

195

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

TEST -2 EXAMINATIONS- 2026

B.Tech-IV Semester (CSE/IT/BT/ECE)

COURSE CODE (CREDITS): 25B1WCI433 (2)

MAX MARKS: 25

COURSE NAME: FUNDAMENTALS OF SMART SYSTEMS AND IOT

COURSE INSTRUCTOR: AAYUSH SHARMA

MAX. TIME: 1 Hour 30 Min

Note: (a) All questions are compulsory.

(b) The candidate is allowed to make Suitable numeric assumptions wherever required for solving problems

(c) Use of calculator is allowed

Q.No	Question	CO	Marks
Q1	<p>An LM35 (10 mV/°C) is connected to an Arduino Uno (10-bit ADC, 5. V ref) on pin A0. The code reads: <code>int raw = analogRead(A0);</code> <code>float voltage = raw * (5.0 / 1024.0);</code> <code>float tempC = voltage / 0.01;</code> Serial Monitor shows 24.41°C; a calibrated thermometer reads 25.0°C. The student then moves the same LM35 and same code to an ESP32 (12-bit ADC, 3.3 V ref), changing only the pin name.</p> <p>(a) The code divides by 1024. Calculate the maximum error (in °C) this introduces over 0–100°C. Is this the main cause of the 0.59°C discrepancy at 25°C? Justify numerically.</p> <p>(b) She wants 0.1°C resolution. What minimum ADC bit-depth is needed (5 V ref)? Why would 16-bit ADC still not guarantee reliable 0.1°C detection? Give two reasons.</p>	[CO2]	[4+4]
Q2	<p>A student builds a freeze-alert system on an Arduino Uno. An RGB LED (R=pin 9, G=pin 10, B=pin 11) and a 16×2 LCD are connected. A push button on pin 2 (INPUT_PULLUP) toggles alert mode ON/OFF. When ON, the LED blinks red at 2 Hz (250 ms ON, 250 ms OFF) using <code>delay(250)</code>, and the LCD shows "ALERT ON". When OFF, the LED is steady green and the LCD shows "ALERT OFF". The core loop structure is:</p> <pre>void loop() { if (digitalRead(2) == LOW) alertOn = !alertOn; if (alertOn) { digitalWrite(R, HIGH); delay(250); digitalWrite(R, LOW); delay(250); lcd.clear(); lcd.print("ALERT ON"); } else { digitalWrite(R, LOW); digitalWrite(G, HIGH); lcd.clear(); lcd.print("ALERT OFF"); } }</pre>	[CO3]	[3+3+3]

	<p>The student notices: the button often takes nearly half a second to respond, quick presses are frequently missed entirely, and the LCD text briefly garbles on every blink cycle.</p> <p>(a) The two <code>delay(250)</code> calls block for a combined 500 ms per blink cycle. <code>digitalRead()</code> is checked only once at the top of each loop iteration. The student now wants the button to respond within 50 ms — what is the maximum <code>delay()</code> value usable, and what blink frequency would that produce? Show mathematically that you cannot simultaneously achieve 2 Hz blinking AND <50 ms response using <code>delay()</code>.</p> <p>(b) A classmate suggests: <i>"Replace <code>delay(250)</code> with <code>delay(10)</code> and toggle the LED every 25th iteration using a counter."</i> Button response improves, but now <code>lcd.clear()</code> executes every 10 ms — 100 times per second. The HD44780 LCD controller requires ~2 ms to process <code>clear()</code> internally. Explain why the LCD now shows flickering/garbled text. Propose a fix that keeps the fast loop but makes the display stable.</p> <p>(c) The student finally adopts <code>millis()</code> with independent timers for blinking, button-reading, and LCD-updating. Everything works — except when toggling from alert-ON (blinking red) to alert-OFF (steady green), a brief flash of yellow (RED + GREEN) appears. Explain the exact cause: what is the state of the R and G pins at the instant the mode switches if the red LED happened to be HIGH during a blink-ON phase? Why does writing <code>GREEN=HIGH</code> before <code>RED=LOW</code> produce yellow (RED + GREEN)? Propose a one-line code fix.</p>		
Q3	<p>A building has 50 rooms, each with DHT11 (temp + humidity) and MQ-135 (air quality), publishing to an MQTT broker. Three subscribers: S1 (dashboard — all data, all rooms), S2 (fire safety — only temperature, all rooms), S3 (HVAC for Room 17 only — all three readings).</p> <p>(a) Design a topic hierarchy so each subscriber can use a single wildcard subscription (+, #) to get exactly what it needs. Write: the publish topic for each sensor type, and the subscribe string for S1, S2, and S3.</p> <p>(b) S2 (fire safety) must never miss a reading even if briefly disconnected. S1 (dashboard) only needs the latest value. For each, state the MQTT QoS level and whether retained messages should be used. Justify each choice in one sentence.</p>	[CO3]	[4+4]