



## API DQ\_HUB

## Mục lục

A. GIỚI THIỆU .....	5
B. CẤU HÌNH .....	7
C. TÍNH NĂNG.....	10
I. Version Hub.....	10
1. Get version Hub .....	10
2. Update FirmWare Hub .....	11
II. BLE SigMesh .....	12
1. Cập nhật mới file json từ server.....	12
2. Set địa chỉ unicast hub .....	13
3. Get địa chỉ unicast Hub .....	14
4. Get danh sách thiết bị hub quản lý .....	14
5. Get danh sách Group hub quản lý .....	17
6. Get danh sách Scene .....	17
7. Set On/Off thiết bị hoặc group .....	18
8. Get On/Off thiết bị hoặc group.....	19
9. Set Lightness thiết bị hoặc group.....	20
10. Get Lightness thiết bị hoặc group .....	21
11. Set HSL .....	21
12. Call Scene .....	22
13. Set vendor model.....	23
14. Reboot Hub .....	24
15. Get status devices .....	24
16. Trạng thái thiết bị trả về .....	26
III. Zigbee.....	29
1. Delete network .....	29
2. Init network.....	30
3. Bật cho phép pair thiết bị mới .....	30
4. Tắt cho phép pair thiết bị mới .....	31
5. Set model cho thiết bị.....	31
6. Delete thiết bị ra khỏi mạng .....	33
7. Set On off .....	34

8.	Set Lightness .....	35
9.	Set bind cho thiết bị .....	36
10.	Get list devices .....	38
11.	Status device telemetry .....	38
D.	KỊCH BẢN .....	40
I.	Chạy tự động.....	40
1.	Creat.....	40
2.	Update.....	42
3.	Get.....	44
4.	Delete.....	45
II.	Schedule.....	45
1.	Creat.....	45
2.	Update.....	47
3.	Get.....	48
4.	Delete.....	49

History Update

Version	Thời gian	Nội dung cập nhật
<b>1.0.2</b>	15/04/2023	Thêm list devices vào lệnh Get status devices, thêm cập nhật danh sách thiết bị disconnect qua telemetry
<b>1.0.2</b>	14/04/2023	Thêm get trạng thái tất cả thiết bị (Get status devices)

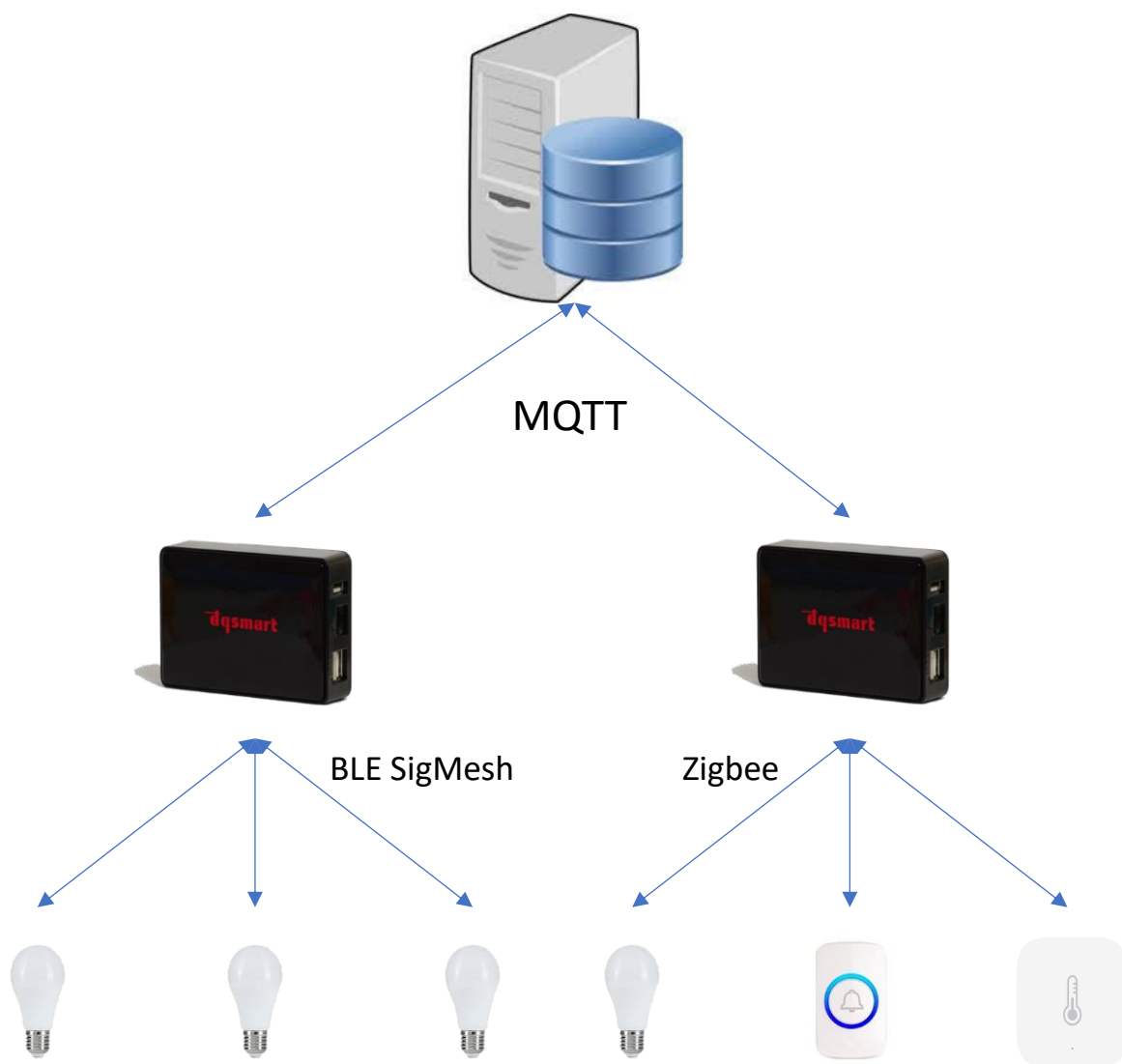
## A. GIỚI THIỆU

DQ\_hub giúp kết nối giữa các thiết bị BLE SIGMESH và ZIGBEE với Server thông qua mạng Internet, sử dụng giao thức MQTT. Hub có thể kết nối đến Server thông qua Gateway Điện Quang, hoặc kết nối trực tiếp đến Server mà không cần gateway, thông qua việc cấu hình Hub. Hub có các tính năng như sau:

- Kiểm tra version Firmware Hub
- Cập nhật FirmWare Hub (chỉ sử dụng với Server Điện Quang)
- Điều khiển và giám sát thiết bị BLE SigMesh
  - + On Off: điều khiển on/off từng đèn hoặc 1 nhóm đèn
  - + Dimming: điều khiển Lightness (độ sáng) từng đèn hoặc 1 nhóm đèn
  - + HSL: điều khiển màu sắc đèn hoặc nhóm đèn
  - + Scene: gọi ngữ cảnh đã cấu hình
  - + Lưu trạng thái mới nhất của đèn
  - + Phát hiện đèn hư hỏng
- Điều khiển và giám sát các thiết bị Zigbee
  - + On Off: điều khiển on off đèn
  - + Dimming: điều khiển Lightness đèn
  - + Cập nhật trạng thái nút nhấn
  - + Cảm biến (nhiệt độ, độ ẩm, ...)
- Đặt lịch hẹn giờ
- Tạo các kịch bản chạy tự động liên kết giữa các thiết bị Zigbee và Ble. Vd: các nút nhấn hoặc cảm biến Zigbee có thể điều khiển đèn Ble



Sơ đồ kết nối:



## B. CẤU HÌNH

Sử dụng UDP để gửi thông tin cấu hình MQTT cho Hub connect với server hoặc DQ\_Gateway

- Broker: địa chỉ máy chủ MQTT
- User name
- User password

Cấu hình Hub bằng cách gửi các Message qua giao thức UDP đến địa chỉ IP của Hub qua các bước bên dưới:

Port sử dụng cho UDP là 10100

(Có thể dùng bất kì ứng dụng nào có hỗ trợ UDP để gửi lệnh, hoặc sử dụng mobile app của Điện Quang)

**Bước 1:** cấp nguồn cho Hub và mạng Internet qua cổng LAN

**Bước 2:** chờ đèn khi Hub khởi động xong. Hub khởi động xong sẽ có led đỏ nhấp nháy hoặc sáng liên tục

- Nếu led nhấp nháy thì Hub đã ở trạng thái reset, lúc này có thể cấu hình được ngay.
- Nếu led sáng liên tục thì Hub đã được cấu hình, trường hợp này cần reset lại hub bằng cách nhấn giữ nút reset đến khi đèn nhấp nháy thả ra. Chờ Hub khởi động lại

**Bước 3:** ở trạng thái reset Hub sẽ gửi thông tin qua tin nhắn broadcast UDP 3 giây 1 lần và gửi trong 5 phút để APP có thông tin cấu hình Hub

*Vd message Hub gửi như sau:*

```
{
  "method": "new_device",
  "params": {
    "ip": "192.168.1.2",
    "mac": "01:02:03:04:05:06",
    "num": 0,
    "version": "1.0.2",
    "name": "DQ_hub_00",
    "profile": "DQ_hub"
  }
}
```

- Ip: địa chỉ ip của hub
- Mac: địa chỉ mac của hub
- Num: số lần gửi tin nhắn, số này sẽ tăng sau mỗi lần hub gửi tin nhắn đi
- Version: version của hub
- Name: tên hub, tên mặc định hub chưa cấu hình là DQ\_hub\_00 lúc cấu hình cần đặt tên khác

**Bước 4:** gửi thông tin cấu hình Hub

*Vd message gửi đến Hub:*

```
{
  "method": "set_config_hub",
```

```

"params": {
  "name": "new_name_hub",
  "type_hub": "BLE_ZIGBEE_HUB",
  "host": "example.com",
  "type_host": "server",
  "userName": "abcd",
  "userPassWord": "123",
  "id": "abc123",
  "port_host": "1883"
}
}

```

- name: tên cấu hình cho hub
- type\_host: hub connect với gateway Điện Quang hoặc server (“DQ\_Gateway” hoặc “server”)
- type\_hub: loại hub có BLE hoặc có cả BLE và Zigbee (“BLE\_HUB” hoặc “BLE\_ZIGBEE\_HUB”)
- host: địa chỉ broker, nếu connect đến server là địa chỉ của server, nếu connect đến Gateway là địa chỉ ip của gateway
- userName: user name để connect mqtt, nếu mqtt broker không cấu hình thì chỗ này để trống (“userName “:””)
- userPassWord: password để connect mqtt tới broker, nếu broker không cấu hình thì để trống
- id: ID cấu hình hub. ID này sử dụng để thao tác với Hub trong các lệnh dưới nếu gửi đúng ID hub mới nhận và phản hồi.
- port\_host: mặc định là 1883

#### **message Hub phản hồi:**

```

{
  "method": "set_config_hub_feedback",
  "params": {
    "status": "00",
  }
}

```

status: trạng thái cấu hình

- “00”: trạng thái cấu hình thành công
- “01”: params cấu hình không đúng

**Bước 5:** kiểm tra lại thông tin đã cấu hình cho hub đúng chưa bằng lệnh get config:

#### **Message gửi đến Hub:**

```

{
  "method": "get_config_hub",
  "params": {
    "id": ""
  }
}

```



```
}  
}
```

- id: id đã cấu hình trong lệnh set\_config\_hub

**Message Hub phản hồi:**

```
{  
  "method": "get_config_hub_feedback",  
  "params": {  
    "status": "00",  
    "name": "hub_pro_01",  
    "type_host": "DQ_Gateway" | "server",  
    "type_hub": "BLE_HUB" | "BLE_ZIGBEE_HUB",  
    "host": "example.com",  
    "userName": "abc",  
    "userPassWord": "123",  
    "version": "1.0.0",  
    "id": "",  
    "port_host": "1883"  
  }  
}
```

status: trạng thái get

- “00”: trạng thái thành công
- “01”: params gửi không đúng
- “02”: userId không đúng
- “03”: hub chưa cấu hình

**Bước 6:** Sau khi cấu hình thành công cần restart lại hub

**Message gửi đến Hub:**

```
{  
  "method": "reboot_hub",  
  "params": {  
    "id": ""  
  }  
}
```

**Message Hub phản hồi:**

```
{  
  "method": "reboot_hub_feedback",  
  "params": {
```

```
"status": "00",  
}  
}
```

Sau khi cấu hình thành công nếu muốn đưa Hub về trạng thái chưa cấu hình bằng lệnh:  
**Message gửi đến Hub:**

```
{  
  "method": "clear_hub",  
  "params": {  
    "id": ""  
  }  
}
```

**Message Hub phản hồi:**

```
{  
  "method": "clear_hub_feedback",  
  "params": {  
    "status": "00",  
  }  
}
```

## C. TÍNH NĂNG

Sau khi connect thành công với server Hub gửi thông tin lên server 1 phút/lần  
Hub gửi qua topic: v1/devices/me/telemetry  
Payload: ví dụ thông tin của hub như sau

```
{  
  "infor": {  
    "ip": "172.16.26.112",  
    "mac": "02:81:71:87:6d:f4",  
    "id": "abc123",  
    "version": "1.0.3",  
    "name": "hub test 1"  
  }  
}
```

### I. Version Hub

#### 1. Get version Hub

Lệnh này để get version hiện tại của hub  
**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

\$request\_id: số nguyên để nhận dạng giữa các lệnh gửi và phản hồi, lệnh gửi và phản hồi sẽ có cùng id

Payload:

```
{
  "method": "get_version",
  "params": {}
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "get_version_status",
  "params": {
    "version": "1.0.0"
  }
}
```

## 2. Update FirmWare Hub

Lệnh này để update FirmWare mới cho Hub

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

\$request\_id: số nguyên để nhận dạng giữa các lệnh gửi và phản hồi, lệnh gửi và phản hồi sẽ có cùng id

Payload:

```
{
  "method": "update_firmware",
  "params": {
    "host": "172.16.23.52:8080",
    "title": "hub",
    "version": "1.0.1"
  }
}
```

Chú thích:

- Host: địa chỉ host để cập nhật firmware
- Title: title firmware
- Version: version cần cập nhật ( có thể lớn hơn hoặc nhỏ hơn so với version hiện tại)

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "update_firmware_status",
  "params": {
    "status": "00",
    "version": "1.0.1"
  }
}
```

status: trạng thái phản hồi

- "00": thành công
- "ERROR": lỗi, thông tin lỗi trong message

Ví dụ:

- Lỗi không download được FirmWare

```
{
  "method": "update_firmware_status",
  "params": {
    "status": "ERROR",
    "message": "400"
  }
}
```

## II. BLE SigMesh

Các lệnh dưới đây giao tiếp giữa server và Hub để điều khiển các thiết bị Ble SigMesh thông qua giao thức MQTT.

Để điều khiển đèn đối với Hub mới setup lần đầu cần thực hiện các bước sau:

- Cập nhật file json cho hub (file data của mạng BLE SigMesh) file này nhận từ bộ phận cài đặt cấu hình mạng BLE SigMesh
- Set địa chỉ address cho Hub (địa chỉ này dạng Hex String có giá trị từ 7777-7FFF) mỗi lần setup lại Hub cần tăng số này lên
- Get danh sách thiết bị hub quản lý
- Get danh sách Group hub quản lý
- Điều khiển đèn

Hub setup rồi chỉ cần gửi lệnh điều khiển

**Để gửi lệnh điều khiển đến Hub, server gửi public đến topic:**

`v1/devices/me/rpc/request/$request_id`

\$request\_id: số nguyên để nhận dạng giữa các lệnh gửi và phản hồi, lệnh gửi và phản hồi sẽ có cùng id

**Để nhận phản hồi từ Hub, server cần Subscribe vào topic:**

`v1/devices/me/rpc/response/$request_id`

### 1. Cập nhật mới file json từ server

Lệnh này để hub cập nhật file json từ server

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

\$request\_id: số nguyên để nhận dạng giữa các lệnh gửi và phản hồi, lệnh gửi và phản hồi sẽ có cùng id

Payload:

```
{
  "method": "updateData",
  "params": {
    "type": "ble_sigmesh",
    "data_bleSigmesh": {}
  }
}
```

data\_bleSigmesh: nội dung file json

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "updateData_status",
  "params": {
    "type": "ble_sigmesh",
    "status": "00"
  }
}
```

## 2. Set địa chỉ unicast hub

Lệnh này để set network cho hub với địa chỉ unicast vừa set, netkey và appkey mặc định là đầu tiên

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "configUnicast",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "199B"
  }
}
```

Chú thích:

- method: "configUnicast"
- params:

- + type: "ble\_sigmoid"
- + unicast: địa chỉ unicast cần set cho Hub

### **Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "configUnicast_status",
  "params": {
    "type": "ble_sigmoid",
    "unicast": "199B"
  }
}
```

## **3. Get địa chỉ unicast Hub**

Lệnh này để đọc về địa chỉ unicast đã set cho Hub

### **Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "getLocalUnicast",
  "params": {
    "type": "ble_sigmoid"
  }
}
```

**Hub feedback to server:** ví dụ

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "getLocalUnicast",
  "params": {
    "status": "00",
    "unicast": "199B"
  }
}
```

## **4. Get danh sách thiết bị hub quản lý**

Lệnh này để lấy danh sách các thiết bị mà hub quản lý tối đa 120 ((thiết bị x số element) + group)

### **Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "get_list_device",
  "params": {
    "type": "ble_sigmesh"
  }
}
```

**Hub feedback to server:**

Ví dụ trả về thông tin của 1 device

Topic: v1/devices/me/rpc/response/[\\$request\\_id](#)

Payload:

```
{
  "method": "list_device_status",
  "params": {
    "type": "ble_sigmesh",
    "nodes": [
      {
        "appKeys": [
          {
            "index": 0,
            "updated": false
          }
        ],
        "cid": "8001",
        "configComplete": false,
        "crpl": "0064",
        "defaultTTL": 15,
        "deviceKey": "35C0E6A5D7BC40498B46DD130C677CA2",
        "elements": [
          {
            "index": 0,
            "location": "0000",
            "models": [
              {
                "bind": [
                  0
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```

        ],
        "modelId": "1000",
        "subscribe": [
            "C000"
        ]
    }
]
}
},
"excluded": false,
"features": {
    "friend": 2,
    "lowPower": 2,
    "proxy": 0,
    "relay": 2
},
"name": "TUBE V1.4",
"netKeys": [
    {
        "index": 0,
        "updated": false
    }
],
"pid": "1098",
"security": "secure",
"unicastAddress": "0004",
"UUID": "01807190-2600-008C-AEB1-514719020000",
"vid": "0001"
}
]
}
}

```

- Name: tên thiết bị
- Pid: nhận dạng loại thiết bị (đèn, cảm biến,...).  
Danh sách PID vui lòng xem file đính kèm
- unicastAddress: địa chỉ của thiết bị được lưu trong mạng



điều khiển thiết bị chỉ cần 3 thông tin trên

## 5. Get danh sách Group hub quản lý

Lệnh này để lấy danh sách các group mà hub quản lý tối đa 120 [(thiết bị)x(số element) + group]

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "get_list_groups",
  "params": {
    "type": "ble_sigmesh"
  }
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "list_groups_status",
  "params": {
    "type": "ble_sigmesh",
    "groups": [
      {
        "address": "C000",
        "name": "New Group",
        "parentAddress": "0000"
      }
    ]
  }
}
```

## 6. Get danh sách Scene

Lệnh này để lấy danh sách các scene hub

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "get_list_scenes",
  "params": {
    "type": "ble_sigmesh"
  }
}
```

```
}
```

**Hub feedback to server:** ví dụ hub trả về có 3 scene

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "list_scenes_status",
  "params": {
    "type": "ble_sigmesh",
    "nodes": [
      {
        "addresses": ["0005"],
        "name": "Scene 1",
        "number": "0001"
      },
      {
        "addresses": [],
        "name": "Scene 2",
        "number": "0002"
      },
      {
        "addresses": [],
        "name": "Scene 3",
        "number": "0003"
      }
    ]
  }
}
```

## 7. Set On/Off thiết bị hoặc group

Lệnh này set on off cho từng thiết bị hoặc 1 nhóm thiết bị

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_onOff",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0004",
```

```
"value": 0
}
```

Chú thích:

- method: "set\_onOff"
- params:
  - + type: "ble\_sigmesh"
  - + unicast: địa chỉ unicast của thiết bị hoặc group cần điều khiển
  - + value: trạng thái cần điều khiển on = 1 hoặc off = 0

### ***Hub feedback to server:***

Topic: v1/devices/me/rpc/response/[\\$request\\_id](#)

Payload:

```
{
  "method": "set_onOff_status",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0004",
    "value": 0
  }
}
```

## **8. Get On/Off thiết bị hoặc group**

Lệnh này get trạng thái on off cho từng thiết bị hoặc 1 nhóm thiết bị

### ***Data send to Hub:***

Topic: v1/devices/me/rpc/request/[\\$request\\_id](#)

Payload:

```
{
  "method": "get_onOff",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0004",
    "value": 0
  }
}
```

Chú thích:

- method: "get\_onOff"
- params:
  - + type: "ble\_sigmesh"
  - + unicast: địa chỉ unicast của thiết bị hoặc group cần kiểm tra
  - + value: trạng thái cần điều khiển on = 1 hoặc off = 0

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "get_onOff_status",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0004",
    "value": 0
  }
}
```

**9. Set Lightness thiết bị hoặc group**

Lệnh này set lightness cho từng thiết bị hoặc 1 nhóm thiết bị

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_lightness",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "C000",
    "value": 50
  }
}
```

Chú thích:

- method: "set\_lightness"
- params:
  - + type: "ble\_sigmesh"
  - + unicast: địa chỉ unicast của thiết bị hoặc group cần điều khiển
  - + value: độ sáng cần điều khiển 0->100%

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "set_lightness_status",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0004",
```

```
"value": 50
}
```

## 10. Get Lightness thiết bị hoặc group

Lệnh này get lightness cho từng thiết bị hoặc 1 nhóm thiết bị

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```
{
  "method": "get_lightness",
  "params": {
    "type": "ble_sigmoid",
    "unicast": "0004"
  }
}
```

Chú thích:

- method: "get\_lightness"
- params:
  - + type: "ble\_sigmoid"
  - + unicast: địa chỉ unicast của thiết bị hoặc group cần kiểm tra

**Hub feedback to server:**

Topic: `v1/devices/me/rpc/response/$request_id`

Payload:

```
{
  "method": "get_lightness_status",
  "params": {
    "type": "ble_sigmoid",
    "unicast": "0004"
  }
}
```

## 11. Set HSL

Lệnh này set HSL cho từng thiết bị hoặc 1 nhóm thiết bị

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```
{
  "method": "set_HSL",
  "params": {
```

```
"type": "ble_sigmoid",
"unicast": "0005",
"Lightness": 100,
"Hue": 360,
"Saturation": 100
}
```

Chú thích:

- Unicast: địa chỉ của đèn
- Lightness: giá trị value 0->100
- Hue: giá trị hue 0->360
- Saturation: giá trị Saturation 0->100

### **Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "set_HSL_status",
  "params": {
    "type": "ble_sigmoid",
    "unicast": "0005",
    "Lightness": 100,
    "Hue": 360,
    "Saturation": 100
  }
}
```

## **12. Call Scene**

Lệnh này gọi ngữ cảnh đã set (danh sách ngữ cảnh có thể đọc về từ hub qua lệnh get scene)

### **Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "scene_recall",
  "params": {
    "type": "ble_sigmoid",
    "unicast": "0005",
    "SceneNumber": 1
  }
}
```

```
}
```

Chú thích:

- Unicast: địa chỉ thiết bị
- SceneNumber: số thứ tự scene cần điều khiển

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "scene_recall_status",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "0005",
    "SceneNumber": 1
  }
}
```

### 13. Set vendor model

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_vendor",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "026C",
    "opcode": "EA",
    "company_id": "8001",
    "message": ""
  }
}
```

Chú thích:

- Unicast: địa chỉ thiết bị
- Opcode: opcode vendor cần set
- Company\_id: company id của thiết bị
- Message: data cần set

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
```

```
"method": "vendor_status",
"params": {
  "address": "026C",
  "company_id": "8001",
  "opcode": "ED",
  "message": "000000000000"
}
```

Chú thích:

- Unicast: địa chỉ thiết bị
- Opcode: opcode vendor cần set
- Company\_id: company id của thiết bị
- Message: data thiết bị trả về

## 14. Reboot Hub

Lệnh này khởi động lại hub

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "reboot",
  "params": {
    "type": "ble_sigmesh"
  }
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "reboot_status",
  "params": {
    "type": "ble_sigmesh"
  }
}
```

## 15. Get status devices

Đọc về trạng thái của tất cả đèn (onOff, Lightness, HSL)

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:



```
{
  "method": "get_status_devices",
  "params": {
    "type": "ble_sigmoid",
    "address": ["0006","0080"]
  }
}
```

Chú thích:

- Address: danh sách địa chỉ thiết bị cần lấy trạng thái. Nếu muốn lấy tất cả thiết bị thì "address":[]

Ví dụ trên get trạng thái của 2 đèn địa chỉ “0006” và “0080”

**Hub feedback to server:**

Ví dụ phản hồi trạng thái của 2 thiết bị

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "status_devices",
  "params": {
    "type": "ble_sigmoid",
    "data_bleSigmoid": {
      "0006": {
        "onOffStatus": 1,
        "time": 1681526584.1812994,
        "status": true,
        "lightnessStatus": 80,
        "HSL_status": {
          "Lightness": 160,
          "Hue": 351,
          "Saturation": 100
        }
      },
      "0070": {
        "onOffStatus": 1,
        "time": 1681526561.7739537,
        "status": true,
        "lightnessStatus": 50,
        "HSL_status": {
```

```
        "Lightness": 100,  
        "Hue": 351,  
        "Saturation": 100  
    }  
}  
}  
}
```

Chú thích:

- Time: thời gian nhận trạng thái cuối cùng của thiết bị
- Status: trạng thái đèn có đang kết nối vào mạng không. Nếu >5 phút mà không nhận được tín hiệu từ thiết bị thì status là False (đèn mất kết nối) ngược lại là True
- onOffStatus: trạng thái on/off đèn
- lightnessStatus: lightness của đèn
- HSL\_status: giá trị HSL của đèn

## 16. Trạng thái thiết bị trả về

Để lắng nghe các trạng thái thiết bị trả về server cần subscribe Topic:

v1/devices/me/telemetry

### Cập nhật On/Off

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{  
  "data_bleSigmesh": [  
    {  
      "method": "onOffStatus",  
      "params": {  
        "address": "0004",  
        "value": 1  
      }  
    },  
    {  
      "method": "onOffStatus",  
      "params": {  
        "address": "0003",  
        "value": 1  
      }  
    }  
  ]  
}
```

```
]
}
```

- Address: địa chỉ đèn
- Value: 1 hoặc 0 tương ứng với ON hoặc OFF

Ví dụ trên trả về trạng thái của 2 đèn có địa chỉ là 0003, 0004

**Cập nhật Lightness:**

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{
  "data_bleSigmesh": [
    {
      "method": "lightnessStatus",
      "params": {
        "address": "0004",
        "value": 100
      }
    }
  ]
}
```

**Cập nhật HSL:**

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{
  "data_bleSigmesh": [
    {
      "method": "HSL_status",
      "params": {
        "address": "0005",
        "Lightness": 100,
        "Hue": 120,
        "Saturation": 100
      }
    }
  ]
}
```

**Cập nhật Scene:**

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{
  "data_bleSigmesh": [
    {
      "method": "scene_status",
      "params": {
        "status": "00",
        "currentScene": 1,
        "address": "0005"
      }
    }
  ]
}
```

**Cập nhật Vendor:**

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{
  "data_bleSigmesh": [
    {
      "method": "vendor_status",
      "params": {
        "address": "026C",
        "company_id": "8001",
        "opcode": "ED",
        "message": "000000000000"
      }
    }
  ]
}
```

**Cập nhật danh sách thiết bị Disconnect:**

Hub send to server:

Topic: v1/devices/me/telemetry

Payload:

```
{
```

```
"data_bleSigmesh": [  
  {  
    "method": "disconnectedEvent",  
    "params": {  
      "type": "ble_sigmesh",  
      "address": [  
        "0040",  
        "0022"  
      ]  
    }  
  }  
]
```

### III. Zigbee

#### 1. Delete network

Lệnh này xóa mạng zigbee hiện tại

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{  
  "method": "delete_network",  
  "params": {  
    "type": "zigbee"  
  }  
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{  
  "method": "delete_network_status",  
  "params": {  
    "type": "zigbee"  
  }  
}
```

## 2. Init network

Lệnh này tạo mới mạng zigbee

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_network",
  "params": {
    "type": "zigbee"
  }
}
```

**Lưu ý:** lệnh này chờ hub phản hồi nên để timeout tối thiểu 10s

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "set_network_status",
  "params": {
    "type": "zigbee",
    "status": "00"
  }
}
```

## 3. Bật cho phép pair thiết bị mới

Lệnh này cho phép hub pair thiết bị mới

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "start_permit_join",
  "params": {
    "type": "zigbee"
  }
}
```

**Hub feedback to server:** nếu pair thành công hub phản hồi như sau

**Lưu ý:** lệnh này cần thời gian chờ hub pair xong thiết bị (khoảng 15s)

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
```

```
"method": "add_new_device",
"params": {
  "type": "zigbee",
  "addr": 36368,
  "name": "lumi.sensor_wleak.aq1",
  "model": "lumi.sensor_wleak.aq1"
}
```

#### 4. Tắt cho phép pair thiết bị mới

Lệnh này không cho phép hub thêm thiết bị mới

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "stop_permit_join",
  "params": {
    "type": "zigbee"
  }
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "stop_permit_join_status",
  "params": {
    "type": "zigbee"
  }
}
```

#### 5. Set model cho thiết bị

Lệnh này set model cho thiết bị, nếu thiết bị không tự trả về model

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_model",
  "params": {
    "type": "zigbee",
    "name": "sensor",

```

```
"address": 12500,  
"model": "lumi.weather"  
}  
}
```

Chú thích:

- Name: tên đặt cho thiết bị
- Address: địa chỉ của thiết bị trong mạng
- Model: model cần set

Danh sách các model:

```
{  
  "dqhome.re1": "DQSmart Switch 1 Relay",  
  "dqhome.re2": "DQSmart Switch 2 Relay",  
  "dqhome.re3": "DQSmart Switch 3 Relay",  
  "dqhome.re4": "DQSmart Switch 4 Relay",  
  "dqhome.re6": "DQSmart Switch 6 Relay",  
  "dqhome.cur": "DQSmart curtain sensor",  
  "lumi.sensor_cube": "Mi Cube Controller",  
  "lumi.sensor_magnet": "Xiaomi Door and Window Sensor",  
  "lumi.sensor_magnet.aq2": "Aqara Door and Window Sensor",  
  "lumi.sensor_motion": "Xiaomi Motion Sensor",  
  "lumi.sensor_motion.aq2": "Aqara Motion Sensor",  
  "lumi.light.aqcn02": "Aqara Smart Bulb",  
  "lumi.plug": "Xiaomi Smart Plug",  
  "lumi.ctrl_86plug.aq1": "Aqara Smart Plug",  
  "lumi.sensor_ht": "Xiaomi Temperature and Humidity Sensor",  
  "lumi.weather": "Aqara Temperature and Humidity Sensor",  
  "lumi.relay.c2acn01": "Aqara Two-Way Wireless Control Relay",  
  "lumi.ctrl_neutral1": "Aqara Wall Switch - Single (no  
Neutral)",  
  "lumi.ctrl_neutral2": "Aqara Wall Switch - Double (no  
Neutral)",  
  "lumi.ctrl_ln1.aq1": "Aqara Wall Switch - Single (w/Neutral)",  
  "lumi.ctrl_ln2.aq1": "Aqara Wall Switch - Double (w/Neutral)",  
  "lumi.sensor_wleak.aq1": "Aqara Water Leak Sensor",  
  "lumi.sensor_switch": "Xiaomi Smart Wireless Switch",  
  "lumi.sensor_switch.aq2": "Aqara Wireless Mini Switch (2015)",  
}
```



```
"lumi.remote.b1acn01": "Aqara Wireless Mini Switch (2018)",
"lumi.sensor_switch.aq3": "Aqara Wireless Mini Switch",
"lumi.sensor_86sw1": "Aqara Wireless Remote Switch - Single",
"lumi.sensor_86sw1lu": "Aqara Wireless Remote Switch - Single",
"lumi.remote.b186acn01": "Aqara Wireless Remote Switch - Single",
"lumi.sensor_86sw2": "Aqara Wireless Remote Switch - Double",
"lumi.sensor_86sw2Un": "Aqara Wireless Remote Switch - Double",
"lumi.remote.b286acn01": "Aqara Wireless Remote Switch - Double",
"lumi.vibration.aq1": "Aqara Vibration Sensor",
"lumi.sensor_smoke": "MiJia Honeywell Smoke Detector",
"lumi.sensor_natgas": "MiJia Honeywell Natural Gas Sensor",
"lumi.gas": "MiJia Honeywell Natural Gas Sensor",
"lumi.curtain": "Aqara Smart Curtain"
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/[\\$request\\_id](#)

Payload:

```
{
  "method": "set_model_status",
  "params": {
    "type": "zigbee",
    "name": "sensor",
    "addr": 12500,
    "model": "lumi.weather"
  }
}
```

## 6. Delete thiết bị ra khỏi mạng

Lệnh này xóa 1 thiết bị ra khỏi mạng

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/[\\$request\\_id](#)

Payload:

```
{
  "method": "device_remove",
```

```
"params": {  
  "type": "zigbee",  
  "address": 12500  
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{  
  "method": "device_remove_status",  
  "params": {  
    "type": "zigbee",  
    "addr": 123,  
    "status": "00"  
  }  
}
```

Chú thích:

- Addr: địa chỉ thiết bị
- Status:
  - + 00: thành công
  - + 8E: lỗi (có thể địa chỉ không đúng hoặc không tồn tại)

## 7. Set On off

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{  
  "method": "set_onOff",  
  "params": {  
    "type": "zigbee",  
    "address": 12308,  
    "DstEndPoint": 2,  
    "SrcEndPoint": 1,  
    "value": 0  
  }  
}
```

Chú thích:

- Address: địa chỉ của thiết bị
- SrcEndPoint: endpoint của hub mặc định = 1

- DstEndPoint: endpoint cần điều khiển của thiết bị
- Value: trạng thái cần điều khiển on/off tương ứng 1/0

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "set_onOff_status",
  "params": {
    "type": "zigbee",
    "address": 12308,
    "DstEndPoint": 2,
    "SrcEndPoint": 1,
    "value": 0
  }
}
```

## 8. Set Lightness

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_lightness",
  "params": {
    "type": "zigbee",
    "address": 12308,
    "DstEndPoint": 2,
    "SrcEndPoint": 1,
    "value": 50
  }
}
```

Chú thích:

- Address: địa chỉ của thiết bị
- SrcEndPoint: endpoint của hub mặc định = 1
- DstEndPoint: endpoint cần điều khiển của thiết bị
- Value: giá trị Lightness cần điều khiển 0->100%

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
```

```
"method": "set_lightness_status",
"params": {
  "type": "zigbee",
  "address": 12308,
  "DstEndPoint": 2,
  "SrcEndPoint": 1,
  "value": 50
}
```

## 9. Set bind cho thiết bị

Để cập nhật trạng thái đèn cần set bind cho đèn:

- Address: địa chỉ đèn
- clusterId: cluster id cần bind, vd: bind on/off cluster id = 6

```
clusterList = {
  0x0000: " (General: Basic)",
  0x0001: " (General: Power Config)",
  0x0002: " (General: Temperature Config)",
  0x0003: " (General: Identify)",
  0x0004: " (General: Groups)",
  0x0005: " (General: Scenes)",
  0x0006: " (General: On/Off)",
  0x0007: " (General: On/Off Config)",
  0x0008: " (General: Level Control)",
  0x0009: " (General: Alarms)",
  0x000A: " (General: Time)",
  0x000F: " (General: Binary Input Basic)",
  0x0012: " (General: Multistate Input)",
  0x0020: " (General: Poll Control)",
  0x0019: " (General: OTA)",
  0x0101: " (General: Door Lock)",
  0x0201: " (HVAC: Thermostat)",
  0x0202: " (HVAC: Fan Control)",
  0x0204: " (HVAC: Thermostat UI Config)",
  0x0300: " (Lighting: Color Control)",
  0x0400: " (Measurement: Illuminance)",
```

```
0x0402: " (Measurement: Temperature)",
0x0405: " (Measurement: Relative Humidity)",
0x0406: " (Measurement: Occupancy Sensing)",
0x0500: " (Security & Safety: IAS Zone)",
0x0501: " (Security & Safety: IAS Ancillary Control Equipme
nt)",
0x0502: " (Security & Safety: IAS WD)",
0x0702: " (Smart Energy: Metering)",
0x0B05: " (Misc: Diagnostics)",
0x1000: " (ZLL: Commissioning)"}

```

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{
  "method": "set_bind",
  "params": {
    "type": "zigbee",
    "address": 12308,
    "DstEndPoint": 1,
    "TargetEndPoint": 2,
    "clusterId": 6
  }
}
```

Chú thích:

- address: địa chỉ thiết bị
- TargetEndPoint: endpoint của thiết bị
- DstEndPoint: endpoint của hub
- clusterId: giá trị cluster trong list trên

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "set_bind_status",
  "params": {
    "type": "zigbee",
    "addr": 12308,
    "DstEndPoint": 1,

```

```
"TargetEndPoint": 2,  
"status": "00"  
}  
}
```

## 10. Get list devices

*Data send to Hub:*

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{  
  "method": "get_list_device",  
  "params": {  
    "type": "zigbee"  
  }  
}
```

*Hub feedback to server:*

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{  
  "method": "list_device_status",  
  "params": {  
    "62986": {  
      "name": "CB1 LAU 4 TM",  
      "addr": 62986,  
      "model": "dqhome.re1"  
    },  
    "62989": {  
      "name": "CB2 LAU 4 TM",  
      "addr": 62989,  
      "model": "dqhome.re1"  
    }  
  }  
}
```

## 11. Status device telemetry

Cập nhật trạng thái thiết bị tự động từ hub lên server lưu ở telemetry

```
{  
  "data_zigbee": [  
    {
```

```
"method": "lumi.weather",
"params": {
  "addr": 15439,
  "ep": 1,
  "state": {
    "name": "Relative Humidity",
    "val": "79", "unit": "%"
  }
},
{
  "method": "lumi.weather",
  "params": {
    "addr": 15439,
    "ep": 1,
    "states": {
      "name": "Temperature",
      "val": "27",
      "unit": "\u00b0C"
    }
  }
},
{
  "method": "lumi.weather",
  "params": {
    "addr": 24224,
    "ep": 1,
    "states": {"name": "Status", "val": 1}
  }
}
]
```

Chú thích:

- Method: model của thiết bị
- Params:
  - + Addr: địa chỉ thiết bị

- + Ep: end point (ví dụ công tắc 4 nút nhấn thì ep 1->4 tương đương nút nhấn 1->4)
- + States: trạng thái
  - Name: loại status
    - Relative Humidity: độ ẩm
    - Temperature: nhiệt độ
    - Status: nút nhấn
    - lightness: lightness đèn 0->100%
    - Battery percent: phần % pin 0->100%
    - Battery voltage: điện áp pin
  - Val: giá trị. Đối với nút nhấn:
    - Val=1 nhấn 1 cái
    - Val=0 nhấn giữ
    - Val = 255 nhả sau khi nhấn giữ
    - Val = 2 nhấn 2 cái
  - Unit: đơn vị

## D. KỊCH BẢN

Setup các kịch bản dùng chung cho BLE và Zigbee hoặc đặt lịch hẹn giờ

### I. Chạy tự động

#### 1. Creat

Tạo mới scene(kịch bản chạy tự động) cho hub

Khi gửi lệnh này hub sẽ tạo mới lại scene theo data đã nhận. (những scene setup trước đó sẽ mất)

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```
{
  "method": "creat_scene_hub",
  "params": [
    {
      "comparison": "less"|"less_equal"|"greater"|"greater_equal"|"equal",
      "enable":true,
      "id": "",
      "input": {
        "method": "status",
        "params": {
          "addr": 9959,
          "state": {"name": "Status", "val": "ON"}
        }
      }
    }
  ]
}
```



```
    },  
    "output": {  
      "method": "set_onOff",  
      "params": {  
        "type": "ble_sigmesh",  
        "unicast": "0004",  
        "value": 1  
      }  
    }  
  }  
]  
}
```

Chú thích:

- enable: trạng thái bật(true) hay tắt (false)
- id: id của scene cần lưu
- input: giá trị đầu vào để điều khiển ngõ ra (đây là giá trị trả về của zigbee)
- output: giá trị cần điều khiển nếu thỏa input (đây là giá trị điều khiển của Ble)
- Comparison: biểu thức so sánh:
  - less: nhỏ hơn
  - less\_equal: nhỏ hơn hoặc bằng
  - greater: lớn hơn
  - greater\_equal: lớn hơn hoặc bằng
  - equal: bằng

### ***Hub feedback to server:***

Topic: v1/devices/me/rpc/response/[\\$request\\_id](#)

Payload:

```
{  
  "method": "creat_scene_hub_status",  
  "params": [  
    {  
      "comparison": "less"|"less_equal"|"greater"|"greater  
_equal"|"equal",  
      "enable":true,  
      "id": "",  
      "input": {  
        "method": "status",  
        "params": {
```

```
        "addr": 9959,  
        "state": {"name": "Status", "val": "ON"}  
    },  
    },  
    "output": {  
        "method": "set_onOff",  
        "params": {  
            "type": "ble_sigmoid",  
            "unicast": "0004",  
            "value": 1  
        }  
    }  
}  
]  
}
```

## 2. Update

Lệnh này để thêm 1 hoặc nhiều Scene mới cho Hub (những scene đã setup trước đó không bị mất)

**Data send to Hub:**

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

```
{  
    "method": "update_scene_hub",  
    "params": [  
        {  
            "comparison": "less"|"less_equal"|"greater"|"greater_equal"|"equal",  
            "enable": true,  
            "id": "",  
            "input": {  
                "method": "status",  
                "params": {  
                    "addr": 9959,  
                    "state": {"name": "Status", "val": "ON"}  
                }  
            }  
        }  
    ],  
}
```

```
    "output": {
      "method": "set_onOff",
      "params": {
        "type": "ble_sigmesh",
        "unicast": "0004",
        "value": 1
      }
    }
  }
}
```

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "update_scene_hub_status",
  "params": [
    {
      "comparison": "less"|"less_equal"|"greater"|"greater_equal"|"equal",
      "enable": true,
      "id": "",
      "input": {
        "method": "status",
        "params": {
          "addr": 9959,
          "state": {"name": "Status", "val": "ON"}
        }
      },
      "output": {
        "method": "set_onOff",
        "params": {
          "type": "ble_sigmesh",
          "unicast": "0004",
          "value": 1
        }
      }
    }
  ]
}
```

```
}  
}  
]  
}
```

### 3. Get

Đọc về danh sách đã set

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```
{  
  "method": "get_scene_hub",  
  "params": ""  
}
```

**Hub feedback to server:**

Topic: `v1/devices/me/rpc/response/$request_id`

Payload:

```
{  
  "method": "scene_hub",  
  "params": [  
    {  
      "comparison": "less"|"less_equal"|"greater"|"greater  
_equal"|"equal",  
      "id": "1",  
      "enable": true,  
      "input": {  
        "method": "lumi.remote.b1acn01",  
        "params": {  
          "addr": 24224,  
          "ep": 1,  
          "state": {"name": "Status", "val": 1}  
        }  
      },  
      "output": {  
        "method": "set_onOff",  
        "params": {  
          "type": "ble_sigmesh",  
          "unicast": "0004",  
        }  
      }  
    }  
  ]  
}
```

```

        "value": 1
      }
    }
  ]
}

```

#### 4. Delete

Xóa 1 hoặc nhiều scene

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```

{
  "method": "delete_scene_hub",
  "params": ["id1", "id2"]
}

```

Chú thích:

- "params": ["id1", "id2"]: "id1", "id2" là các id của scene cần xóa, Nếu muốn xóa tất cả ở đây để mảng trống

Vd xóa tất cả scene gateway: "params": []

**Hub feedback to server:**

Topic: `v1/devices/me/rpc/response/$request_id`

Payload:

```

{
  "method": "delete_scene_hub_status",
  "params": ["id1", "id2"]
}

```

## II. Schedule

### 1. Creat

Tạo mới schedule cho hub

Khi gửi lệnh này hub sẽ tạo mới lại schedule theo data đã nhận. (những schedule setup trước đó sẽ mất)

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```

{
  "method": "creat_schedule_hub",
  "params": [
    {

```

```
"enable": true,
"id": "",
"input": {
  "times": "00:00",
  "weekdays": [0, 1, 2, 3, 4, 5, 6]
},
"output": {
  "method": "set_onOff",
  "params": {
    "type": "ble_sigmesh",
    "unicast": "C000",
    "value": 0
  }
}
}
```

Chú thích:

- enable: trạng thái bật(true) hay tắt (false)
- id: id của schedule cần lưu
- input: giá trị đầu vào để điều khiển ngõ ra
- times: thời gian cần set ( theo định dạng 24h)
- weekdays: thứ trong tuần, 0->6 tương ứng thứ 2 -> chủ nhật
- output: giá trị cần điều khiển nếu thỏa input (giá trị điều khiển ble)

### ***Hub feedback to server:***

Topic: v1/devices/me/rpc/response/[\\$request\\_id](#)

Payload:

```
{
  "method": "creat_schedule_hub_status",
  "params": [
    {
      "enable":true,
      "id": "",
      "input": {
        "times": "00:00",
        "weekdays": [0, 1, 2, 3, 4, 5, 6]
      },

```

```
    "output": {
      "method": "set_onOff",
      "params": {
        "type": "ble_sigmesh",
        "unicast": "C000",
        "value": 0
      }
    }
  }
}
```

## 2. Update

Lệnh này để thêm 1 hoặc nhiều Schedule mới cho Hub (những Schedule đã setup trước đó không bị mất)

**Data send to Hub:**

Topic: `v1/devices/me/rpc/request/$request_id`

Payload:

```
{
  "method": "update_schedule_hub",
  "params": [
    {
      "enable": true,
      "id": "",
      "input": {
        "times": "00:00",
        "weekdays": [0, 1, 2, 3, 4, 5, 6]
      },
      "output": {
        "method": "set_onOff",
        "params": {
          "type": "ble_sigmesh",
          "unicast": "C000",
          "value": 0
        }
      }
    }
  ]
}
```

```
]
}
```

Chú thích:

- enable: trạng thái bật(true) hay tắt (false)
- id: id của schedule cần lưu
- input: giá trị đầu vào để điều khiển ngõ ra
- times: thời gian cần set
- weekdays: thứ trong tuần, 0->6 tương ứng thứ 2 -> chủ nhật
- output: giá trị cần điều khiển nếu thỏa input

### ***Hub feedback to server:***

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{
  "method": "update_schedule_hub_status",
  "params": [
    {
      "enable": true,
      "id": "",
      "input": {
        "times": "00:00",
        "weekdays": [0, 1, 2, 3, 4, 5, 6]
      },
      "output": {
        "method": "set_onOff",
        "params": {
          "type": "ble_sigmesh",
          "unicast": "C000",
          "value": 0
        }
      }
    }
  ]
}
```

## **3. Get**

### ***Data send to Hub:***

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:



```
{  
  "method": "get_schedule_hub",  
  "params": ""  
}
```

*Hub feedback to server:*

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{  
  "method": "schedule_hub",  
  "params": [  
    {  
      "enable": true,  
      "id": "off",  
      "input": { "times": "08:45", "weekdays": [0, 1, 2, 3, 4,  
5, 6] },  
      "output": {  
        "method": "set_onOff",  
        "params": { "type": "ble_sigmesh", "unicast": "C000", "  
value": 0 }  
      }  
    },  
    {  
      "enable": true,  
      "id": "on",  
      "input": { "times": "08:43", "weekdays": [0, 1, 2, 3, 4,  
5, 6] },  
      "output": {  
        "method": "set_onOff",  
        "params": { "type": "ble_sigmesh", "unicast": "C000", "  
value": 1 }  
      }  
    }  
  ]  
}
```

#### 4. Delete

*Data send to Hub:*

Topic: v1/devices/me/rpc/request/\$request\_id

Payload:

Xóa 1 hay nhiều schedule

```
{  
  "method": "delete_schedule_hub",  
  "params": ["id1", "id2"]  
}
```

Chú thích:

- "params": ["id1", "id2"]
- + "id1", "id2" là các id của schedule cần xóa, Nếu muốn xóa tất cả ở đây để mảng trống

Vd xóa tất cả schedule: "params": []

**Hub feedback to server:**

Topic: v1/devices/me/rpc/response/\$request\_id

Payload:

```
{  
  "method": "delete_schedule_hub_status",  
  "params": ["id1", "id2"]  
}
```