

# Testing MQTT messaging with a server in the network

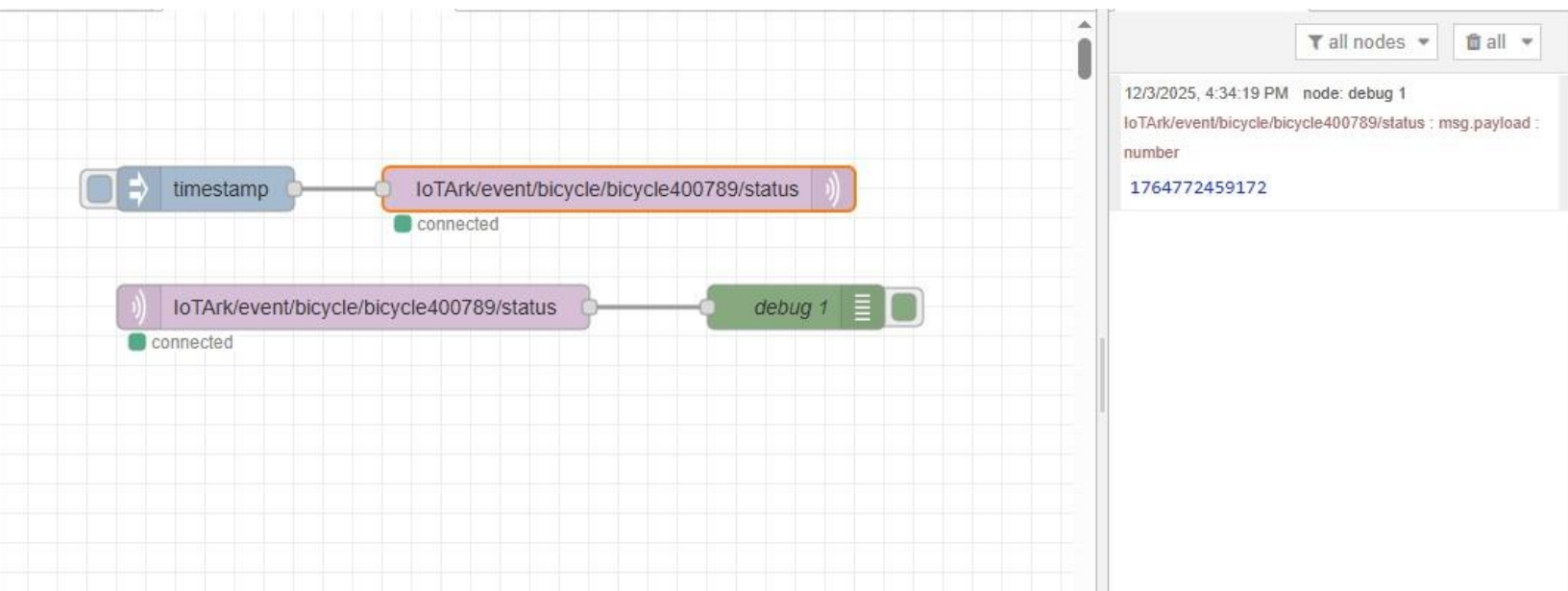
Name: Dipen Gaihre

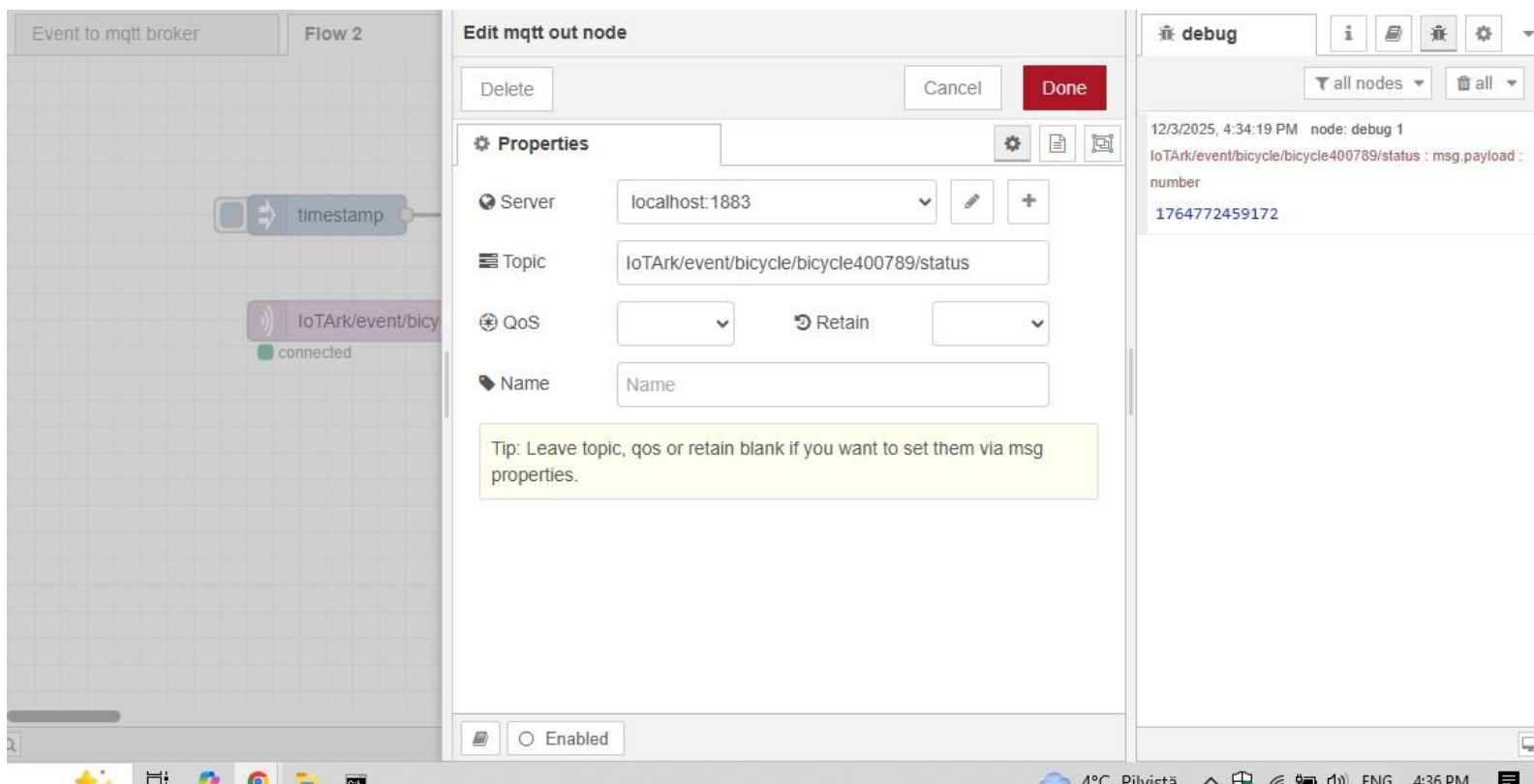
## Report Explanation:

This report shows the results of testing MQTT topic subscriptions using Node-RED. For each exercise, the flow diagrams and debug panel outputs are provided to demonstrate that the MQTT client correctly receives messages according to the specified topics. The tests include subscribing to a single bicycle status, all bicycle statuses, and all messages from the IoTArk factory, while ignoring messages from other factories or unrelated topics.

I have submitted screenshots of the tasks:

## Question 1 – Exact Topic Subscription



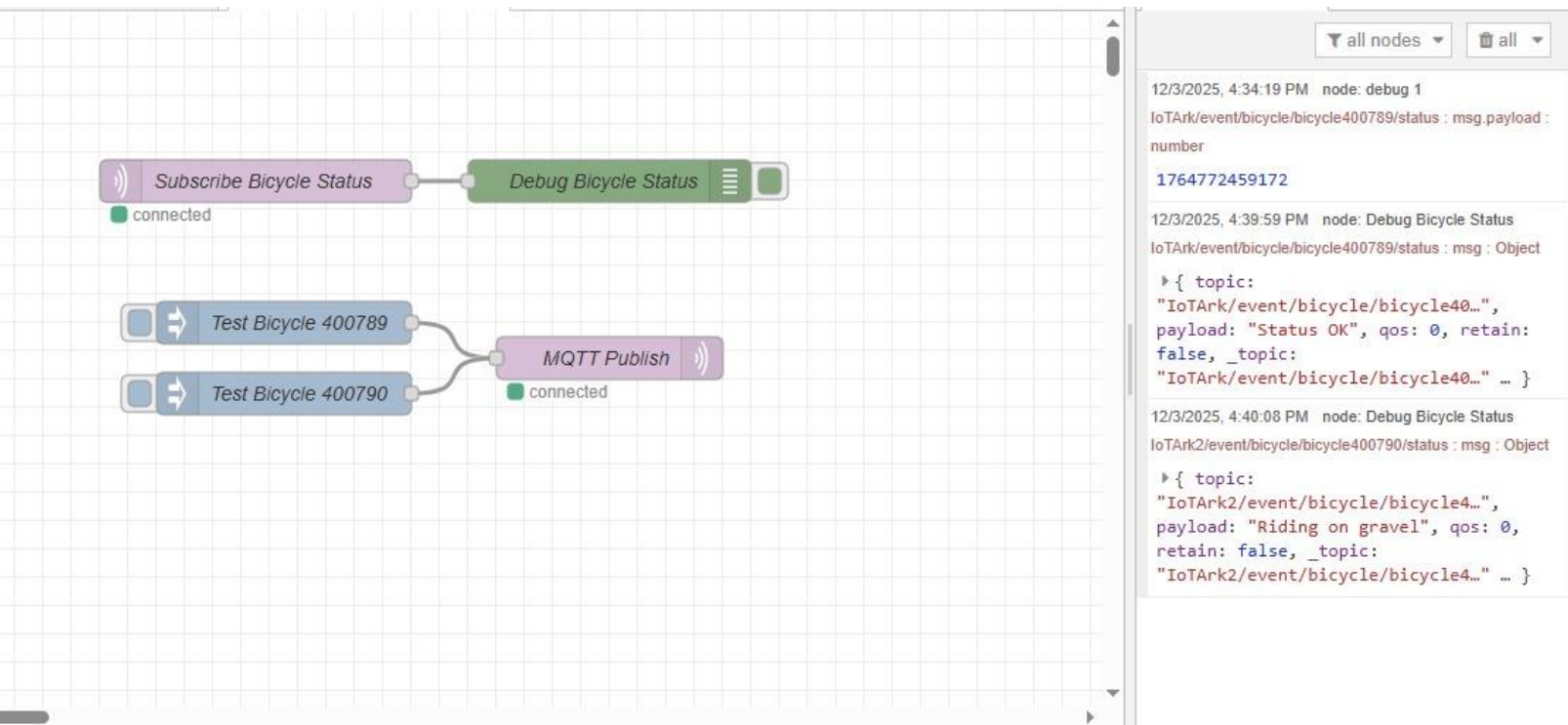


*Fig: question 1*

**Comment:**

- Subscribed to the exact topic of one bicycle.
- Only messages from this topic appear in the debug panel.
- Confirms subscription works for a single, specific topic.

## Question 2 – Status of Any Bicycle



*Fig: question 2*

### Comment:

- Subscribed using `+/event/bicycle/+/status` to catch any bicycle status message.
- Messages from all factories and all bicycle serial numbers appear.
- Messages from other device types (scooters, alarms) do not appear.

### Question 3 – All IoTArk Messages

The screenshot shows the Node-RED web interface. On the left, a flow named 'Flow 4' is active. It starts with a 'Subscribe All IoTArk' node (purple) which is connected to a 'Debug IoTArk Messages' node (green). Below this, there are four input nodes: 'Bicycle Status', 'Bicycle Alarm', 'Scooter Status', and 'Other Factory', all connected to an 'MQTT Publish' node (purple). The 'MQTT Publish' node is also connected to the 'Debug IoTArk Messages' node. The right panel shows the 'debug' console with a list of received messages. The messages are JSON objects with a 'topic' and a 'payload'. The topics include 'IoTArk/event/bicycle/bicycle400789/status', 'IoTArk/event/bicycle/bicycle400789/incident', 'IoTArk/event/scooter/scooter123/status', and 'IoTArk2/event/bicycle/bicycle400789/status'. The payloads include 'Riding on asphalt', 'Battery fire!', 'Idle', and 'Should not appear'.

```
12/3/2025, 4:43:55 PM node: Debug Bicycle Status
IoTArk/event/bicycle/bicycle400789/status : msg : Object
  { topic: "IoTArk/event/bicycle/bicycle40...", payload: "Riding on asphalt", qos: 0, retain: false, _topic: "IoTArk/event/bicycle/bicycle40..." }

12/3/2025, 4:43:55 PM node: Debug IoTArk Messages
IoTArk/event/bicycle/bicycle400789/status : msg : Object
  { topic: "IoTArk/event/bicycle/bicycle40...", payload: "Riding on asphalt", qos: 0, retain: false, _topic: "IoTArk/event/bicycle/bicycle40..." }

12/3/2025, 4:43:58 PM node: Debug IoTArk Messages
IoTArk/alarm/alarmClass3/bicycle/bicycle400789/incident : msg : Object
  { topic: "IoTArk/alarm/alarmClass3/bicyc...", payload: "Battery fire!", qos: 0, retain: false, _topic: "IoTArk/alarm/alarmClass3/bicyc..." }

12/3/2025, 4:44:02 PM node: Debug IoTArk Messages
IoTArk/event/scooter/scooter123/status : msg : Object
  { topic: "IoTArk/event/scooter/scooter12...", payload: "Idle", qos: 0, retain: false, _topic: "IoTArk/event/scooter/scooter12..." }

12/3/2025, 4:44:04 PM node: Debug Bicycle Status
IoTArk2/event/bicycle/bicycle400789/status : msg : Object
  { topic: "IoTArk2/event/bicycle/bicycle4...", payload: "Should not appear", qos: 0, retain: false, _topic: "IoTArk2/event/bicycle/bicycle4..." }
```

*Fig: question 3*

#### Comment:

- Subscribed using IoTArk/# to catch all messages from IoTArk factory.
- Includes all devices and all events or alarms.
- Messages from other factories (e.g., IoTArk2) are excluded.

**The End**