U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

June 2025 Semester End Main Examinations

Programme: B.E. **Semester: VI**

Branch: Electronics and Instrumentation Engineering **Duration: 3 hrs.**

Course Code: 23EI6PCESD / 22EI6PCESD **Max Marks: 100**

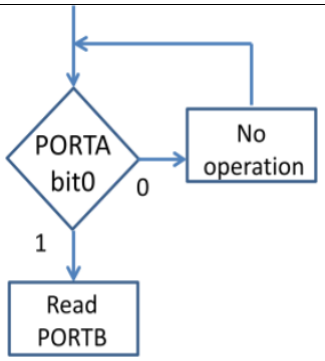
Course: Embedded System Design

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

			UNIT - I	*CO*	*PO*	Marks
1	a)		Explain an Embedded system with the help of a block diagram.	*CO1*	*PO1*	**07**
	b)		List and explain the characteristics of an Embedded system.	*CO1*	*PO1*	**08**
	c)		How is program memory in an embedded system classified based on its type for storing program code? Discuss	*CO1*	*PO1*	**05**
			OR			
2	a)		Explain the significance of throughput and response time as key quality attributes of an embedded system, providing a relevant example to illustrate each	*CO1*	*PO1*	**07**
	b)		Evaluate the key phases of the embedded system product development life cycle, with a supporting diagram to demonstrate the interdependencies and flow between stages	*CO1*	*PO1*	**08**
	c)		Describe (i) Interrupt Latency (ii) Real time Issues for Embedded system design	*CO1*	*PO1*	**05**
			UNIT - II			
3	a)		It is required to build a remote temperature monitoring system for a plant consisting of a furnace. Assuming that the RTD measuring the temperature is already placed near the furnace; develop a digital temperature measurement, storage and transmitting system using a microcontroller. The solution should consist of a block diagram comprising the necessary electronic circuit for signal conditioning and the programming modules to be used in microcontroller. Also elaborate on working of the proposed system.	*CO2*	*PO2*	**07**
	b)		Analyze the following code and discuss about its output. unsigned char count; void mathop () { int i;	*CO2*	*PO2*	**07**

			static int j;
			if (count == 0)
			j = 0;
			i = count;
			j = j + i;
			}
			void main(void) {
			count = 0;
			while (1) {
			mathop ();
			count++;
			}			

		c)	 The above flow chart show, "Wait for a 1 to be applied to bit 0 of GPIOA and then read GPIOB". Write Embedded 'C' code to illustrate this operation	*CO2*	*PO2*	**06**
			OR			
	4	a)	What is the role of volatile in memory optimization? Explain the difference between stack memory and heap memory in embedded systems.	*CO2*	*PO2*	**07**
		b)	What is an Interrupt Service Routine (ISR), and how is it implemented in embedded C?	*CO2*	*PO2*	**06**
		c)	How do functions in embedded C handle pass-by-value and pass-by-reference, and what are the key differences between these two approaches? Discuss with an example for each.	*CO2*	*PO2*	**07**
			UNIT - III			
	5	a)	Explain how instruction pipelining works in the ARM Cortex-M3 processor and describe its effect on instruction execution	*CO3*	*PO2*	**07**
		b)	Discuss the on-chip peripherals of the STM32F1xx Cortex-M3 microcontroller and illustrate their interconnection with the ARM CPU using a detailed block diagram	*CO3*	*PO2*	**08**
		c)	Describe the memory organization of the STM32 microcontroller using its memory model, and explain the concept of memory-mapped I/O operations	*CO3*	*PO2*	**05**

			OR			
	6	a)	What is an Interrupt Request (IRQ) handler in an ARM-based STM32 microcontroller, and how is it used for handling peripheral interrupts? Illustrate your explanation with a suitable example	*CO3*	*PO2*	**08**
		b)	Identify the four key sequences of operations and the essential software tools involved in executing an Embedded C program on hardware during the development of an application and explain.	*CO3*	*PO2*	**07**
		c)	Explain the Embedded 'C' data types used in Embedded 'C' programming using STM Microcontroller	*CO3*	*PO2*	**05**
			UNIT - IV			
	7	a)	List the different modes of operation for General-Purpose Input/Output (GPIO) ports in STM32 microcontrollers. For the following modes, explain the register configurations required for their operation: (i)Simple Input / Output mode. (ii) Alternate Function mode.	*CO4*	*PO2*	**07**
		b)	Using register-level programming, explain the process of serial data transmission with the inbuilt USART peripheral in the STM32F1xx microcontroller. Include the necessary register configurations and their roles in enabling communication	*CO4*	*PO2*	**08**
		c)	Identify the blocks associated with Timer operation and also write the program if the timer is having TMIxCLK as 48 MHz and Prescaler and ARR values are 49999 and 3 respectively.	*CO4*	*PO2*	**05**
			OR			
	8	a)	Write an Embedded C program to demonstrate the operation of GPIO programming in an STM32 microcontroller. The program should switch on an LED connected to pin PC13 when a switch connected to pin PA5 is closed. Include the necessary GPIO configurations and logic. Additionally, provide a port connection diagram to illustrate the hardware setup	*CO4*	*PO2*	**10**
		b)	Differentiate Bluetooth from ZigBee protocol in terms of number of users , power consumption and data rate supported. Show that the protocols are built using the underlying USART communication module of the ST microcontroller	*CO4*	*PO2*	**10**
			UNIT - V			
	9	a)	What are the key components of the ARM-GCC toolchain**,** and how does it facilitate cross-compilation for ARM-based embedded systems? Discuss	*CO5*	*PO2*	**07**

		b)	What is the role of an emulation board in embedded system development, and how does it differ from a real hardware prototype in terms of debugging and performance analysis?	*CO5*	*PO2*	**08**
		c)	Write a C program that runs on an ARM-based embedded system using the ARM-GCC toolchain. The program should: i. Print "Hello,ARM!" to the console. ii. Perform a simple bitwise operation on an integer	*CO5*	*PO2*	**05**
			OR			
	10	a)	Explain the open architecture of Raspberry -Pi , as an example for ARM6 microprocessor.	*CO5*	*PO2*	**10**
		b)	How does RTOS help in handling timing constraints while multiple tasks are to be executed on a single core processor? Discuss considering a programming example with three tasks	*CO5*	*PO2*	**10**

****