

# Документация: Система команд MQTT для многофункционального сенсора на ESP32

Версия: 1.1  
Дата: 2024-04-26  
Автор: [Укажите автора]

## Оглавление

- 1. Введение
- 2. Термины и определения
- 3. Архитектура топиков MQTT
- 4. Общие соглашения
- 5. Спецификация команд и событий
- 6. Примеры Payload
- 7. Рекомендации по реализации
- 8. Таблица команд
- 9. Контакты

## 1. Введение

Данный документ описывает структуру команд, событий и обмена данными для многофункционального устройства на базе ESP32, взаимодействующего по протоколу MQTT. Устройство поддерживает:

- Считывание RFID меток (13.56 MHz)
- Адресную светодиодную индикацию (кольцо, ~24 LED)
- Внешнюю адресную светодиодную ленту
- Два силовых релейных выхода
- Два цифровых входа
- Wi-Fi и Bluetooth
- Питание от 5V, 12V и аккумулятора 3.7V
- Свободные GPIO для расширения

## 2. Термины и определения

Термин	Описание
MQTT	Легковесный протокол обмена сообщениями publish/subscribe
Payload	Содержимое сообщения (обычно JSON или текст)
Topic	Иерархический путь сообщения в MQTT
QoS	Уровень качества доставки сообщения (Quality of Service)
Retain	Флаг сохранения последнего сообщения в брокере
NVS	Энергонезависимая память ESP32
	Логическое/физическое расположение устройства
<device_type>	Тип устройства (например, RFIDRelay)
<device_id>	Уникальный идентификатор устройства

## 3. Архитектура топиков MQTT

Все топики строятся по шаблону:

```
<prefix>/<location>/<device_type>/<device_id>/<command_path>
```

Сегменты топика:

- prefix** — тип сообщения:

- `cmdnd/` — команды устройству
- `stat/` — статусы/ответы от устройства
- `tele/` — телеметрия и события
- — расположение (например, `Floor1`)
- `<device_type>` — тип устройства (например, `RFIDRelay`)
- `<device_id>` — уникальный ID (например, `UnitA5`)
- `command_path` — путь функции (например, `relay/1/set`)

## 4. Общие соглашения

- **Подписка:** Устройство подписывается на:  
`cmdnd/<location>/<device_type>/<device_id>/#`
- **Публикация:** Устройство публикует в:  
`stat/<location>/<device_type>/<device_id>/...` и `tele/<location>/<device_type>/<device_id>/...`
- **Конфигурируемость:** `<location>`, `<device_type>`, `<device_id>` — настраиваемы, сохраняются в NVS.
- **Payload:**
  - По умолчанию — JSON.
  - Для простых команд — текст (`ON`, `OFF`, `TOGGLE`).
  - Все строки в JSON — в двойных кавычках.
- **Timestamp:** Используйте Unix timestamp (миллисекунды).
- **QoS:** Рекомендуется QoS 1 для команд и важных статусов.
- **Retain:** Используйте для `stat/.../status` и `stat/.../config`.

**Внимание!** Передача паролей в открытом виде не рекомендуется. Используйте безопасные методы первичной настройки.

## 5. Спецификация команд и событий

### 5.1 RFID (Считыватель меток)

Событие/Действие	Тип топика	Топик	Payload/Пример
Считана метка	tele	<code>tele/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/rfid/tag</code>	<code>{ "tag": "UID", "timestamp": 1678886400123 }</code>
Запрос последней метки	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/rfid/last/get</code>	( <i>нустой</i> )
Статус последней метки	stat	<code>stat/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/rfid/last</code>	<code>{ "tag": "UID", "timestamp": ... }</code> или <code>null</code>

### 5.2 LED (Кольцо и внешние)

Событие/Действие	Тип топика	Топик	Payload/Пример
Установить цвет/яркость	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/led/ring/set</code>	<code>{ "num": 4, "color": [0, 255, 0], "br": 128 }</code>
Включить паттерн	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/led/ring/pattern</code>	<code>{ "name": "rainbow", ... }</code>
Запрос состояния	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/led/ring/get</code>	( <i>нустой</i> )
Статус состояния	stat	<code>stat/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/led/ring/state</code>	<code>{ "mode": "pattern", ... }</code>
...	...	...	...

### 5.3 Реле

Событие/Действие	Тип топика	Топик	Payload/Пример
Управление реле (1шт)	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/relay/1/set</code>	<code>ON</code> , <code>OFF</code> , <code>TOGGLE</code>
Управление группой	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/relay/set</code>	<code>{ "relay1": "ON", "relay2": "OFF" }</code>
PulseTime для реле	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/relay/1/pulsetime</code>	<code>1000 (ms)</code>
Запрос состояния	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/relay/get</code>	( <i>нустой</i> )
Статус реле	stat	<code>stat/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/relay</code>	<code>{ "relay1": {...}, "relay2": {...} }</code>

### 5.4 Входы (Input)

Событие/Действие	Тип топика	Топик	Payload/Пример
Изменение входа	tele	<code>tele/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/input</code>	<code>{ "input1": { "state": 1, "timestamp": ... } }</code>
Запрос состояния	cmdnd	<code>cmdnd/&lt;L&gt;/&lt;T&gt;/&lt;D&gt;/input/get</code>	( <i>нустой</i> )

Событие/Действие	Тип топика	Топик	Payload/Пример
Статус входов	stat	stat/<L>/<T>/<D>/input	{ "input1": { "state": 1 }, ... }
Настроить вход	cmdnd	cmdnd/<L>/<T>/<D>/input/1/config	{ "mode": "INPUT_PULLUP", ... }
...	...	...	...

## 5.5 Питание (Power)

Событие/Действие	Тип топика	Топик	Payload/Пример
Телеметрия питания	tele	tele/<L>/<T>/<D>/power	{ "source": "battery", ... }

## 5.6 Система (System)

Событие/Действие	Тип топика	Топик	Payload/Пример
Запрос всего статуса	cmdnd	cmdnd/<L>/<T>/<D>/status	(нужной)
Полный статус	stat	stat/<L>/<T>/<D>/status	См. пример ниже
Перезагрузка	cmdnd	cmdnd/<L>/<T>/<D>/reboot	(нужной)
...	...	...	...

## 5.7 Конфигурирование (Config)

Событие/Действие	Тип топика	Топик	Payload/Пример
Установить конфиг	cmdnd	cmdnd/<L>/<T>/<D>/config	См. пример ниже
Запросить конфиг	cmdnd	cmdnd/<L>/<T>/<D>/config/get	(нужной)
Статус конфига	stat	stat/<L>/<T>/<D>/config	...без паролей...

## 5.8 GPIO

Событие/Действие	Тип топика	Топик	Payload/Пример
Настроить режим GPIO	cmdnd	cmdnd/<L>/<T>/<D>/gpio/5/mode	{ "mode": "OUTPUT" }
Управление состоянием	cmdnd	cmdnd/<L>/<T>/<D>/gpio/5/set	1, 0, TOGGLE
Запрос состояния	cmdnd	cmdnd/<L>/<T>/<D>/gpio/5/get	(нужной)
Статус GPIO	stat	stat/<L>/<T>/<D>/gpio/5	{ "state": 1 }
Событие входа на GPIO	tele	tele/<L>/<T>/<D>/gpio/5	{ "state": 0, "timestamp": ... }

## 5.9 Bluetooth (BT)

Событие/Действие	Тип топика	Топик	Payload/Пример
Сканирование BLE	cmdnd	cmdnd/<L>/<T>/<D>/bt/scan	{ "duration_ms": 10000 }
Статус сканирования	stat	stat/<L>/<T>/<D>/bt/status	{ "scanning": true }
Обнаружено BLE-устр.	tele	tele/<L>/<T>/<D>/bt/device	{ "mac": "...", "rssi": -70, ... }

## 6. Примеры Payload

```
// Топик: cmdnd/Floor1/RFIDRelay/UnitA5/led/ring/set
{
  "leds": [
    { "num": 0, "color": [255, 0, 0] },
    { "num": 1, "color": [0, 255, 0], "br": 128 },
    { "num": 2, "color": [0, 0, 255] }
  ]
}
```

```
// Топик: cmdnd/Floor1/RFIDRelay/UnitA5/relay/set
{
  "relay1": "ON",
}
```

```
"relay2": "TOGGLE"
}
```

```
// Топик: stat/Floor1/RFIDRelay/UnitA5/status
{
  "device": {
    "location": "Floor1",
    "type": "RFIDRelay",
    "id": "UnitA5",
    "name": "Main Entrance Reader",
    "fw_version": "1.1.0",
    "uptime_s": 14400,
    "free_heap": 180000
  },
  "network": {
    "wifi_ssid": "MyHomeWiFi",
    "ip": "192.168.1.105",
    "mac": "DE:AD:BE:EF:01:A5",
    "rssi": -62
  },
  "mqtt": {
    "connected": true
  },
  "power": {
    "source": "12V"
  },
  "relay": {
    "relay1": {"state": "OFF", "pulsetime_ms": 0},
    "relay2": {"state": "OFF", "pulsetime_ms": 0}
  },
  "input": {
    "input1": {"state": 1},
    "input2": {"state": 0}
  },
  "led_ring": {"mode": "off"},
  "rfid_last": {"tag": "04FF1A89CD6180", "timestamp": 1678888000123}
}
```

## 7. Рекомендации по реализации

- Используйте надежную библиотеку для парсинга JSON (например, ArduinoJson)
- Сохраняйте конфигурацию в NVS
- Используйте асинхронный код (FreeRTOS задачи)
- Обработывайте ошибки (WiFi, MQTT, NVS, датчики)
- Храните учетные данные безопасно, используйте TLS для MQTT
- Тестируйте все команды и события

## 8. Таблица команд

Функция	Событие/ Действие	Тип топика	Путь топика (после ...//)	Пример Payload	Напра
RFID	Считана метка	tele	rfid/tag	{"tag": "04FF1A", "timestamp": 1678886400123}	Device Client
	Запрос последней метки	cmnd	rfid/last/get	( <i>нужной</i> )	Client - Device
	Статус последней метки	stat	rfid/last	{"tag": "04FF1A", "timestamp": 1678886400123} или null	Device Client
LED Ring	Установить цвет/яркость	cmnd	led/ring/set	{"num": 4, "color": [0, 255, 0], "br": 128} или другой формат	Client - Device
	Включить паттерн	cmnd	led/ring/pattern	{"name": "rainbow", "speed": 50, "brightness": 200}	Client - Device

Функция	Событие/ Действие	Тип топика	Путь топика (после .../)	Пример Payload	Напра
<b>LED Внешн.</b>	Запрос состояния	cmnd	led/ring/get	<i>(пустой)</i>	Client - Device
	Статус состояния	stat	led/ring/state	<code>{"mode":"pattern","name":"rainbow",...}</code>	Device Client
	Установить цвет/яркость	cmnd	led/external/set	Аналогично ring	Client - Device
	Включить паттерн	cmnd	led/external/pattern	Аналогично ring	Client - Device
	Запрос состояния	cmnd	led/external/get	<i>(пустой)</i>	Client - Device
	Статус состояния	stat	led/external/state	Аналогично ring	Device Client
<b>Реле</b>	Управление реле (1шт)	cmnd	relay/1/set (или 2/set)	<code>ON, OFF, TOGGLE</code>	Client - Device
	Управление группой	cmnd	relay/set	<code>{"relay1":"ON","relay2":"OFF"}</code>	Client - Device
	PulseTime для реле	cmnd	relay/1/pulsetime (или 2/...)	<code>1000 (ms)</code>	Client - Device
	Запрос состояния	cmnd	relay/get	<i>(пустой)</i>	Client - Device
	Статус реле	stat	relay	<code>{"relay1":{"state":"ON","pulsetime_ms":0},...}</code>	Device Client
<b>Входы</b>	Изменение входа	tele	input	<code>{"input1":{"state":1,"timestamp":...}}</code>	Device Client
	Запрос состояния	cmnd	input/get	<i>(пустой)</i>	Client - Device
	Статус входов	stat	input	<code>{"input1":{"state":1},"input2":{"state":0}}</code>	Device Client
	Настроить вход	cmnd	input/1/config (или 2/...)	<code>{"mode":"INPUT_PULLUP","inverted":false,"debounce_ms":50}</code>	Client - Device
	Запрос конфига входов	cmnd	input/config/get	<i>(пустой)</i>	Client - Device
	Статус конфига входов	stat	input/config	<code>{"input1":{...},"input2":{...}}</code>	Device Client
<b>Питание</b>	Телеметрия питания	tele	power	<code>{"source":"battery","voltage":3.8,"level_percent":70,"charging":false}</code>	Device Client
<b>Система</b>	Запрос всего статуса	cmnd	status	<i>(пустой)</i>	Client - Device
	Полный статус	stat	status	См. структуру в документе	Device Client
	Перезагрузка	cmnd	reboot	<i>(пустой)</i>	Client - Device
	Запрос возможностей	cmnd	capabilities	<i>(пустой)</i>	Client - Device
	Статус возможностей	stat	capabilities	<code>{"features":["rfid","led_ring:24","relay:2",... ]}</code>	Device Client
<b>Конфиг</b>	Установить конфиг	cmnd	config	См. структуру в документе	Client - Device
	Запросить конфиг	cmnd	config/get	<i>(пустой)</i>	Client - Device

Функция	Событие/ Действие	Тип топика	Путь топика (после .../)	Пример Payload	Напра
	Статус конфига	stat	config	...без паролей...	Device Client
GPIO	Настроить режим GPIO	cmnd	gpio/5/mode (пример для GPIO5)	<code>{"mode": "OUTPUT"}</code>	Client - Device
	Управление состоянием	cmnd	gpio/5/set	<code>1, 0, TOGGLE</code>	Client - Device
	Запрос состояния	cmnd	gpio/5/get	<i>(нуммой)</i>	Client - Device
	Статус GPIO	stat	gpio/5	<code>{"state": 1}</code>	Device Client
	Событие входа на GPIO	tele	gpio/5	<code>{"state": 0, "timestamp": ...}</code>	Device Client
Bluetooth	Сканирование BLE	cmnd	bt/scan	<code>{"duration_ms": 10000}</code>	Client - Device
	Статус сканирования	stat	bt/status	<code>{"scanning": true}</code>	Device Client
	Обнаружено BLE-устр.	tele	bt/device	<code>{"mac": "...", "rssi": -70, "name": "..."} </code>	Device Client

## 9. Контакты

Для вопросов и предложений: [укажите контактную информацию]