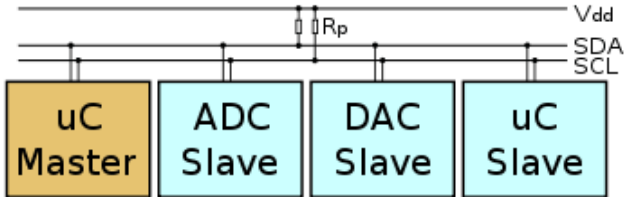
	RV College of Engineering® Department of Computer Science and Engineering CIE - II: Test and Quiz Paper		
Course & Code	IOT and Embedded Computing (CS344AI)	Semester: 4 th Sem BE	
Date : July 2024	Duration:120 minutes	Max.Marks:(10+50)=60 Marks	Staff : KB, SDV, MSS, MH
USN :	Name :		Section : A/B/C/D/CD/CY

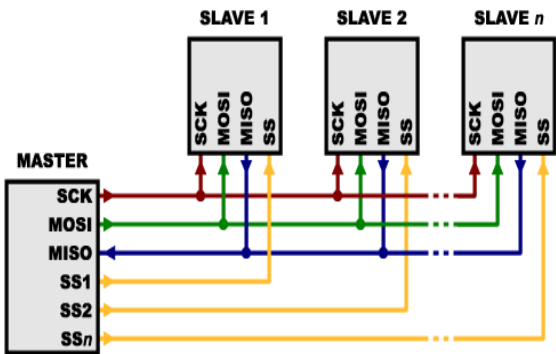
NOTE: Answer all the questions from Part-A (10 M) and Part-B (50 M)

Sl.no	PART - A	Marks	* BT	*CO
1	<p>Indicate the value to be loaded into match Register MR0, so that timer counter T0TC reaches the MR0 value after 5 milliseconds. Assume the PCLK = 10MHz, CCLK=40MHz, T0TC=0, Pre-scaler Register=0</p> <p>Ans: 50000</p>	2	L2	CO3
2	<p>Calculate the delay produced by the following program run on LPC2148. Given PCLK = 15MHz. Choose the answer in milli-seconds.</p> <pre>void delay(void) { T0MCR = 0X04; T0TC = 0X00; T0MR0 = 75000; T0TCR = 0X01; while(T0TC != T0MR0); T0TCR = 0X02; }</pre> <p>Ans: 5ms</p>	2	L3	CO2
3	<p>Given PCLK=15MHz, Required baud rate=9600, Choose the values of DLM:DLL. (Assume DivVal=0, MulVal=1)</p> <p>Ans: U0DLM=00;U0DLL=97;</p>	2	L2	CO2
4	<p>What are the different types of communication models used in IoT.</p> <p>Ans:</p>	2	L2	CO2

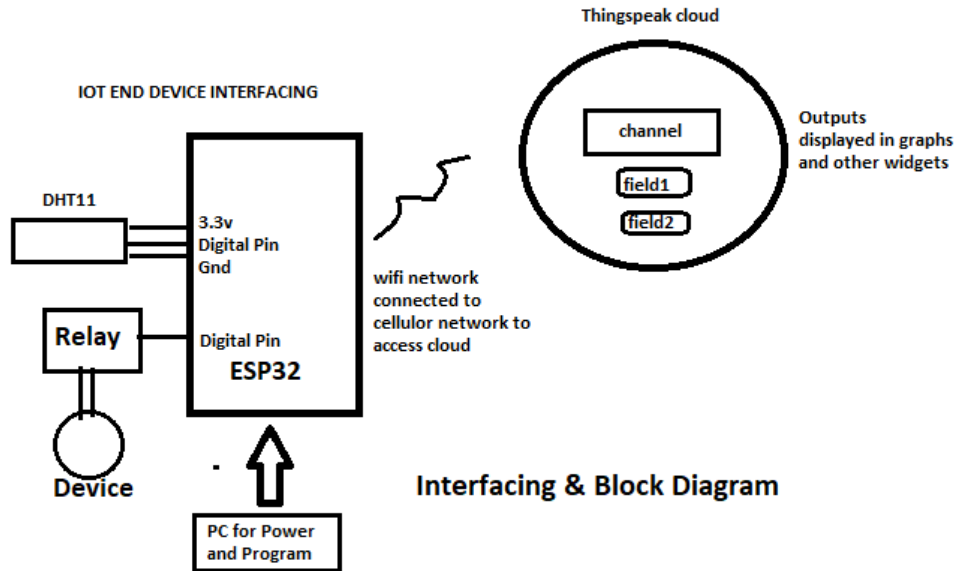
5	List any four most commonly used sensors in IoT and mention any two applications of PWM in IoT Ans: Sensors- Temperature, Humidity, Moisture, Air Pollution, Vibration PWM Applications: LED Lighting, Servo Motor Control, DC Motor Control	2	L3	CO3
---	--	---	----	-----

Sl.no.	PART - B	Marks	* BT	*CO
1a.	Generate the 200KHz, 25% duty cycle waveform using LPC 2148 PWM channel. Assume PCLK = 15MHz. Make suitable assumptions, and explain clearly the calculations and the working of the program. Sindhu madam	5	L2	CO2
1b.	Generate the 10KHz square waveform using LPC 2148 GPIO pin P0.1. Use timers to calculate the timings and assume PCLK = 60MHz. Explain the working of the program Sindhu madam	5	L2	CO2
2a.	Design an activity LED (one which is blinking once in 10 seconds to indicate the system/product is working) using interrupts and timers, with suitable comments <pre>#include <LPC2148x.h> unsigned int x=0; __irq void Timer0_ISR(void) // an ISR program { x = x ^ 1; if (x) I0SET1 = 1 << 16; //P1.16 = 1 else I0CLR1 = 1 <<16; // P1.16 = 0 T0IR = 0x01; // clear match0 interrupt, and get ready for the next interrupt VICVectAddr = 0x00000000 ; //End of interrupt } int main(void) { I0DIR1 = 0x0001 0000; //set P1.16 as output T0TCR = 0x00; // stop the timer, to initialize different registers T0MCR= 0x0003; // Enable Interrupt and reset timer after match T0TC = 0x00; // make TC = 0</pre>	5	L2	CO3

	<pre> T0MR0 = 150000; // generates 10ms //load interrupt related registers , assigning Timer0 to IRQ slot 4 VICVectAdd4 = (unsigned long)Timer0_ISR; // set the timer ISR vector address VICVectCntl4 = 0x0000024; // set the channel VICIntEnable = 0x0000010; // enable the timer0 interrupt T0TCR = 0x01; // start the timer while(1) { //do other works }; // now timer interrupt is serviced automatically using the ISR } </pre>			
2b.	<p>Discuss the Features and Applications of serial protocols I2C and SPI</p>  <p>I2C Features</p> <ul style="list-style-type: none"> ▪ It is multi-master, multi-slave, packet switched, single-ended, serial computer bus. It is widely used for attaching lower-speed peripheral IC's to processors and microcontrollers in short-distance, intra board communication. ▪ Support multi master system, more than one master can communicate with the devices, for every 8 bits of data sent, one extra bit of meta data (ACK/NACK bit) must be transmitted. ACK/NACK bit gives confirmation that each frame is transferred successfully. ▪ Only uses two wires (Serial, half duplex), Hardware is less complicated than with UARTs. The hardware required to implement I2C is more complex than SPI but simpler than UART. ▪ It Supports 128 devices (7bit address) in normal mode. ▪ Data transfer rates up to 100 kbits/s and 7-bit addressing possible in normal mode. (It supports 400Khz (fast mode), 1Mhz –fast mode plus, 3.4Mhz for high speed mode, 5Mhz for ultra fast mode) <p>SPI (Note: connections for one device good enough)</p>	5	L3	CO1

	 <ul style="list-style-type: none"> ➤ SPI Interface uses four wires for communication. Hence it is also known as four wire serial communication protocol. ➤ SPI is a full duplex master-slave communication protocol. This means that only a single master and a single slave can communicate on the interface bus at the same time. It has separate send & receive lines unlike I2C. ➤ SPI enabled devices work in two basic modes of SPI operation i.e. SPI Master Mode and SPI Slave Mode. Master Device is responsible for initiation of communication. Master Device generates Serial Clock for synchronous data transfer. There is always only one master (most of the times it is microcontroller). ➤ Faster than asynchronous serial (UART), operate around 1Mhz. (can go upto 10Mhz) ➤ <i>Hardware requirement for SPI is very simple (as simple as shift register) compare to UART & I2C.</i> ➤ Master Device can handle multiple slave devices on the bus by selecting them one by one using multiple slave select pins. In general, each slave will need a separate SS line. 			
3a.	<p>Define IoT and Explain the functional blocks of IoT with the help of neat block diagram.</p> <p>Refer Reference book</p>	5	L3	CO3
3b.	<p>Suggest (With brief description) any one-use case of IOT pertaining to following domains: Energy, Retail, Logistics, Agriculture, Cities.</p> <p>Refer Reference book</p>	5	L4	CO1
4a.	<p>Design an IOT Level 2 deployment application for weather monitoring and Device control in the house using ESP32 and Thing speak cloud platform, with suitable block diagram, interfacing, flowcharts and brief</p>	10	L4	CO2

description. The proposed system consists of single node that monitors the room temperature and humidity using DHT 11 sensor, and based on the temperature / humidity, device(fan) should be turned on using a Relay. The controller also sends the sensor data to the cloud, where it will be displayed on the dash board.



Algorithm:

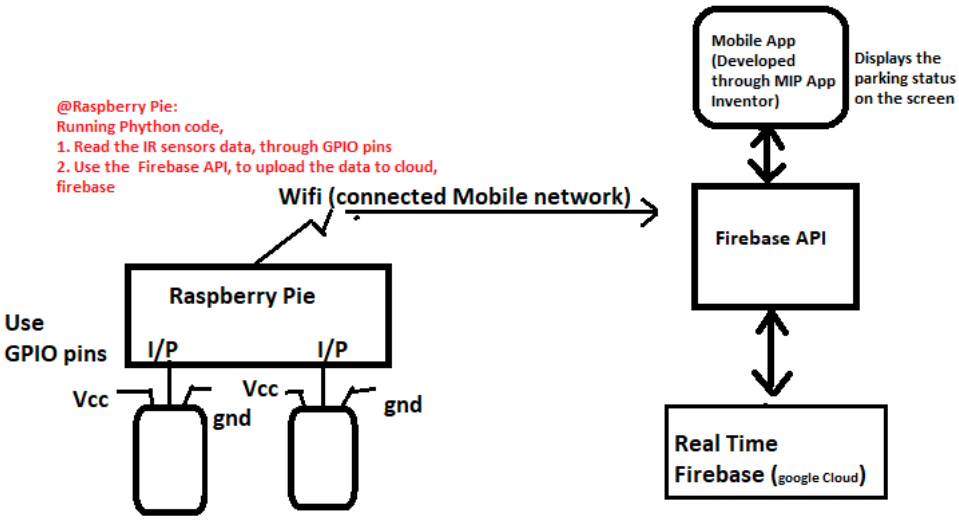
On the Cloud:

1. Goto thingspeak cloud, do following things: Create the channel and two fields for storing temperature and humidity
2. Copy the Channel no and write API to be used at the IOT Edge device.

On the End Device:

1. Import the Libraries for DHT11, ThingSpeak cloud
2. Initialize the required variables, libraries (start functions) with appropriate digital pins used for interfacing. Set the temperature threshold for fan/device control
3. Read the temperature and humidity
4. Upload the values to the thingspeak cloud channel fields, using the channel ID and fields.
5. Compare the room temperature with threshold and switch on / off the relay to control the device/fan.
6. delay, after each reading
7. repeat the steps 3 - 6

4b.	Design an IOT Leve2 deployment application for Smart Parking using RasberryPie with IR sensors and Cloud with Mobile Application to show the parking slots status. Draw the block diagram, interfacing, flowchart	5	L4	CO2
-----	---	---	----	-----

	<p>and brief description.</p>  <p>@Raspberry Pie: Running Python code, 1. Read the IR sensors data, through GPIO pins 2. Use the Firebase API, to upload the data to cloud, firebase</p> <p>Hardware Intefacing: IR sensors will have 3pins, Vcc,Gnd,Data. IR sensor is used to identify the presence / absence of vehicle. Raspberry Pie through its GPIO pins reads the data from IR sensors (connections shown in the above diagram). Wifi is used (through password and user name) to connect to the cellular mobile network (wifi router /mobile hotspot).</p> <p>The Rpie runs the python code to do the following tasks:</p> <ol style="list-style-type: none"> 1. Read the IR sensors 2. Upload the data to firebase cloud using firebase API, periodically <p>The above steps 1-2 are repeated.</p> <p>The Mobile App uses firebase API to read the digital values stored in the firebase real time database (key-value pairs). The app visualizes, whether slot is available or not by displaying red/green boxes.</p>			
5	<p>Interface LDR and LED bulb to LPC 2148 and write an embedded C program to read the data from LDR and suitably turn on/off the LED bulb and also send the suitable message to computer using UART interface. Clearly show the connections between LPC 2148 and Computer Serial Port and explain the UART initialization steps, clearly showing the registers used and the baud rate calculations.</p> <p>Refer the notes (Unit 2,3)</p>			

Course Outcomes: After completing the course, the students will be able to:-	
CO 1	Apply Embedded System and IoT fundamentals and formulate sustainable societal relevant cost effective solutions.
CO 2	Demonstrate the development of software programs using Embedded C, using Microcontrollers and different sensors and peripherals to build embedded system applications.
CO3	Design smart systems using various I/O peripherals, Sensors, embedded protocols like UART,I2C,SPI

	using modern tools like Keil IDE software for various domains like Healthcare, automation, agriculture, smart cities and others.										
CO 4	Indulge in developing Novel multi-disciplinary IoT projects using prototype boards, with effective oral & written communication skills and working in teams.										
CO 5	Engage in Lifelong Learning by investigating and executing real world societal problems using engineering tools – Cross compilers, debuggers and simulators, emerging processor and controller-based hardware platforms, IOT cloud infrastructure & protocols.										
BT LEVELS	L1	L2	L3	L4	L5	L6	COS	CO1	CO2	CO3	CO4
MARKS		10	30	10					20	30	