering with Minor in Mechanical Engineering.

xiv) There shall be a minimum enrolment of 20% OR 20 enrollments from the list of

eligible students to offer Minor program.

xv) There is no fee for registration of courses for Minor program offered.

xvi) A student can register for either Minor / Honor but not both.

xvii) Student shall submit an application for either Minor / Honor at least one week before the commencement of the V Semester.

MINOR PROGRAMS OFFERED

Offering Department	Title	Who can Register
Civil Engineering	Civil Engineering	B.Tech. EEE / ME / ECE / CSE / AIML / CSE(AIML) / CSE(DS)
Mechanical Engineering	3D Printing	B.Tech. CE / EEE / ECE / CSE / AIML / CSE(AIML) / CSE(DS)
	Industrial Engineering	B.Tech. CE / EEE / ECE / CSE / AIML / CSE(AIML) / CSE(DS)
Electrical and Electronics Engineering	Energy Systems	B.Tech. CE / ME / ECE / CSE / AIML / CSE(AIML) / CSE(DS)
Electronics and Communication Engineering	Embedded Systems	B.Tech. CE / EEE / ME / CSE / AIML / CSE(AIML) / CSE(DS)
Computer Science and Engineering	Computer Science and Engineering	B.Tech. CE / EEE / ME / ECE
	Data Science	B.Tech. CE / EEE / ME / ECE
	Computer Science and Engineering – Artificial Intelligence and Machine Learning	B.Tech. CE / EEE / ME / ECE

COURSE STRUCTURE

for

MINOR

in

EMBEDDED SYSTEMS

S. No	COURSE CODE	COURSE TITLE	Semester Offered	L	T	P	IM	EM	CR
1	2304571M	Embedded Systems Technology	V	3	0	0	30	70	3
2	2304572M	Principles of IOT	V	3	0	0	30	70	3
3	2304671M	Real Time Operating Systems	VI	0	0	3	30	70	1.5
4	2304672M	Principles of Wireless Networks	VI	3	0	0	30	70	3
5	2304673M	Principles of IOT Lab	VI	3	0	0	30	70	3
6	2304771M	Industrial Internet of things	VII	3	0	0	30	70	3
7	2304772M	Embedded Systems Lab	VII	0	0	3	30	70	1.5
Total				**15**	**0**	**6**	**210**	**490**	**18**

2304571M	MINOR IN EMBEDDED SYSTEMS EMBEDDED SYSTEMS TECHNOLOGY (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	L	T	P	C
		3	0	0	3

Pre-Requisites: Nil

Course Outcomes:
After completing the course, the student will be able to,

CO1. Understand the basics of embedded systems, including their history, classification, and processor selection.
CO2. Analyze different embedded processor architectures, including ARM, RISC, and application-specific processors.
CO3. Evaluate various communication interfaces and protocols
CO4. Implement rapid prototyping techniques using Arduino, sensors, and wearable system modules.
CO5. Develop and interface embedded GUI systems, including LCDs, touchscreens, and VGA cameras, for real-world applications.

SYLLABUS:

UNIT - 1: INTRODUCTION TO EMBEDDED SYSTEM (10 Periods)

Introduction to Embedded Systems, Embedded Systems Vs General Computing Systems, History of Embedded Systems, Classification, Major Application Areas, Embedded Processor Requirements, Features, Types, RISC Processors, Harvard Architecture, Super Harvard Architecture, Selection of Processors &Microcontrollers.

UNIT-II: ARCHITECTURE OF EMBEDDED SYSTEM PROCESSOR (10 Periods)

Embedded processor models, ARM core processor, Application specific processor like network processors, multimedia processors, industrial processors, superscalar processor, Advanced RISC processors. Architecture of Embedded OS, Categories of Embedded OS, Application Software, Communication Software, Development and Testing Tools.

UNIT-III: COMMUNICATION INTERFACES (09 Periods)

Need for Communication Interfaces, OSI Reference Model, Basic of Networks, Network Topology, RS232/UART, RS422/RS485, USB, Infrared, Ethernet, IEEE 802.11, Bluetooth, SPI, I2C, CAN, Wifi, FlexRay, LIN Bus, Zigbee.

UNIT-IV: RAPID PROTOTYPING (08 Periods)

Arduino platform, hardware and software, Sensor’s modules, Robo Control modules,

3D printing module, ADC module, wearable systems. etc.

UNIT-V: EMBEDDED GUI INTERFACING (08 Periods)

Arduino based graphic LCD, Touch screen, joy stick, VGA camera interfacing and programming in Python, Creative applications of Arduino, Design Examples & Case Studies of Embedded System: Digital Thermometer, Navigation Systems, Smart Card, RF Tag.

Total Periods: 45

TEXT BOOKS:

T1. An embedded Software Primer, David Simon, Pearson Publication 2021.

T2. Embedded system A Unified Hardware Software Introduction, Frank Vahid, John Wiley and Sons,2005.

REFERENCE BOOKS:

R1. Embedded System Architecture, Tammy Noergaard, Elsevier publication, 2014**.**

WEBRESOURCES:

1. https://nptel.ac.in/courses/108102045
2. https://developer.arm.com/documentation/101108/latest/

2304572M	MINOR IN EMBEDDED SYSTEMS PRINCIPLES OF IOT (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	L	T	P	C
		3	0	0	3

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Understand thefundamentals of IoT, including its architecture, characteristics, and challenges.
CO2. Analyze IoT communication protocols and networking techniques used in wireless sensor networks and M2M communication.
CO3. Design and manage IoT platforms, including device integration, service models, and application development.
CO4. Implement IoT networking and computing techniques, including cloud storage, APIs, and Python-based IoT programming.
CO5. Develop and evaluate IoT applications in real-world domains such as smart cities, connected vehicles, and industrial automation.

SYLLABUS:

UNIT-I: INTRODUCTION & BASIC OF IOT (11 Periods)

Definition, Characteristics, Physical and Logical Designs, challenges, Technological trends in IOT, IoT Examples, M2M.

UNIT-II: IOT COMPONENTS, COMMUNICATION AND NETWORKING (10 Periods)

Introduction to Sensing and Networking: Sensing & actuation, Wireless Senor network, Sensor nodes, Communication Protocols, M2M Communication, Networking Hardware, Networking Protocols.

UNIT-III: IOT SYSTEM MANAGEMENT (10 Periods)

Network Operator Requirements, IoT Platform Design Specification – Requirements, Process, Domain Model, Service, IoT Level, Function, Operational view, Device and Component Integration, Application development.

UNIT-IV: NETWORKING AND COMPUTING (07 Periods)

File Handling, Python Packages for IoT, IoT Physical Servers – Cloud Storage Models, Communication APIs.

UNIT-V: IOT CLOUDS AND DATA ANALYTICS (07 Periods)

RESTful Web API, Amazon Web Services for IoT, Apache Hadoop, Batch Data Analysis, Chef, Chef Case Studies, Puppet, NETCONF-YANG IoT Applications: case studies: smart cities, smarthome, connected vehicles, IndustrialIOT.

Total Periods: 45

TEXT BOOKS:

T1. Internet of Things – Architecture and Design Principles, R. Kamal, McGraw Hill, 2017 First Edition.

T2. Internet of Things – Architectures, Protocols and Standards, Simone Cirani, Wiley, 2018.

REFERENCE BOOKS:

R1. Enabling Things to Talk – Designing IoT Solutions with the IoT Architectural Reference Model, Alessandro Bassi**,** Springer**,** 2013**.**

2304671M	MINOR IN EMBEDDED SYSTEMS REAL TIME OPERATING SYSTEMS (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	L	T	P	C
		3	0	0	3

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Analyze the core features of VxWorks, including task management, synchronization, and its practical use in real-time embedded systems.
CO2. Explain the principles and characteristics of real-time operating systems and differentiate them from general-purpose systems.
CO3. Analyze process management and thread synchronization techniques for real-time applications.
CO4. Evaluate inter-process communication methods and memory management strategies in RTOS environments.
CO5. Demonstrate working knowledge of VxWorks features, multitasking models, and task control mechanisms in embedded systems.

SYLLABUS:

UNIT - I: INTRODUCTION TO OPERATING SYSTEM (08 Periods)

Introduction to Operating System, Computer Hardware Organization, BIOS and Boot Process, Multi-threading concepts, Processes, Threads, Scheduling.

UNIT - II: BASICS OF REAL-TIME CONCEPTS (10 Periods)

Terminology: RTOS concepts and definitions, Differences between GPOS and RTOS, Real-time design issues, Examples, Hardware considerations: logic states, CPU, memory, I/O, Architectures, RTOS building blocks, Real-Time Kernel.

UNIT - III: PROCESS MANAGEMENT (08 Periods)

Concepts, Scheduling, IPC, RPC, CPU Scheduling, Scheduling criteria, Scheduling algorithms, Threads: Multi-threading models, threading issues, Thread libraries, Synchronization, Mutex: Creating, deleting and prioritizing mutex, Mutex internals.

UNIT - IV: INTER-PROCESS COMMUNICATION AND MEMORY MANAGEMENT (10 Periods)

Messages, Buffers, Mailboxes, Queues, Semaphores, Deadlock, Priority inversion, Pipes, Memory Management: Process stack management, Run-time buffer size, Swapping, Overlays, Block/page management, Replacement algorithms, Real-time garbage collection.

UNIT-V: INTRODUCTION TO VX WORKS (09 Periods)

Salient features, Multitasking, Task state transition, Task control: Task creation and activation, Task stack, Task names and IDs, Task options, Task information, Task deletion and safety, Semaphore and message queue related functions.

Total Periods: 45

TEXT BOOKS:

T1. MicroC/OS-II: The Real–Time Kernel by J.J. Labrosse, Newnes, 2002.
T2. Real-Time Systems by Jane W. S. Liu, Prentice Hall, 2000.
T3. Operating Systems Internals and Design Principles by William Stallings, Pearson Education, 2014, Seventh Edition.

REFERENCE BOOKS:

R1. Advanced Programming in the UNIX® Environment by W. Richard Stevens, Pearson Education India, 2011 Second Edition.
R2. Real-Time System Design and Analysis by Philips A. Laplante, John Wiley & Sons, 2004 Third Edition.
R3. Linux for Embedded and Real-Time Applications by Doug Abbott, Newnes, 2011 Second Edition.
R4. VxWorks Programmer's Guide by Wind River Systems Inc., 1997.
R5. Real-Time Systems by Rajib Mall, Pearson Education, 2007 Second Edition.

WEB RESOURCES:

1. https://www.youtube.com/watch?reload=9&v=hek-CUcLlTw
2. https://learning.windriver.com/vxworks-essentials-the-basics

2304672M	MINOR IN EMBEDDED SYSTEMS PRINCIPLES OF WIRELESS NETWORKS (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	L	T	P	C
		3	0	0	3

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Demonstrate the system and protocol architecture of IEEE 802.11 WLAN standards Including 802.11a/b.

CO2. Analyze and simulate routing protocols in mobile ad hoc networks, especially DSDV.

CO3. Compare TD-CDMA and TD-SCDMA technologies and their application in different network structures.

CO4. Discuss the vision, features, and challenges associated with 4G technologies.

CO5. Understand basic 5G communication scenarios.

SYLLABUS

UNIT - I: WIRELESS LAN (12 Periods)

Introduction-WLAN technologies: - IEEE802.11: System architecture, protocol architecture, 802.11b, 802.11a – Hiper LAN: WATM, BRAN, HiperLAN2 – Bluetooth: Architecture, WPAN – IEEE 802.15.4, Wireless USB, Zigbee, 6LoWPAN, Wireless HART

UNIT - II: MOBILE NETWORK LAYER (09 Periods)

Introduction - Mobile IP: IP packet delivery, Agent discovery, tunneling and encapsulation, IPV6-Network layer in the internet- Mobile IP session initiation protocol - mobile ad-hocNetwork: Routing: Destination Sequence distance vector, IoT: CoAP.

UNIT - III: 3G OVERVIEW (08 Periods)

Overview of UTMS Terrestrial Radio access network-UMTS Core Network Architecture: 3GPP Architecture, User equipment, CDMA2000 overview- Radio and Network components, Network structure, Radio Network, TD-CDMA, TD – SCDMA.

UNIT - IV: 4G & BEYOND (08 Periods)

Introduction – 4G vision – 4G features and challenges - Applications of 4G – 4G Technologies: Multicarrier Modulation, Smart antenna techniques, IMS Architecture, LTE, Advanced Broadband Wireless Access and Services.

UNIT - V: 5G RADIO INTERFACE ARCHITECTURE (08 Periods)

5G architecture options, core network architecture, RAN Architecture. 5G PHYSICAL LAYERS: Physical channels and signals, 5G frame structure, Physical layer procedures (MIMO, Power control, link adaptation, beam forming).

Total Periods: 45

TEXT BOOKS:

T1. Mobile Communications, Jochen Schiller, Pearson Education, 2012Second Edition.

T2. Wireless Communications and Networking, Vijay Garg, Elsevier, 2007First Edition.

T3. 5G Technology: 3GPP New Radio, HarriHolma, Antti Toskala, Takehiro Nakamura, John Wiley & Sons, 2020First Edition.

REFERENCE BOOKS:

R1. 3G Evolution: HSPA and LTE for Mobile Broadband, Erik Dahlman, Stefan Parkvall, Johan Skold, Per Beming, Academic Press, Second Edition, 2008.

R2. Wireless Networking, Anurag Kumar, D. Manjunath, Joy Kuri, Elsevier, First Edition, 2011.

R3. Network Function Virtualization: Concepts and Applicability in 5G Networks, Ying Zhang, John Wiley & Sons, First Edition, 2018.

WEBRESOURCES:

1. https://www.itu.int/en/ITU-R/study-groups/rsg5/Pages/default.aspx
2. https://www.qualcomm.com/invention/5g

2304673M	MINOR IN EMBEDDED SYSTEMS PRINCIPLES OF IOT LAB (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	L	T	P	C
		0	0	3	1.5

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Demonstrate the ability to control and monitor sensors and actuators using microcontrollers and IoT platforms.
CO2. Develop web-based IoT applications using HTTP and MQTT protocols for remote device management.
CO3. Apply IoT principles in real-world applications such as home automation, security systems, and industrial monitoring.
CO4. Apply appropriate Tools and Techniques to understand and analyze the problems following professional ethics with a focus on societal and environmental aspects.
CO5. Work as a team and communicate results in an effective way.
CO6. Make decisions as an individual or as team member to manage tasks and also engage in independent and lifelong learning with the ability to adapt to new and technological changes.

List of Experiments: (Any 10 Experiments are to be conducted)

1. **Serial Monitor, LED, Servo Motor - Controlling**
   Controlling actuators through Serial Monitor. Creating different led patterns and
   a. controlling them using push button switches. Controlling servo motor with the help of
   b. joystick.

2. **Distance Measurement of an object**
   Calculate the distance to an object with the help of an ultrasonic sensor and display it on
   a. an LCD.

3. **LDR Sensor, Alarm, and Temperature and humidity measurement**
   Controlling relay state based on ambient light levels using an LDR sensor.
   Basic burglar alarm security system with the help of a PIR sensor and buzzer.
   Displaying humidity and temperature values on LCD

4. **Experiments using Raspberry Pi**
   Controlling relay state based on input from IR sensors
   Interfacing stepper motor with R-Pi
   Advanced burglar alarm security system with the help of a PIR sensor, buzzer and keypad. The alarm gets disabled if correct keypad password is entered.

Automated LED light control based on input from PIR (to detect if people are present) and
LDR (ambient light level)

5. **IOT Framework**

   Upload humidity & temperature data to Thing Speak, periodically logging ambient light
   a. level to Thing Speak.
   Controlling LEDs, relay & buzzer using Blynk app

6. **HTTP Based**
   Introduction to HTTP. Hosting a basic server from the ESP32 to control various digital
   a. based actuators (LED, buzzer, relay) from a simple web page.

7. **Displaying various sensor readings on a simple web page hosted on the ESP32.**

8. **MQTT Based**
   Controlling LEDs/Motors from an Android/Web app, Controlling AC Appliances from an
   a. Android/web app with the help of Relay.

9. **Displaying humidity and temperature data on a web-based application**

10. **UAV/Drone:**
    Demonstration of UAV elements, Flight Controller
    Mission Planner flight planning design

11. **Python program to read GPS coordinates from Flight Controller**

REFERENCE:

R1. Designing the Internet of Things, Adrian McEwen, Hakim Cassimally, Wiley Publications, 2012.
R2. Business Model Generation, Alexander Osterwalderand Yves Pigneur, Wiley, 2011.
R3. Internet of Things: A Hands-On Approach, ArshdeepBahga,VijayMadisetti, Universities
Press, 2014.
R4. The Internet of Things, Enabling technologies and use cases, Pethuru Raj, Anupama C. Raman,
CRC Press.

WEBRESOURCES:

1. https://www.arduino.cc/
2. https://www.raspberrypi.org/

2304771M	**MINOR IN EMBEDDED SYSTEMS INDUSTRIAL INTERNET OF THINGS** (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Explain the key differences between IoT and IIoT, sensor calibration, placement, and the role of communication systems in IIoT applications.
CO2. Demonstrate knowledge of IIoT methodologies, including operating systems, networking protocols, and wireless sensor networks.
CO3. Apply data analytics techniques, including big data processing and machine learning, to industrial IoT applications.
CO4. Analyze and implement IP and non-IP communication protocols for efficient IIoT connectivity and data exchange.
CO5. Utilize IoT cloud platforms for device localization, tracking, and robotics applications in
industrial environments.

SYLLABUS:

UNIT - I: INTRODUCTION IIOT, MARKET SIZE AND POTENTIAL **(10 Periods)**

Definition, IoT v IIoT, Next Generation Sensors, Sensor's calibration and validate sensor measurements, placement of IoT devices, sensors, low-cost communication system design, Top application areas include manufacturing, oil & gas, Embedded systems in the Automotive and Transportation market segment.

UNIT - II: IIOT METHODOLOGY **(09 Periods)**

Top operating systems used in IIoT deployments, Networking and wireless communication protocols used in IIoT deployments. Smart Remote Monitoring Unit, components of monitoring system, control and management, Wireless Sensor Network(WSN).

UNIT - III: DATA DRIVEN ANALYTICS OF IIOT **(08 Periods)**

Implementing industrial IoT Data flow, big data and how to prepare data for machine learningalgorithms, Machine Learning algorithms, supervised learning & Un-supervised learning algorithms, Basics of neural network, activation functions, back-propagation.

UNIT - IV: IP AND NON-IP PROTOCOLS FOR IOT **(08 Periods)**

WPAN, IEEE 802.15.4, Bluetooth, NFC, 6LoWPAN; RFID, Zigbee Wireless HART

Protocol,MQTT, IP and Non-IP Protocols, REST, CoAP

UNIT - V: IOT CLOUDS AND DATA ANALYTICS (09 Periods)

Develops a physics-based and data-driven digital equipment model to monitor assets and systems, Introduction to device localization and tracking; different types of localization techniques, Radio-Frequency Identification (RFID) and fingerprinting, Device diversity/heterogeneity issue in IIoT networks

Internet of Robotic Things (IoRT): Introduction to stationary and mobile robots, Brief introduction to localization, mapping, planning, and control of robotic systems; Introduction to cloud-enabled robotics; Applications of IIoT in robotics; Architectures for IoRT, Examples and case studies: Open issues and challenges.

Total Periods: 45

TEXTBOOKS:

T1. Industry4.0: The Industrial Internet of Things, Alasdair Gilchrist, Apress, 2016.

T2. Introduction to Industrial Internet of Things and Industry4.0", SudipMisra, Chandana Roy, Anadarup Mukherjee, CRC Press, 2021.

REFERENCE BOOKS:

R1. Hands on Industrial Internet of Things, Giacomo Veneri, Antonio Capasso, Packt Press, 2018.

WEB RESOURCES:

1. https://en.wikipedia.org/wiki/IEEE_Internet_of_Things_Journal?utm_source=chatgpt.com
2. https://www.solisplc.com/industrial-internet-of-things-iiot?utm_source=chatgpt.com
3. https://en.wikipedia.org/wiki/IEEE_Internet_of_Things_Journal?utm_source=chatgpt.com

2304772M	**MINOR IN EMBEDDED SYSTEMS** **EMBEDDED SYSTEM LAB** (CE, EEE, ME, CSE, CSE-DS, CSE-AIML, AIML)	**L**	**T**	**P**	**C**
		0	**0**	**3**	**1.5**

Pre-Requisites: Nil

Course Outcomes:

On successful completion of the course, student will be able to

CO1. Develop Assembly Language Programs (ALP) to perform arithmetic operations using ARM Cortex-M3 processor registers.

CO2. Demonstrate proficiency in writing and executing Embedded C programs for ARM Cortex-M3 using Keil Vision IDE1.

CO3. Interface and control basic electromechanical devices like DC and stepper motors using GPIO and PWM techniques.

CO4. Able to apply appropriate Tools and Techniques to understand and analyze the problems following professional ethics with focus on societal and environmental aspects.

CO5. Work as a team and communicate results in an effective way.

CO6. Make decisions as an individual or as team member to manage tasks and also engage in independent and life-long learning with ability to adapt to new and technological changes.

List of Exercises/List of Experiments:

A minimum of Ten experiments are to be conducted

PART-A: Conduct the following Study experiments to learn ALP using ARM Cortex M3

Registers using an Evaluation board and the required software tool.
1. ALP to multiply two 16 bit binary numbers.
2. ALP to find the sum of first 10 integer numbers

PART-B: Conduct the following experiments on an ARM CORTEX M3evaluation board using

Evaluation version of Embedded 'C' & Keil uVision-4tool/compiler.

1. Display “Hello World” message using Internal UART.
2. Interface and Control a DC Motor.
3. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.
4. Interface a DAC and generate Triangular and Square waveforms.
5. Interface a 4x4 keyboard and display the key code on an LCD.

6. Using the Internal PWM module of ARM controller generate PWM and vary its duty cycle.
7. Demonstrate the use of an external interrupt to toggle an LED On/Off.
8. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.
9. Interface a simple switch and display its status through a relay, Buzzer and LED.
10. Measure ambient temperature using a sensor and SPI ADC IC.

REFERENCE BOOKS/LABORATORY MANUALS:

1.The Definitive Guide to the ARM Cortex-M3, Joseph Yiu, Newnes (Elsevier), 2010, Second Edition.