📘 IoT Unit–4: Communication Protocols

RGPV IMPORTANT QUESTIONS WITH EASY EXPLAINATION

---

1. MQTT

Introduction

MQTT ka full form **Message Queuing Telemetry Transport** hai. Ye IoT devices ke liye lightweight communication protocol hai.

Definition

MQTT is a lightweight publish-subscribe messaging protocol used for communication between IoT devices through a broker.

Why It Is Needed

IoT devices low-power, low-memory aur low-bandwidth hote hain. MQTT कम data use karta hai, isliye IoT ke liye best hai.

Easy Explanation

MQTT me device direct dusre device ko message nahi bhejta. Message pehle **broker** ke paas jata hai, fir broker correct subscriber ko message bhejta hai.

Step-by-Step Working

1. Sensor data collect karta hai.

2. Sensor **publisher** ban kar data broker ko bhejta hai.
3. Broker topic check karta hai.
4. Jo device us topic ko subscribe karta hai, use data milta hai.
5. User mobile app par output dekh sakta hai.

Flow of Process

```
Publisher → Broker → Subscriber
```

Diagram

```
Temperature Sensor
       ↓ publish
   MQTT Broker
       ↓ forward
   Mobile App
```

Real-Life Analogy

Broker ko WhatsApp group admin samjho. Publisher message bhejta hai, broker message correct group members tak pahunchata hai.

Advantages

- Lightweight
- Low bandwidth use
- Low power devices ke liye useful
- Many subscribers possible
- Remote monitoring ke liye best

Disadvantages

- Broker par dependency hoti hai
- Security setup required
- Broker fail hua to communication ruk sakta hai

Applications

- Smart home
- Smart agriculture
- Vehicle tracking
- Healthcare monitoring
- Industrial IoT

Important Keywords

MQTT, Publisher, Subscriber, Broker, Topic, Lightweight, Publish-Subscribe

Conclusion

MQTT IoT ke liye popular lightweight protocol hai jo publisher-broker-subscriber model par kaam karta hai.

---

2. MQTT Publish-Subscribe Model

Introduction

MQTT ka main working model **publish-subscribe** hai. Is model me sender aur receiver directly connected nahi hote.

Definition

Publish-subscribe model is a communication model where publishers send messages to a broker and subscribers receive messages based on subscribed topics.

Why It Is Needed

Isse devices loosely connected rehte hain. Publisher ko pata nahi hota subscriber kaun hai.

Easy Explanation

- Publisher = data bhejne wala
- Subscriber = data receive karne wala
- Broker = beech ka manager
- Topic = message ka subject/path

Step-by-Step Working

1. Subscriber topic subscribe karta hai.
2. Publisher same topic par message publish karta hai.
3. Broker topic match karta hai.
4. Broker message subscriber ko send karta hai.

Flow

```
Subscriber subscribes topic
        ↓
Publisher publishes message
        ↓
Broker matches topic
        ↓
Subscriber receives message
```

Diagram

```
Sensor publishes: home/temp
          ↓
       Broker
          ↓
App subscribed: home/temp
```

Example

Topic: `home/room1/temperature`

Sensor sends: `30°C`

Mobile app receives: `30°C`

Advantages

- Flexible communication
- Multiple subscribers possible
- Sender and receiver independent
- Scalable for IoT

Disadvantages

- Broker failure issue
- Topic management needed
- Security configuration required

Important Keywords

Publisher, Subscriber, Broker, Topic, Decoupling, Message Delivery

Conclusion

Publish-subscribe model MQTT ko scalable and efficient banata hai.

3. MQTT Topics and Methods

Introduction

MQTT me topic message ka address hota hai. MQTT methods communication ko control karte hain.

Definition

MQTT topics are named paths used to categorize messages, and MQTT methods are control messages used for connection, publishing and subscription.

MQTT Topics

Topic ek path jaisa hota hai.

Examples:

```
home/light/status
farm/soil/moisture
hospital/patient/heartRate
```

Important MQTT Methods

Method	Work
CONNECT	Broker se connection start
CONNACK	Connection accepted response
PUBLISH	Message send
SUBSCRIBE	Topic subscribe
SUBACK	Subscription confirmation

Method	Work
UNSUBSCRIBE	Topic remove
PINGREQ	Connection alive check
DISCONNECT	Connection close

Flow

```
CONNECT → SUBSCRIBE/PUBLISH → MESSAGE TRANSFER → DISCONNECT
```

Diagram

```
Client → CONNECT → Broker
Client → PUBLISH → Topic
Client → SUBSCRIBE → Topic
Client → DISCONNECT
```

Advantages

- Topic-wise message filtering
- Organized communication
- Easy monitoring
- Many devices manage ho sakte hain

Applications

- Smart home topics
- Sensor monitoring
- Industrial alerts
- Weather systems

Important Keywords

Topic, CONNECT, PUBLISH, SUBSCRIBE, DISCONNECT, Client

Conclusion

MQTT topics and methods IoT communication ko structured and manageable banate hain.

---

4. SMQTT

Introduction

SMQTT ka full form **Secure MQTT** hai. Ye MQTT ka secure version hai.

Definition

SMQTT is a secure extension of MQTT that provides encrypted and authenticated communication for IoT devices.

Why It Is Needed

Normal MQTT me sensitive data leak ho sakta hai. Smart home, healthcare aur industry me security important hoti hai.

Easy Explanation

SMQTT message ko lock karke bhejta hai. Sirf authorized receiver hi message read kar sakta hai.

Step-by-Step Working

1. Client secure connection banata hai.
2. Authentication hoti hai.

3. Message encrypt hota hai.
4. Broker encrypted message forward karta hai.
5. Subscriber message decrypt karta hai.

Flow

```
Authentication → Encryption → Broker Transfer → Decryption
```

Diagram

```
Publisher → Encrypted Message → Broker → Subscriber
```

Advantages

- Secure data transfer
- Privacy protection
- Authentication support
- Sensitive IoT systems ke liye useful

Disadvantages

- Processing power zyada lagti hai
- Low-power devices par load badhta hai
- Implementation complex hoti hai

Applications

- Healthcare IoT
- Smart locks

- Banking IoT
- Industrial security systems

Important Keywords

SMQTT, Security, Encryption, Authentication, Secure Messaging

Conclusion

SMQTT IoT communication ko secure banata hai and sensitive data protection ke liye important hai.

---

5. CoAP

Introduction

CoAP ka full form **Constrained Application Protocol** hai. Ye small IoT devices ke liye lightweight web protocol hai.

Definition

CoAP is a lightweight request-response protocol designed for constrained IoT devices and low-power networks.

Why It Is Needed

IoT devices ki memory, battery aur processing power limited hoti hai. CoAP unke liye HTTP jaisa but lighter protocol provide karta hai.

Easy Explanation

CoAP me client request bhejta hai aur server response deta hai.

Step-by-Step Working

1. Client request send karta hai.
2. Server request receive karta hai.
3. Server data process karta hai.
4. Server response bhejta hai.

Flow

```
Client → Request → Server
Client ← Response ← Server
```

Diagram

```
Mobile App → GET /temperature → Sensor
Mobile App ← 28°C ← Sensor
```

Advantages

- Very lightweight
- Low power devices ke liye suitable
- REST style communication
- Fast because UDP based

Disadvantages

- UDP reliability issue
- Security setup needed
- Complex systems ke liye limited

Applications

- Smart lighting
- Smart meters
- Sensor networks
- Home automation

Important Keywords

CoAP, Constrained Device, Request-Response, REST, UDP, Lightweight

Conclusion

CoAP small and constrained IoT devices ke liye lightweight request-response protocol hai.

---

6. CoAP Message Types

Introduction

CoAP me different message types hote hain jo reliability and communication control karte hain.

Definition

CoAP message types define how messages are sent and acknowledged between client and server.

Types of CoAP Messages

Message Type	Meaning
Confirmable	ACK required
Non-confirmable	ACK not required
Acknowledgement	Message received confirmation

Message Type	Meaning
Reset	Message not processed

Easy Explanation

Confirmable

Important message. Receiver must reply with ACK.

Non-confirmable

Less important message. ACK required nahi.

Acknowledgement

Receiver bolta hai: “Message mil gaya.”

Reset

Receiver bolta hai: “Message process nahi hua.”

Diagram

```
Confirmable:
Client → CON Message → Server
Client ← ACK ← Server

Non-confirmable:
Client → NON Message → Server
```

Advantages

- Reliability control
- Important messages confirm hote hain

- Lightweight communication maintain hoti hai

Disadvantages

- Confirmable messages me delay ho sakta hai
- Non-confirmable message lost ho sakta hai

Applications

- Sensor request
- Smart light control
- Smart meter reading

Important Keywords

Confirmable, Non-confirmable, ACK, Reset, Reliability

Conclusion

CoAP message types IoT communication me reliability and speed balance karte hain.

---

7. MQTT vs CoAP

Introduction

MQTT and CoAP dono IoT ke lightweight protocols hain, but dono ka working model different hai.

Definition

MQTT is a publish-subscribe protocol, while CoAP is a request-response protocol for constrained IoT devices.

Comparison Table

Basis	MQTT	CoAP
Full Form	Message Queuing Telemetry Transport	Constrained Application Protocol
Model	Publish-subscribe	Request-response
Transport	TCP	UDP
Middle Component	Broker required	Broker not required
Best Use	Continuous sensor updates	Direct device control
Reliability	TCP based reliable	ACK based reliability
Architecture	Many-to-many	Client-server
Example	Temperature monitoring	Smart bulb ON/OFF

Which is Better and Why?

MQTT is better for continuous monitoring like temperature, humidity, vehicle tracking.

CoAP is better for constrained devices and direct control like turning light ON/OFF.

Exam line:

MQTT and CoAP are both important; selection depends on application requirement.

Diagram

```
MQTT:
Publisher → Broker → Subscriber

CoAP:
Client → Request → Server
Client ← Response ← Server
```

Important Keywords

Publish-Subscribe, Request-Response, TCP, UDP, Broker, Lightweight

Conclusion

MQTT is best for publish-subscribe communication, while CoAP is best for lightweight request-response communication.

---

8. XMPP

Introduction

XMPP ka full form **Extensible Messaging and Presence Protocol** hai. Ye real-time messaging ke liye use hota hai.

Definition

XMPP is a communication protocol used for real-time messaging, presence information and device-to-device communication.

Why It Is Needed

IoT systems me real-time alerts and notifications chahiye hote hain. XMPP ye support karta hai.

Easy Explanation

XMPP chat app jaisa kaam karta hai. Sensor alert bhejta hai and server user ko message forward karta hai.

Step-by-Step Working

1. IoT device XMPP server se connect hota hai.
2. Device message send karta hai.
3. XMPP server message forward karta hai.

4. Receiver real-time notification receive karta hai.

Flow

```
IoT Device → XMPP Server → User Device
```

Diagram

```
Sensor Alert → XMPP Server → Mobile App
```

Advantages

- Real-time messaging
- Presence support
- Extensible
- Secure communication possible

Disadvantages

- MQTT/CoAP se heavy
- Low-power devices ke liye less suitable
- More bandwidth use kar sakta hai

Applications

- Smart home alerts
- Device status updates
- Emergency notifications
- Real-time IoT messaging

Important Keywords

XMPP, Real-time Messaging, Presence, Extensible, Alerts

Conclusion

XMPP IoT me real-time messaging and notification ke liye useful protocol hai.

---

9. AMQP Features and Components

Introduction

AMQP ka full form **Advanced Message Queuing Protocol** hai. Ye reliable message queuing ke liye use hota hai.

Definition

AMQP is a reliable message-oriented protocol used for secure message queuing, routing and delivery.

Why It Is Needed

Large enterprise systems me message loss nahi hona chahiye. AMQP reliable message delivery provide karta hai.

AMQP Features

Feature	Meaning
Reliability	Message safely deliver hota hai
Queuing	Message queue me store hota hai
Routing	Message correct receiver tak jata hai
Security	Secure transfer support

Feature	Meaning
Interoperability	Different systems connect ho sakte hain

AMQP Components

Component	Work
Producer	Message create/send karta hai
Exchange	Message route karta hai
Queue	Message store karti hai
Consumer	Message receive karta hai
Broker	Messaging system manage karta hai

Step-by-Step Working

1. Producer message bhejta hai.
2. Exchange message receive karta hai.
3. Exchange routing rule ke according queue select karta hai.
4. Queue message store karti hai.
5. Consumer queue se message receive karta hai.

Flow

```
Producer → Exchange → Queue → Consumer
```

Diagram

```
Producer
   ↓
Exchange
   ↓
```

```
Queue
   ↓
Consumer
```

Real-Life Analogy

Courier system jaisa: sender parcel deta hai, sorting center route karta hai, warehouse store karta hai, receiver receive karta hai.

Advantages

- Highly reliable
- Message queuing support
- Secure
- Enterprise applications ke liye useful

Disadvantages

- Heavy protocol
- Setup complex
- Low-power IoT devices ke liye less suitable

Applications

- Banking systems
- Enterprise IoT
- Industrial systems
- Cloud messaging

Important Keywords

AMQP, Producer, Exchange, Queue, Consumer, Broker, Reliable Messaging

Conclusion

AMQP reliable and secure message queuing protocol hai, especially enterprise-level IoT systems ke liye.

---

10. AMQP Frame Types

Introduction

AMQP me data frames ke form me transfer hota hai. Frames structured communication provide karte hain.

Definition

AMQP frame types are structured data units used to transfer commands, message properties, actual data and connection status.

Main AMQP Frame Types

Frame Type	Work
Method Frame	Command carry karta hai
Header Frame	Message properties carry karta hai
Body Frame	Actual message data carry karta hai
Heartbeat Frame	Connection alive check karta hai

Easy Explanation

AMQP message ek file jaisa hai:

- Method = kya karna hai
- Header = details
- Body = actual message

- Heartbeat = connection alive hai ya nahi

Step-by-Step Working

1. Method frame command send karta hai.
2. Header frame message details send karta hai.
3. Body frame actual message bhejta hai.
4. Heartbeat frame connection check karta hai.

Diagram

```
AMQP Communication
     |
Method Frame
Header Frame
Body Frame
Heartbeat Frame
```

Advantages

- Structured communication
- Reliable transfer
- Connection monitoring possible
- Enterprise messaging support

Disadvantages

- More overhead
- Complex protocol
- Not ideal for very small devices

Important Keywords

Method Frame, Header Frame, Body Frame, Heartbeat Frame, AMQP

Conclusion

AMQP frame types reliable and structured message communication provide karte hain.

🔥 Comparison Table: MQTT vs CoAP vs XMPP vs AMQP

Basis	MQTT	CoAP	XMPP	AMQP
Model	Publish-subscribe	Request-response	Real-time messaging	Message queuing
Transport	TCP	UDP	TCP	TCP
Weight	Lightweight	Very lightweight	Medium	Heavy
Main Use	Sensor updates	Constrained devices	Alerts/chat	Enterprise messaging
Component	Broker	Client-server	XMPP server	Broker/exchange/queue
IoT Suitability	Very high	Very high	Medium	Medium

🔥 Most Important 7-Mark Questions

1. Explain MQTT with components.
2. Explain MQTT publish-subscribe model.
3. Explain MQTT topics and methods.
4. Explain CoAP protocol.
5. Explain CoAP message types.
6. Differentiate between MQTT and CoAP.
7. Explain XMPP.
8. Explain AMQP features and components.

9. Explain AMQP frame types.
10. Explain SMQTT.

⭐ Most Important 14-Mark Questions

1. Explain MQTT protocol with methods, topics, components and applications.
2. Explain CoAP with message types and request-response model.
3. Compare MQTT, CoAP, XMPP and AMQP.
4. Explain AMQP features, components and frame types.
5. Explain IoT communication protocols in detail.

🎯 PYQ-Based Expected Questions

Very High Probability

✅ MQTT
✅ CoAP
✅ MQTT vs CoAP
✅ MQTT publish-subscribe model

High Probability

✅ CoAP message types
✅ AMQP features and components
✅ XMPP

Medium Probability

✅ SMQTT
✅ AMQP frame types
✅ MQTT topics and methods

⚡ One-Night Revision Notes

Topic	Quick Revision
MQTT	Publish-subscribe lightweight protocol
Broker	Message distributor
Topic	Message category/path
CoAP	Lightweight request-response protocol
XMPP	Real-time messaging
AMQP	Reliable message queuing
SMQTT	Secure MQTT
ACK	Message received confirmation
Heartbeat	Connection alive checking

📌 Smart Study Plan

2-Hour Plan

Time	Topic
30 min	MQTT
25 min	CoAP
20 min	MQTT vs CoAP
20 min	AMQP
15 min	XMPP
10 min	SMQTT + frame types

5-Hour Plan

Time	Topic
1 hour	MQTT full
1 hour	CoAP full

Time	Topic
1 hour	AMQP full
45 min	XMPP + SMQTT
45 min	Comparison tables
30 min	Diagrams practice

🧠 Memory Tricks

MQTT Model

PBS

- P = Publisher
- B = Broker
- S = Subscriber

CoAP Methods

GPUD

- G = GET
- P = POST
- U = PUT
- D = DELETE

CoAP Message Types

CNAR

- C = Confirmable
- N = Non-confirmable
- A = Acknowledgement
- R = Reset

AMQP Components

PEQCB

- P = Producer
- E = Exchange
- Q = Queue
- C = Consumer
- B = Broker

🚀 Topper Answer Writing Tips

For 7 marks:

```
Definition
↓
Working model
↓
Diagram
↓
Advantages
↓
Applications
↓
Conclusion
```

For 14 marks:

```
Introduction
↓
Definition
↓
```

```
Components
↓
Methods / Message types
↓
Detailed working
↓
Diagram
↓
Comparison table
↓
Applications
↓
Conclusion
```

Keywords to Underline

MQTT, Broker, Publisher, Subscriber, Topic, CoAP, Request-Response, Confirmable, Acknowledgement, AMQP, Exchange, Queue, XMPP, SMQTT

Final tip: Unit-4 me sabse pehle **MQTT + CoAP + MQTT vs CoAP + AMQP** prepare karo. Ye sabse scoring topics hain.