Seat No.: _____ Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY

BE - SEMESTER–VII (NEW) EXAMINATION – WINTER 2021

Subject Code:3171108

Date:23/12/20

21 Subject Name: Internet of things

Time: 10:30 AM TO 01:00 PM **Total Marks: 70**

MARKS

Q.1 **(a)** Define IOT. Also few Applications of IOT. **03**

IoT stands for Internet of Things. It is a network of physical devices, vehicles, buildings, and other items embedded with electronics, software, sensors, and connectivity which enables these objects to connect and exchange data.

Applications of IoT include:

1. Smart Home: Control lights, temperature, and security remotely using a smartphone app.
2. Industrial IoT: Monitor and control industrial equipment and machinery to improve efficiency and productivity.
3. Connected Cars: Monitor vehicle performance and provide driver assistance.
4. Healthcare: Monitor patients remotely and improve their treatment outcomes.
5. Smart City: Collect and analyze data from various sensors to improve city services and infrastructure.
6. Supply Chain and Logistics: Track inventory and shipments in real-time to improve efficiency and reduce costs.

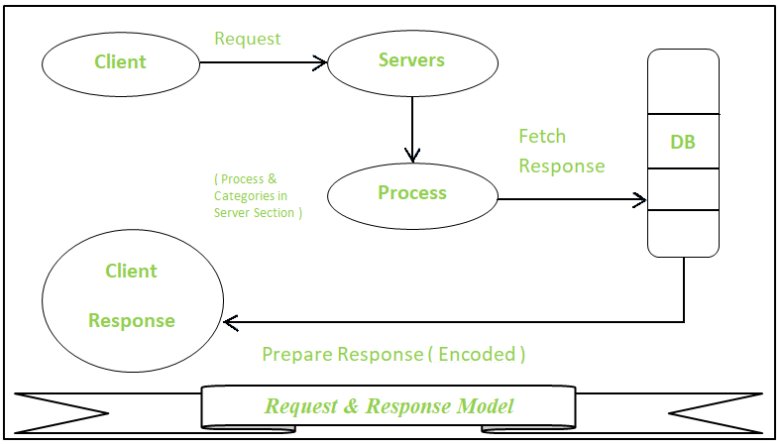
(b) Write Various characteristics of IOT. **04**

1. Connectivity: IoT devices are connected to the internet and can communicate with each other.
2. Sensors: IoT devices are equipped with sensors that collect data and send it to other devices or systems.
3. Intelligence: IoT devices are capable of making decisions and performing actions based on the data they collect.
4. Automation: IoT devices can automate tasks and processes, reducing the need for human intervention.
5. Scalability: IoT can include a large number of devices and can be easily expanded.
6. Remote Access: IoT devices can be accessed and controlled remotely via a smartphone or computer.
7. Real-time data: IoT devices can collect and transmit data in real-time, allowing for immediate action to be taken.
8. Interoperability: IoT devices can connect and communicate with different types of devices and systems.

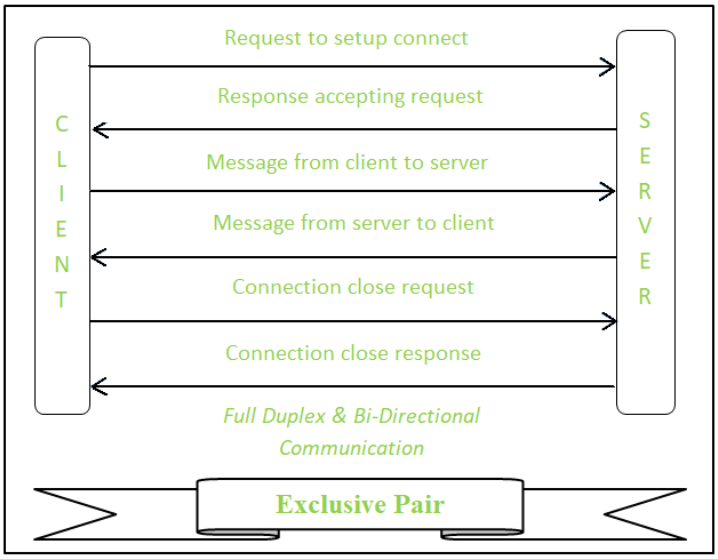
9. Security: IoT devices need to be secure to protect the data they collect and transmit.
10. Adaptability: IoT systems can be adapted to changing requirements and environments.

(c) Write about Request – Response and Exclusive Pair Communication model of IOT. **07**

The Request-Response model is a common communication pattern in IoT, where a device (the client) sends a request to another device (the server) and the server responds with the requested information. This pattern is used in many applications, such as when a smart thermostat requests the temperature from a sensor or when a smartphone app requests the status of a connected light bulb.



Exclusive Pair Communication model is a communication model where two devices are exclusively paired with each other and can only communicate with each other. This model is particularly useful when two devices need to exchange sensitive information and there is a need to ensure that the information is only exchanged between the two devices and no one else can access it. An example of this model is when a smartwatch is paired with a smartphone to exchange sensitive health data.



Both the Request-Response and Exclusive Pair Communication models are widely used in IoT to enable communication between devices and enable the collection, analysis and transmission of data.

Q.2 (a) Describe Bluetooth and BLE. **03**

Bluetooth is a wireless communication technology that enables devices to communicate with each other over short distances. It uses a 2.4 GHz radio frequency to transmit data and can be used to connect devices such as smartphones, laptops, speakers, and headphones. Bluetooth devices can be paired with each other to establish a connection, and once connected, they can exchange data. Bluetooth technology is widely used in IoT applications such as home automation, wearables, and smart home devices.

BLE (Bluetooth Low Energy) is a version of Bluetooth technology that is designed for low-power devices and applications. BLE uses a lower amount of power compared to classic Bluetooth and is ideal for devices that need to run on batteries for long periods of time. BLE is also designed to consume less power when it is in standby mode, making it a good choice for IoT devices that need to be always on and ready to communicate. BLE is commonly used in applications such as fitness trackers, smartwatches, and other wearable devices.

BLE is different from classic Bluetooth in that it has a lower data rate and is optimized for low power consumption. This makes it suitable for IoT devices that need to be always on and ready to communicate, but not necessarily transfer large amounts of data.

(b) What is REST? Write various methods of REST. **04**

REST (Representational State Transfer) is an architectural style for building web services. RESTful web services use HTTP requests to POST (create), PUT (update), GET (read), and DELETE data. A RESTful web service typically defines a URI (Uniform Resource Identifier), which is a service endpoint and a set of HTTP methods that operate on the resource identified by the URI.

The main methods of REST are:

1. GET: Retrieves information about a resource. It is used to retrieve data from the server.
2. POST: Creates a new resource. It is used to send data to the server to create a new resource.
3. PUT: Updates an existing resource. It is used to send data to the server to update an existing resource.
4. DELETE: Deletes a resource. It is used to delete a resource from the server.
5. PATCH: partially updates a resource. It is used to send data to the server to partially update an existing resource.

These methods are also referred to as CRUD (Create, Read, Update and Delete) operations, which are used to perform actions on resources. RESTful web services use these methods to interact with a resource, and the resource is identified by a URI.

(c) Write about IOT Level – 5 and IOT Level – 6. **07**

IoT Level 5 refers to fully autonomous systems, where the IoT devices are able to make decisions and take actions without human intervention. These systems use advanced technologies such as machine learning and artificial intelligence to analyze data and make decisions. Examples of Level 5 IoT systems include self-driving cars, drones, and robots. **07**

IoT Level 6 refers to systems that are integrated into the physical environment and can interact with it in a natural way. These systems use technologies such as augmented reality, virtual reality, and haptic feedback to enhance the user's experience. Examples of Level 6 IoT systems include smart homes, smart cities, and virtual/augmented reality-based applications.

At the highest level, Level 6 IoT systems are able to learn from the environment and adapt to the user's preferences and habits. They allow for the seamless integration of technology into the physical world, creating an immersive and intuitive experience for the user.

It's worth mentioning that these levels are not an official classification and may vary across different sources, and also these levels are not absolute, as many of the IoT systems can have a mix of characteristics of different levels.

OR

What is NFV(Network Function Virtualization)? And write key elements of NFV.

Network Function Virtualization (NFV) is a technology that enables the virtualization of network functions that were previously provided by dedicated hardware. Instead of using physical network devices, NFV allows these functions to be implemented in software and run on commercial off-the-shelf (COTS) servers.

The key elements of NFV include:

1. Virtualized Network Functions (VNFs): These are the software implementations of network functions such as firewalls, routers, and load balancers.
2. Virtualized Infrastructure Manager (VIM): This is the management layer that controls the virtualized resources, such as servers and storage, that are used to run VNFs.
3. NFV Orchestration (NFVO): This is the management and orchestration layer that coordinates the deployment and management of VNFs and other network functions.
4. Virtual Network Functions Management and Orchestration (VNFM/O): This is the management layer that is responsible for the lifecycle management of VNFs.

5. Hardware Abstraction Layer (HAL): This is the layer that abstracts the physical resources, such as servers and storage, from the VNFs.

NFV enables network operators to be more agile and flexible in deploying and managing network services, as well as reduces costs by using commodity hardware instead of dedicated network devices. It also allows for the dynamic scaling of network resources, making it easier to handle fluctuations in traffic and user demand.

Q.3 (a) Write various Security concerns dealing with IOT. **03**

IoT security is a major concern as it involves a wide range of devices that collect, transmit, and process sensitive information. Some of the security concerns with IoT include:

1. Device security: IoT devices are often vulnerable to hacking and malware attacks, which can compromise the device's security and expose sensitive information.
2. Network security: IoT devices communicate over a network, and if the network is not secure, it can be vulnerable to attacks such as man-in-the-middle (MitM) and eavesdropping.
3. Data security: IoT devices collect and transmit sensitive information, and if this data is not properly secured, it can be intercepted and used for malicious purposes.
4. Privacy: IoT devices can collect and transmit personal information, and if this information is not properly protected, it can be used to track and profile users.
5. Interoperability: IoT devices are often built by different manufacturers and may use different protocols, which can make it difficult to ensure that they are secure.
6. Lack of security standards: There is a lack of security standards for IoT devices, which makes it difficult to ensure that devices are secure.
7. Distributed Denial of Service (DDoS) attacks: IoT devices can be used as part of a botnet to launch DDoS attacks on other systems.

(b) What is M2M? Describe it with few examples. **04**

M2M stands for Machine-to-Machine communication, it refers to the communication between devices and systems without human intervention. These devices can be computers, smartphones, sensors, machines, vehicles, and other equipment that are connected to the internet and can communicate with each other.

Examples of M2M include:

1. Smart grid: Smart meters and other devices in the power grid communicate with each other to manage energy consumption and optimize the distribution of electricity.
2. Remote monitoring: Sensors and cameras can be used to remotely monitor industrial equipment and machinery, allowing for early detection of problems and reducing downtime.

3. Automated inventory management: RFID tags and sensors can be used to track inventory and automatically reorder products when stock is low.
4. Telematics: GPS and other sensors can be used to track vehicles and monitor their performance, which can improve fleet management and reduce costs.
5. Smart cities: Sensors and cameras can be used to collect data on traffic, air quality, and other environmental factors, which can be used to improve city services and infrastructure.

M2M communication enables devices to communicate with each other, share data, and make decisions automatically. This allows for more efficient and effective operations, improved decision making, and the ability to monitor and control devices remotely.

(c) Discuss Difference between IOT and M2M. **07**

IoT (Internet of Things)	M2M (Machine-to-Machine)
IoT refers to the network of physical devices, vehicles, buildings, and other items embedded with electronics, software, sensors, and connectivity which enables these objects to connect and exchange data.	M2M refers to the communication between devices and systems without human intervention. These devices can be computers, smartphones, sensors, machines, vehicles, and other equipment that are connected to the internet and can communicate with each other.
IoT devices are connected to the internet and can communicate with each other.	M2M devices are connected to each other and can communicate with each other.
IoT includes a wide range of devices, such as smartphones, laptops, home appliances, and industrial equipment.	M2M mainly focuses on industrial and commercial applications, such as manufacturing, transportation, and logistics.
IoT is designed to improve the efficiency and effectiveness of daily life.	M2M is designed to improve the efficiency and effectiveness of industrial and commercial operations.

03

OR

Q.3 **(a)** Write functionality of XMPP.

XMPP (Extensible Messaging and Presence Protocol) is an open-source communication protocol used for real-time messaging and presence information. It is designed for instant messaging, online gaming, and other real-time applications.

The main functionality of XMPP includes:

1. Instant Messaging: XMPP allows users to send and receive instant messages in real-time. It supports one-to-one and multi-user chats, and can also be used for group chats and conference calls.

2. Presence Information: XMPP allows users to share their status (e.g. online, offline, away) and other presence information with their contacts.
3. Roster Management: XMPP allows users to manage their contacts, also known as "roster" in XMPP parlance, and see the presence information of their contacts.
4. Extensibility: XMPP is designed to be extensible, which means that new features and functionality can be added through the use of "XMPP Extensions".
5. Security: XMPP supports encryption and authentication for secure communication.
6. Publish-Subscribe: XMPP allows for publish-subscribe pattern which means that the client can subscribe to certain topics and receive updates on those topics.
7. Interoperability: XMPP allows communication between different clients and servers, regardless of their platform or vendor.
8. Server-to-Server Communication: XMPP allows communication between different servers, this enables users to communicate across different domains.

XMPP is widely used in IoT and M2M communication, it allows devices to communicate with each other, share data and make decisions automatically, this is particularly useful for IoT applications such as home automation, remote monitoring, and smart cities.

(b) Write an arduino code for connecting NodeMCU with temperature and Humidity sensor. And display current temperature and humidity with 5 second delay. **07**

Here is an example of Arduino code for connecting a NodeMCU board to a DHT11 temperature and humidity sensor and displaying the current temperature and humidity with a 5-second delay:

```
#include <Adafruit_Sensor.h>

#include <DHT.h>

#include <DHT_U.h>


#define DHTPIN 2     // Pin where DHT sensor is connected

#define DHTTYPE DHT11   // DHT 11


DHT dht(DHTPIN, DHTTYPE);


void setup() {

  Serial.begin(115200);

  dht.begin();
```

```
}

void loop() {
  delay(5000);
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print("%  Temperature: ");
  Serial.print(t);
  Serial.println("°C ");
}
```

In this code, we first include the necessary libraries: Adafruit_Sensor, DHT, and DHT_U. Then, we define the pin and type of the DHT11 sensor. In the setup function, we start the serial communication and initialize the DHT sensor. In the loop function, we use the delay function to wait for 5 seconds, then we use the dht.readHumidity() and dht.readTemperature() functions to read the current humidity and temperature values from the sensor. The humidity and temperature values are stored in variables h and t. We then use the serial.print() function to display the humidity and temperature values on the serial monitor.

Note: Make sure you have the DHT library installed in your Arduino IDE before uploading the code.

Also, ensure that the NodeMCU board is properly connected to the DHT11 sensor and that the correct pin number is used in the code.

(c) Explain SDN(Software Define Networking). **04**

Software-Defined Networking (SDN) is an approach to networking in which the control plane of network devices is separated from the data plane. The control plane is responsible for making decisions about how data is forwarded and the data plane is responsible for forwarding the data.

In traditional networking, the control plane and data plane are tightly coupled and are implemented in the same device. This makes it difficult to make changes to the network and to add new features. With SDN, the control plane is implemented in software and can run on a separate device, such as a server or a virtual machine. This separation allows for more flexibility and easier management of the network.

The key elements of SDN include:

1. The SDN Controller: This is the control plane of the network and is responsible for making decisions about how data is forwarded. The controller can be a physical or virtual device and can run a variety of software.
2. The Network Device: This is the data plane of the network and is responsible for forwarding data. The device can be a switch, router, or other network equipment.
3. The SDN Application: This is the application that runs on the controller and provides network services such as routing, load balancing, and security.
4. The SDN API: This is the interface between the controller and the network devices and allows for the controller to send instructions to the devices.

SDN can be used to improve the scalability, security, and manageability of the network. It allows for the network to be programmed and controlled in a more efficient way, which can lead to faster deployment of new services and reduced operational costs. It also allows for more flexibility in network design and can make it easier

Q.4 **(a)** Write Names of various protocols that are used at Network Layer of IOT. **03**

At the Network Layer of the IoT, a variety of protocols are used to enable communication between devices. Some of the most commonly used protocols at this layer include:

1. Internet Protocol (IP): This is the most widely used protocol at the Network Layer and is responsible for routing data packets between devices. IP can be used in both IPv4 and IPv6 versions.
2. Internet Control Message Protocol (ICMP): This protocol is used to send error messages and operational information about the

network.
3. Address Resolution Protocol (ARP): This protocol is used to map a network address to a physical address.
4. Routing Information Protocol (RIP): This is a distance-vector routing protocol used to distribute routing information within a network.
5. Open Shortest Path First (OSPF): This is a link-state routing protocol used to distribute routing information within a network.
6. Border Gateway Protocol (BGP): This is a path-vector routing protocol used to distribute routing information between different autonomous systems.
7. Multicast: This protocol is used to send data to multiple devices at the same time.
8. 6LoWPAN: This protocol is used to enable communication between IPv6 devices over low-power wireless networks.
9. CoAP: This is a protocol designed specifically for IoT devices and is used for resource-constrained devices and low-power networks.
10. MQTT: This is a publish-subscribe protocol that is designed for machine-to-machine (M2M) and IoT applications.

(b) Write Names of Protocols that are used at Link Layer of IOT. **04**

At the Link Layer of the IoT, a variety of protocols are used to enable communication between devices on the same network segment. Some of the most commonly used protocols at this layer include:

1. Media Access Control (MAC) protocol: This protocol is responsible for controlling access to the shared medium, such as a wireless channel or an Ethernet cable, and for addressing devices at the link layer.
2. Bluetooth: This is a wireless technology that is widely used in IoT devices for short-range communications.
3. Zigbee: This is a wireless protocol that is designed for low-power, low-data-rate communications and is often used in IoT applications.
4. Z-Wave: This is another wireless protocol that is designed for low-power, low-data-rate communications and is often used in IoT applications.
5. LoRaWAN: This is a long-range wireless protocol that is designed for low-power, low-data-rate communications and is often used in IoT applications.
6. Wi-Fi: This is a wireless protocol that is widely used in IoT devices for high-speed data communications.
7. Ethernet: This is a wired protocol that is widely used in IoT devices for high-speed data communications.
8. Thread: This is a protocol designed for IoT devices, which is based on IPv6 and 6LoWPAN, it's designed for creating low-power, secure, and reliable networks for connected devices in the home and building automation.
9. Zigbee 3.0: This is a new version of Zigbee protocol which is based on the Zigbee Pro and Smart Energy profile, it's designed for creating low-power, secure, and reliable networks for connected devices.

These protocols are suitable for different types of IoT applications and different types of networks, depending on the requirements of range, power consumption, data rate, and security.

(c) Write various Static Characteristics of Sensors. **07**

Static characteristics of sensors refer to the performance of a sensor when it is not changing or in a steady state. Some of the key static characteristics of sensors include:

1. Sensitivity: This refers to the ratio of the output of the sensor to the input. It is typically measured in units of output per unit of input.
2. Linearity: This refers to the degree to which the output of the sensor is proportional to the input. A sensor with high linearity will have a small deviation from a straight line when plotted on a graph.
3. Hysteresis: This refers to the difference in the output of the sensor when the input is increased or decreased. A sensor with low hysteresis will have a small difference in output when the input is increased or decreased.
4. Repeatability: This refers to the degree to which the sensor produces the same output for the same input. A sensor with high repeatability will produce the same output for the same input multiple times.
5. Accuracy: This refers to the degree to which the sensor's output is close to the true value of the input. A sensor with high accuracy will produce an output that is close to the true value of the input.
6. Resolution: This refers to the smallest change in the input that the sensor can detect. A sensor with high resolution will be able to detect small changes in the input.
7. Drift: This refers to the gradual change in the output of the sensor over time. A sensor with low drift will have a small change in the output over time.
8. Offset: This refers to the output of the sensor when the input is zero. A sensor with low offset will have a small output when the input is zero.
9. Noise: This refers to the unwanted variations in the output of the sensor that are not due to the input. A sensor with low noise will have a small unwanted variations in the output.
10. Range: This refers to the range of input that the sensor can detect. A sensor with a wide range will be able to detect a wide range of input.

OR

Q.4 (a) Define Sensors, Actuators, Transducers. **03**

A sensor is a device that detects changes in the environment and converts them into an electrical signal or other form of output. Sensors can measure a wide range of physical phenomena such as temperature, light, humidity, pressure, motion, and more. They are used in a variety of

applications such as industrial, automotive, and consumer electronics.

An actuator is a device that converts an electrical signal or other form of input into a physical action. Actuators can be used to control a wide range of physical phenomena such as movement, position, temperature, pressure, and more. Examples of actuators include motors, solenoids, and pneumatic and hydraulic cylinders.

A transducer is a device that converts one form of energy into another. Transducers can be used to convert physical phenomena such as temperature, light, sound, pressure, and motion into an electrical signal or other form of output. They can also be used to convert electrical signals into physical phenomena such as movement, position, and sound. Sensors and actuators are types of transducers, they convert physical quantities into electrical signals (sensors) or electrical signals into physical actions (actuators) respectively.

(b) Write about any 4 sensors that you know. And their usages. **04**

1. Temperature Sensor: A temperature sensor is a device that measures the temperature of an object or environment. They are used in a wide range of applications such as HVAC systems, automotive, industrial processes, and consumer electronics. Examples of temperature sensors include thermistors, thermocouples, and RTDs (Resistance Temperature Detectors).
2. Light Sensor: A light sensor is a device that measures the intensity of light. They are used in a wide range of applications such as cameras, smartphones, and industrial automation. Examples of light sensors include photodiodes, phototransistors, and CCD (charge-coupled device) sensors.
3. Proximity Sensor: A proximity sensor is a device that detects the presence of an object without physical contact. They are used in a wide range of applications such as smartphones, industrial automation, and automobiles. Examples of proximity sensors include infrared sensors, ultrasound sensors, and capacitive sensors.
4. Accelerometer: An accelerometer is a device that measures the acceleration and tilt of an object. They are used in a wide range of applications such as smartphones, automobiles, and industrial automation. Examples of accelerometers include MEMS (microelectromechanical systems) accelerometers and piezoelectric accelerometers.

(c) Write an arduino code for connecting NodeMCU with moisture sensor, servo motor and water pipe lines for automatic watering a plant when moisture level is decreasing. **07**

Here is an example of Arduino code for connecting a NodeMCU board to a moisture sensor, a servo motor, and water pipe lines for automatic watering of a plant when the moisture level is decreasing:

```
#include <Servo.h>

int moistureSensor = A0;
int moistureValue = 0;
int thresholdValue = 700;
Servo myservo;

void setup() {
  myservo.attach(D5);
  Serial.begin(115200);
}

void loop() {
  moistureValue = analogRead(moistureSensor);
  Serial.print("Moisture Value: ");
  Serial.println(moistureValue);

  if (moistureValue < thresholdValue) {
    Serial.println("Watering the plant...");
    myservo.write(90);
    delay(1000);
    myservo.write(0);
  }

  delay(1000);
}
```

In this code, we first include the necessary library Servo.h. Then, we define the pin of the moisture sensor, servo motor and the threshold value of moisture level, below which the plant needs to be watered. In the setup function, we attach the servo to pin D5 and start the serial communication. In the loop function, we use the analogRead function to read the moisture level from the sensor, and the value is stored in the variable moistureValue. We then use the serial.print() function to display the moisture level on the serial monitor.

We then use an if-else statement to check whether the moisture level is less than the threshold value. If it is, the servo motor will rotate to open the water

Q.5(a) Write about LoraWan. **03**

LoRaWAN (Long Range Wide Area Network) is a low-power wide area network (LPWAN) protocol that is designed for IoT applications. It operates in the unlicensed ISM (Industrial, Scientific, and Medical) band and uses a chirp spread spectrum modulation technique to achieve long-range communication.

One of the key features of LoRaWAN is its ability to support a large number of devices while maintaining low power consumption. This makes it well-suited for applications such as smart metering, asset tracking, and environmental monitoring.

The architecture of a LoRaWAN network consists of three main components:

1. End devices: These are the devices that are connected to the network and typically have limited power and processing capabilities. Examples of end devices include sensors, actuators, and other IoT devices.
2. Gateways: These are the devices that bridge the end devices to the network and are responsible for forwarding data between the end devices and the network server.
3. Network server: This is the central point of the network that is responsible for managing the communication between the end devices and gateways.

LoRaWAN uses a star-of-stars topology in which gateways forward data to the network server, which then routes the data to the appropriate end device. This allows for a scalable network that can support a large number of devices.

Security is an important aspect of LoRaWAN, it uses AES-128 encryption to secure the communication between devices and the network, also it uses a secure join process to authenticate devices to the network.

(b) Differentiate ESP8266 and ESP32. **04**

ESP8266	ESP32
CPU: 32-bit L106 RISC microprocessor	CPU: 32-bit LX6 microprocessor
Clock Speed: 80 MHz	Clock Speed: up to 240 MHz
Memory: 32-64 kB RAM	Memory: 520 kB SRAM
Communication: WiFi	Communication: WiFi, Bluetooth
Power Consumption: 80mA	Power Consumption: 150mA
Pin Count: 17	Pin Count: 36
ADC Resolution: 10 bit	ADC Resolution: 12 bit
PWM: 2	PWM: 16
UART: 1	UART: 2
I2C: 1	I2C: 2

- The ESP32 has a more powerful processor and more memory than the ESP8266, which makes it better suited for more complex

projects.
- The ESP32 also has built-in Bluetooth support, which the ESP8266 does not have.
- The ESP32 also has a higher power consumption than the ESP8266, so it may require a larger power supply or a battery with a larger capacity.
- The ESP32 has more number of pins, more number of PWM and UART than ESP8266, which can be useful for projects that require more I/O options.

In summary, the ESP8266 is a good choice for simple projects that only require WiFi connectivity, while the ESP32 is a more powerful option for more complex projects that require WiFi and Bluetooth connectivity, and more number of I/O options.

(c) Write about Encapsulation Protocol 6LowPan. **07**

6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks) is a communication protocol that enables the transmission of IPv6 packets over low-power wireless networks such as Zigbee, Z-Wave, and other similar technologies. The goal of 6LoWPAN is to enable low-power, low-data-rate devices to communicate using the same Internet Protocol as the rest of the Internet.

The 6LoWPAN protocol encapsulates IPv6 packets into smaller frames that can be transmitted over the low-power wireless network. The 6LoWPAN frames include a header that contains information about the packet, such as the destination address, and a payload that contains the actual IPv6 packet.

The 6LoWPAN protocol also includes several features that are designed to optimize the transmission of IPv6 packets over low-power wireless networks. Some of these features include:

- Header compression: 6LoWPAN uses a header compression mechanism to reduce the size of the IPv6 header, which reduces the amount of data that needs to be transmitted over the wireless network.
- Fragmentation: 6LoWPAN supports fragmentation of IPv6 packets, which allows larger packets to be broken down into smaller fragments that can be transmitted over the wireless network.
- Address management: 6LoWPAN includes mechanisms for managing addresses, such as stateless address autoconfiguration, which allows devices to automatically configure their IPv6 addresses.
- Security: 6LoWPAN includes security mechanisms such as link-layer security and end-to-end security to protect the communication of the devices over the low-power wireless network.

6LoWPAN is widely used in IoT applications and it is supported by many IoT platforms such as Contiki, OpenWSN, and RIOT. It also supports

different link layer technologies such as IEEE 802.15.4, Zigbee, and Z-Wave, which makes it a versatile protocol for different wireless networks.

OR

Q.5 (a) Briefly write about CARP protocol. **03**

CARP (Common Address Redundancy Protocol) is a protocol that is used to implement redundancy in a network. It allows multiple hosts to share a virtual IP address, so that if one host fails, another host can take over the virtual IP address and continue to provide services without interruption.

The CARP protocol works by electing a master host and one or more backup hosts. The master host is responsible for responding to requests for the virtual IP address, while the backup hosts monitor the master host and take over the virtual IP address if the master host fails.

The CARP protocol uses a multicast address to send advertisements that contain the virtual IP address, the host's MAC address, and a priority value. The host with the highest priority value becomes the master host, and the others become backup hosts.

One of the key benefits of CARP is that it allows for automatic failover, which means that the virtual IP address can be quickly and seamlessly transferred to a backup host in the event of a failure. This helps to ensure that services are always available, even in the event of a failure.

CARP is typically used in conjunction with other redundancy protocols such as VRRP (Virtual Router Redundancy Protocol) and HSRP (Hot Standby Router Protocol) to provide redundancy at different layers of the network stack.

(b) Discuss Zigbee. **04**

Zigbee is a communication protocol that is used to connect devices in a low-power, low-data-rate wireless personal area network (WPAN). It is an open standard that is developed and maintained by the Zigbee Alliance, an organization of companies that develop and promote Zigbee technology.

The Zigbee protocol is based on the IEEE 802.15.4 standard, which defines the physical and media access control (MAC) layers of the protocol. Zigbee extends this standard by adding a network layer and application layer that provide additional functionality and services.

One of the key features of Zigbee is its low power consumption, which makes it well-suited for devices that run on batteries or have limited power resources. Zigbee devices can operate for several years on a single

battery, making it a popular choice for IoT applications such as home automation, building automation, and smart metering.

Another important feature of Zigbee is its flexibility, it allows the creation of a mesh network, which means that devices can communicate with each other even if they are not in direct range of a central hub or gateway. This enables devices to communicate over a large area and allows for a scalable network.

Zigbee also provides a wide range of application profiles, which are pre-defined sets of commands and procedures that enable devices to communicate with each other and interact with other devices. Some examples of application profiles include the Zigbee Home Automation profile, the Zigbee Smart Energy profile, and the Zigbee Light Link profile.

In summary, Zigbee is a communication protocol that enables low-power, low-data-rate wireless communication between devices in a personal area network. It is an open standard that is widely used for IoT applications and provides low power consumption, flexibility and a wide range of application profiles.

(c) Discuss MQTT protocol in Detail. **07**

MQTT (Message Queuing Telemetry Transport) is a lightweight publish-subscribe messaging protocol that is designed to be used in low-bandwidth, high-latency, or unreliable networks. It is often used in IoT applications, where devices need to send and receive data in real-time, even in challenging network conditions.

The MQTT protocol is based on a publish-subscribe model, in which clients (devices or applications) connect to a central broker, and publish and subscribe to messages on specific topics. The broker is responsible for routing messages between clients, and for maintaining the state of the network.

One of the key features of MQTT is its low overhead, which makes it well-suited for low-bandwidth, high-latency, or unreliable networks. The protocol uses a binary format for messages, which reduces the amount of data that needs to be transmitted.

MQTT also includes a number of features that are designed to improve reliability and security, such as:

- Quality of Service (QoS) levels: MQTT supports three levels of QoS (0, 1, and 2) which allow clients to control the level of reliability for the messages they send and receive.
- Keep-Alive: MQTT includes a keep-alive mechanism that allows clients to detect when a connection has been lost and re-establish a connection if necessary.
- Clean Session: MQTT supports clean session, which allows clients to start a new session and discard any previous session data.
- Authentication and Encryption: MQTT supports username/password-based authentication, as well as Transport

Layer Security (TLS) encryption to secure the communication between clients and the broker.

MQTT is widely used in IoT applications such as smart home, industrial automation, and transportation. Many IoT platforms such as AWS IoT, Azure IoT, and Google IoT Core, support MQTT as a protocol for communication.

In summary, MQTT is a lightweight publish-subscribe messaging protocol that is designed for low-bandwidth, high-latency, or unreliable networks. It is widely used in IoT applications and provide features such as low overhead, Quality of Service, Keep-Alive, Clean Session, Authentication and Encryption which makes it a reliable and secure protocol for communication.

****