Introduction to Internet of Things (IO4041)

Date: March 1st 2025

Course Instructor

Ammar Haider

Sessional-I Exam

Total Time (Hrs): 1
Total Marks: 44
Total Questions: 6

_____ _____ _____ _____

Student Name Roll No Section Student Signature

Instructions:

1. Write all answers on the question booklet. Rough pages may be used but won't be collected.
2. Write in blue or black ink only.
3. If you think some information is missing then make an assumption and state it clearly.

CLO 1	Discuss the basic ecosystem, architecture and key components of an Internet of Things solution.
CLO 2	Create Arduino sketches that involve analog & digital communication with sensors and actuators.
CLO 3	Determine appropriate higher layer protocols as well as lower layer wireless connectivity technologies for a given IoT application.
CLO 4	Design and develop a prototype IoT solution that includes electronic circuits for collecting environmental data, cloud-hosted analytics and processing modules, and frontend software for user interfacing.

CLO 1

Q1: MCQs **[6 marks]**

Important Note: More than one option MIGHT BE correct.
To get credit, you need to choose all of the correct options, and no incorrect ones.

1. A device that converts chemical energy to electrical energy would be called _____.

   A. sensor B. actuator C. transducer D. calibrator

2. If an Analog-to-Digital converter (ADC) output width is 8 bits, how many different values can it produce on the output?

   A. 8 B. 64 C. 128 D. 256

3. Compared to client-server architecture, pub-sub architecture provides _____.

   A. better scalability B. privacy

   C. higher bandwidth D. guaranteed delivery

4. Select MQTT topic(s) that is invalid for publishing data.

   A. `/fleet/trucks/t26/status`

   B. `farm/eastZone/نمی/مٹی`

   C. `farm/north zone/green house/temperature`

   D. `wearable/+/steps`

5. MQTT-SN always runs on top of UDP transport.

   A. True B. False

6. Arduino function `analogWrite()` creates a(n) _____ signal on the specified pin.

   A. analog B. digital

CLO 1

Q2: Short answer questions **[3 + 3 + 2 + 3 marks]**

Imagine a real world IoT application from any one of the following domains: (i) smart mobility (ii) smart health. Give application details, including type of sensors used and how the application helps improve the quality of life.

Open ended

An illumination sensor has negative bias of 0.2 lux, and stated inaccuracy of up to 0.15 lux. Vendor also reports that readings are subject to noise of up to 0.03 lux. If real illumination value is 320 lux, find the set of possible values reported by the sensor.

Adjusting for bias only, sensor would report 319.8 lux. Factor in the inaccuracy, possible range becomes [319.65, 319.95]. Effect of noise will make the possible range [319.62, 319.98] lu

List any three differences between microprocessor and microcontroller.

Compared to microprocessors, microcontrollers consume less power, operate on slower clock speeds, have a smaller instruction set, require less external circuitry, have fixed on-chip RAM, ROM and I/O ports

Analyze and discuss the statement as true or false: “Program memory in a microcontroller is both readable and writable”.

Program memory is only readable when µC is running. To change the contents of program memory, special electrical signals need to be applied and a program is then transferred (burnt/flashed) into it.

CLO 2

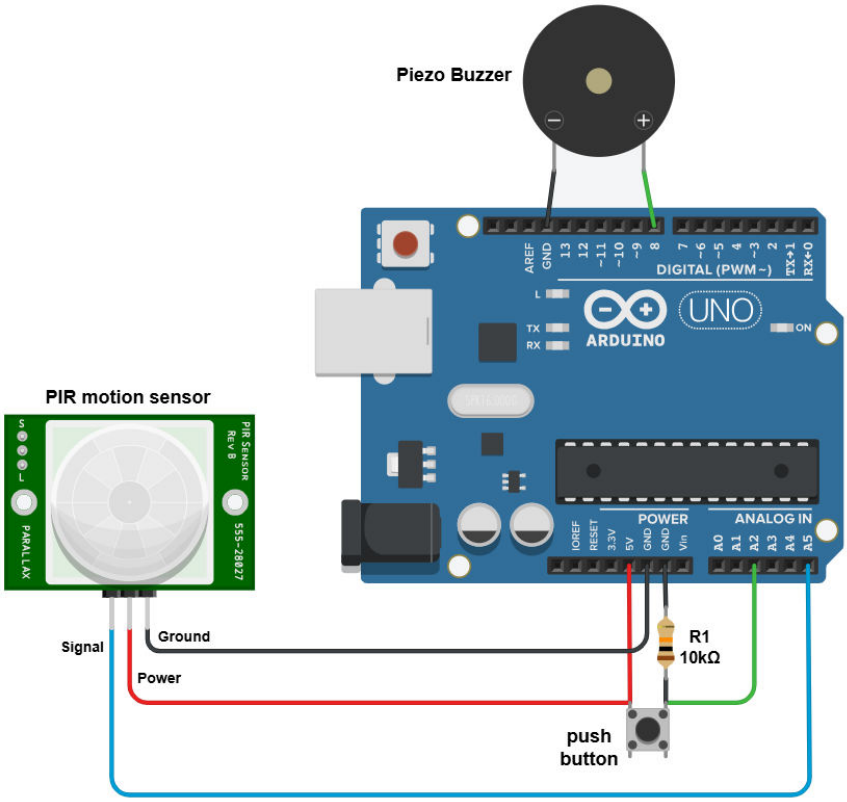
Q3: Arduino Programming **[2 + 9 marks]**

Piezo buzzers are small sound speakers that convert an electrical signal of specific frequency into a sound signal of the same frequency (called a tone).

Suppose we are creating an Arduino smart-home application with a piezo buzzer that performs two functions. It produces a small melody (sequence of tones) when someone rings the doorbell. Secondly it makes a beeping alert when an intruder is detected by the motion sensor installed at the back side of the house.

To create a signal of specific frequency, Arduino library includes the function **`tone(pin, frequency, duration)`** where `pin` is the Arduino pin number, `frequency` is any positive integer, and `duration` is in milliseconds.

An Ardunio circuit is shown below, it is designed to meet the above requirements. The push button acts as the doorbell.



An example melody can be created by the following sequence of frequencies played for the indicated durations, including a pause of 100ms in between. That is, you should call the tone() function to create a note of 262Hz for 250ms, then wait for 100ms. Then call tone() again to create a note of 196Hz for 125ms, then a pause of 100ms, and so on.

Frequency (Hz)	262	196	196	247	196	0	247	262	0
Duration (ms)	250	125	125	250	250	250	250	250	250

To create the alert sound, you should send a signal of 400Hz for half second, followed by a silence of half second. Repeat this process three times, so that alert will sound like (beep, silence, beep, silence, beep, silence).

Q3A. In the circuit, why do we have a resistor R1 connected to push button?

It is a pull down resistor. In the absence of this resistor, the pin will be floating (undefined) when button is not pressed. Resistor forces the pin to be logic 0 (ground).

Q3B. Write the full Arduino sketch in the space provided. Remember, the analog pins A0-A5 can also be used for digital I/O. Assume that PIR sensor behaves like a simple digital sensor – a HIGH output on sensor means motion is detected. Do not worry about its calibration.

```
// create global variables if needed
int freqs[] = {262, 196, 196, 247, 196, 0, 247, 262, 0};
int durations[] = {250, 125, 125, 250, 250, 250, 250, 250, 250};
int bellPressed = LOW;
int buzzer = 8, pushButton = A2, pir = A5;
int motion = 0;

void melody() {  // create the melody sound
  for (int i = 0; i < 9; i++) {
    tone(buzzer, freqs[i], durations[i]);
    delay(durations[i]+100); // tone() call is non blocking
  }
}

void alert() {  // create the alert sound
  for (int i = 0; i < 3; i++) {
    tone(buzzer, 400, 500);
    delay(1000);
  }
}

void setup() { // one-time setup code goes here
  pinMode(buzzer, OUTPUT);
  pinMode(pushButton, INPUT);  // optional, since INPUT mode is default
  pinMode(pir, INPUT);
}
```

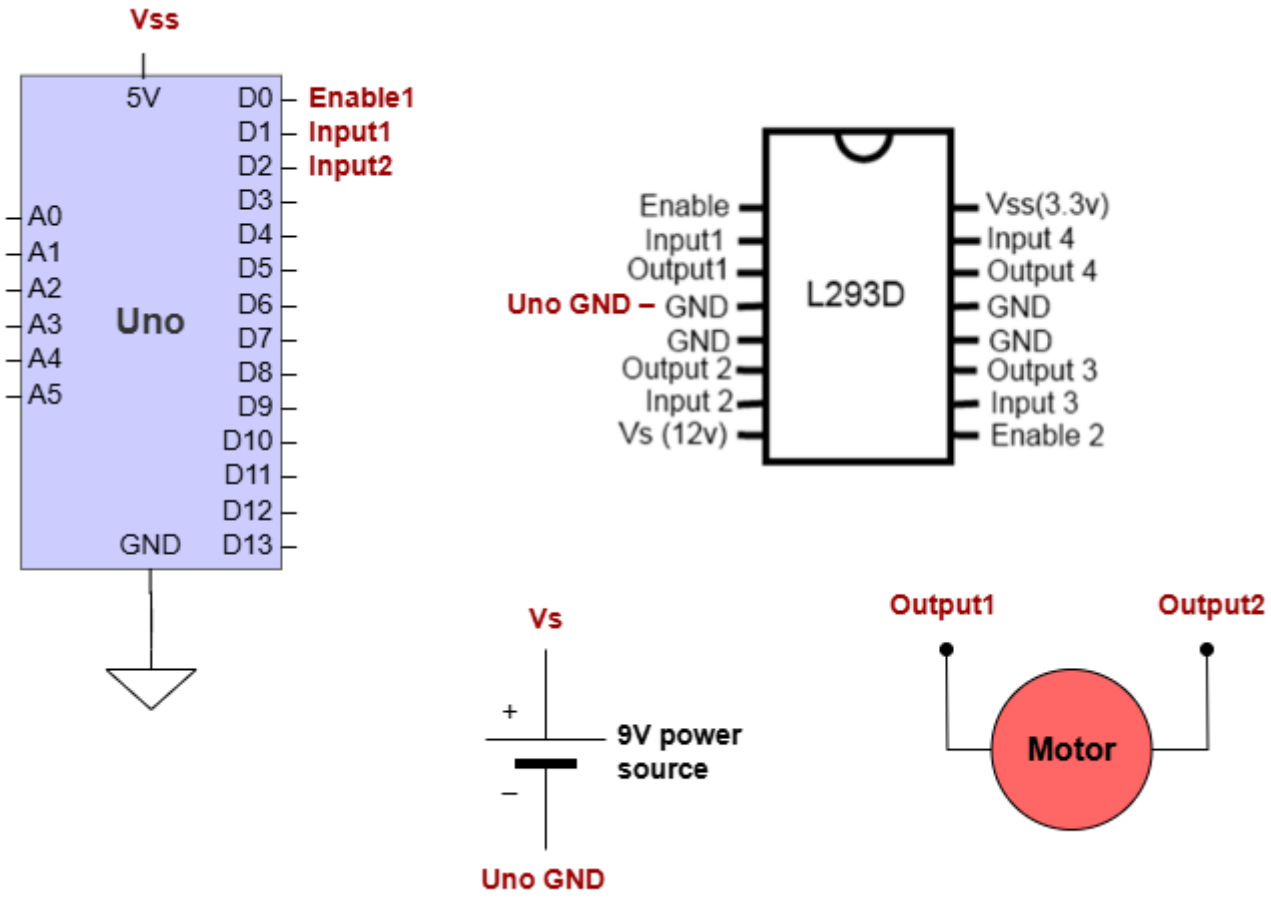
```
void loop() {
  // create the alert sound if motion is being detected
  // otherwise, react to doorbell with melody sound
  bellPressed = digitalRead(pushButton);
  motion = digitalRead(pir);
  if (motion)
      alert();
  else if (bellPressed)
    melody();

  delay(100);
}
```

CLO 2

Q4: Arduino circuits **[4 marks]**

Complete the circuit diagram below by drawing the connection-wires. Goal of the circuit is to be able to control the motor speed and rotation direction by Arduino code.

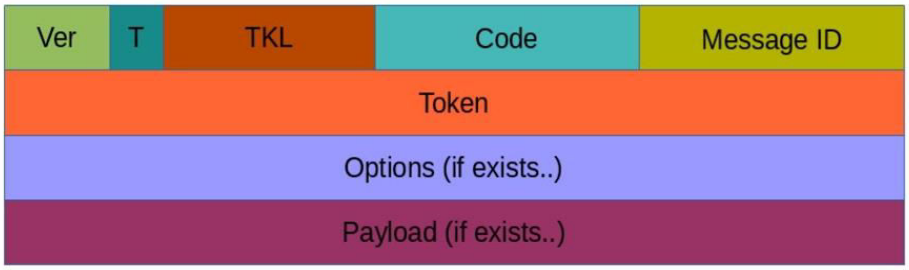


Notes: Any three digital pins on Uno can be used. For L293D, the set of motor2 pins can also be used (enable2, input3,4 output3,4)

CLO 3

Q5: CoAP protocol **[3 + 3 marks]**

Given below is the format of CoAP messages. What is the purpose of Message ID and Token fields?



MessageID matches a CON message with its corresponding ACK or RST message. This field is used by messaging sublayer.

Token is used to associate a response with a request. This value is used by req/res sublayer.

A motion detector implements a CoAP server. How can a client app stay updated with sensor readings without continuous requests?

Client should use the observation feature of CoAP, by sending a GET request with Observe option. Then whenever reading changes, sensor will send a response message, containing new value, to client.

CLO 3

Q6: MQTT protocol **[6 marks]**

A factory staff uses MQTT to monitor the state of all machinery. Below is an incomplete diagram showing messages sent by two clients to the broker. Complete the diagram by drawing all messages sent by broker to clients in reaction to client commands.

For data messages (PUBLISH), make sure to indicate the QoS level.