ESP8266 <-> Backend MQTT 联调协议

本文档基于当前后端代码实际行为整理，供单片机（ESP8266）联调使用。

- 后端 MQTT 前缀：`dorm`（可由 `MQTT_TOPIC_PREFIX` 修改）
- Broker：`175.27.162.174:1883`（按你当前环境）
- 建议 QoS：`1`
- 编码：`UTF-8`

---

1. 主题命名（Topic）规范

后端兼容两种格式：

1. 单段设备 ID（历史兼容）

- `dorm/{deviceId}/status`
- `dorm/{deviceId}/telemetry`
- `dorm/{deviceId}/ack`
- `dorm/{deviceId}/event`
- `dorm/{deviceId}/lwt` / `will` / `offline`
- `dorm/{deviceId}/cmd`（设备订阅）

2. 房间 + 设备（推荐）

- `dorm/{room}/{device}/status`
- `dorm/{room}/{device}/telemetry`
- `dorm/{room}/{device}/ack`
- `dorm/{room}/{device}/event`
- `dorm/{room}/{device}/lwt` / `will` / `offline`
- `dorm/{room}/{device}/cmd`（设备订阅）

推荐统一用第 2 种，例如：

- `dorm/A-303/strip01/status`
- `dorm/A-303/strip01/cmd`

---

2. 设备端需要订阅的主题

设备至少订阅：

- `dorm/{room}/{device}/cmd`

示例：

- `dorm/A-303/strip01/cmd`

后端在下发命令时会同时尝试发布到：

- `dorm/{deviceId}/cmd`
- `dorm/{room}/{device}/cmd`（当设备 ID 可拆分时）

---

3. 后端下发给设备的命令消息格式（cmd payload）

Topic：

- `dorm/{room}/{device}/cmd`

Payload(JSON)：

```
{
  "cmdId": "cmd_1772000000_ab12cd34",
  "ts": 1772000000,
  "type": "ON",
  "socketId": 1,
  "payload": {},
  "mode": null,
  "duration": null,
  "source": "web"
}
```

字段说明：

- `cmdId`：命令唯一 ID，ACK 必须原样带回
- `type`：动作类型（常见 `ON` / `OFF`，或 `MODE` 等）
- `socketId`：目标插孔号，可为空
- `mode`、`duration`、`payload`：模式/定时/扩展参数

---

4. 设备上报给后端的消息格式

4.1 状态上报（status）

Topic：

- `dorm/{room}/{device}/status`

Payload(JSON)：

```
{
  "ts": 1772000001,
  "online": true,
  "total_power_w": 128.6,
  "voltage_v": 220.9,
  "current_a": 0.58,
  "sockets": [
    { "id": 1, "on": true,  "power_w": 82.0,  "device": "PC" },
    { "id": 2, "on": false, "power_w": 0.0,   "device": "None" },
```

```
    { "id": 3, "on": true,  "power_w": 46.6,  "device": "Lamp" },
    { "id": 4, "on": false, "power_w": 0.0,   "device": "None" }
  ]
}
```

最关键字段：

- `ts`
- `total_power_w` / `voltage_v` / `current_a`
- `sockets[].id` / `sockets[].on`

4.2 遥测上报（telemetry）

Topic：

- `dorm/{room}/{device}/telemetry`

Payload(JSON)：

```
{
  "ts": 1772000002,
  "power_w": 128.6,
  "voltage_v": 220.9,
  "current_a": 0.58
}
```

说明：

- `power_w` 推荐上报（后端也兼容从 `total_power_w` 取值）

4.3 命令回执（ack）

Topic：

- `dorm/{room}/{device}/ack`

Payload(JSON)：

```
{
  "cmdId": "cmd_1772000000_ab12cd34",
  "status": "success",
  "costMs": 320,
  "errorMsg": ""
}
```

字段说明：

- `cmdId`：必须与命令一致

- `status`：`success` 或 `failed`
- `costMs`：可选，执行耗时
- `errorMsg`：失败时建议填写

4.4 事件上报（event，可选）

Topic：

- `dorm/{room}/{device}/event`

Payload：业务自定义 JSON（当前后端不做核心业务处理，可用于扩展）

---

5. 设备离线（LWT/Will）规则

后端支持以下离线主题，收到即立即标记 Offline：

- `dorm/{room}/{device}/lwt`
- `dorm/{room}/{device}/will`
- `dorm/{room}/{device}/offline`

Payload 可为 JSON 或纯文本。

JSON 示例：

```
{
  "reason": "power_off",
  "ts": 1772000010
}
```

文本示例：

```
power_off
```

建议在 MQTT 连接时配置遗嘱消息（Will）：

- Will Topic：`dorm/{room}/{device}/lwt`
- Will Payload：`{"reason":"power_off"}`
- Will QoS：`1`
- Retain：`false`

---

6. 在线/离线判定行为

后端离线判定有两条：

1. 收到 `lwt/will/offline`：立即离线（强制）
2. 超过 `ONLINE_TIMEOUT_SECONDS` 未收到设备消息：超时离线（默认 10 秒）

设备重新上线后，只要继续上报 `status/telemetry/ack/event`，状态会恢复为 Online。

---

7. MQTTX 快速联调清单

7.1 订阅（观察后端下发）

- `dorm/A-303/strip01/cmd`

7.2 上报在线状态

Topic：

- `dorm/A-303/strip01/status`

Payload：

```
{
  "ts": 1772000100,
  "online": true,
  "total_power_w": 90.6,
  "voltage_v": 220.9,
  "current_a": 0.58,
  "sockets": [
    { "id": 1, "on": true, "power_w": 90.6, "device": "PC" }
  ]
}
```

7.3 模拟离线

Topic：

- `dorm/A-303/strip01/lwt`

Payload：

```
{
  "reason": "power_off"
}
```

7.4 命令 ACK 回执

Topic：

- `dorm/A-303/strip01/ack`

Payload：

```
{
  "cmdId": "cmd_1772000000_ab12cd34",
  "status": "success",
  "costMs": 280
}
```

---

8. 设备端实现建议（ESP8266）

- 建立 MQTT 后立即订阅 `.../cmd`
- 每 1~2 秒上报一次 `status`（或 2~5 秒按带宽折中）
- 每 1~5 秒上报一次 `telemetry`
- 收到命令后尽快执行并上报 `ack`
- 配置 MQTT Will，异常断开自动发布 `lwt`