APP界面数据说明书

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 修订人 | 修订日期 | 修订描述 |
| V 1.0.0 | 石伟丞 | 2017-08-10 | 第一版 |
| V 1.0.0 | 石伟丞 | 2017-09-6 | 1.修改（2.2.1）照明服务详情页面，提供独立接口回去照明服务概要信息 |
|  |  |  |  |
|  |  |  |  |

# 服务主界面

## 能源类设备



图1：主页面-能源服务

1. 发电 ：

<Device.type.id == 040B01000001>.extra.items.toGrid

1. 用电：

<Device.type.id == 040B01000001>.extra.items.toUser

## 照明类设备



图2：主页面-照明服务

1. 开启：

SUM(<Device.type.id IN [040B09050101, 040B09050203, 040B09050113]>.status.switch == “ON”)

1. 关闭：

SUM(<Device.type.id IN [040B09050101, 040B09050203, 040B09050113]>.status.switch == “OFF”)

## 温控类设备



图3：主页面-温控服务

1. 平均室温：AVG(<Device.type.id == 050608070001>.extra.items.dis\_temp)

如果没有该类设备显示:”unkown”

1. 热水温度：<Device.type.id == 040B08080001>.extra.items. dis\_temp)

如果没有该类设备显示:”unkown”

## 开关类设备

# 服务详情页面

## 能源详情页面（简化）

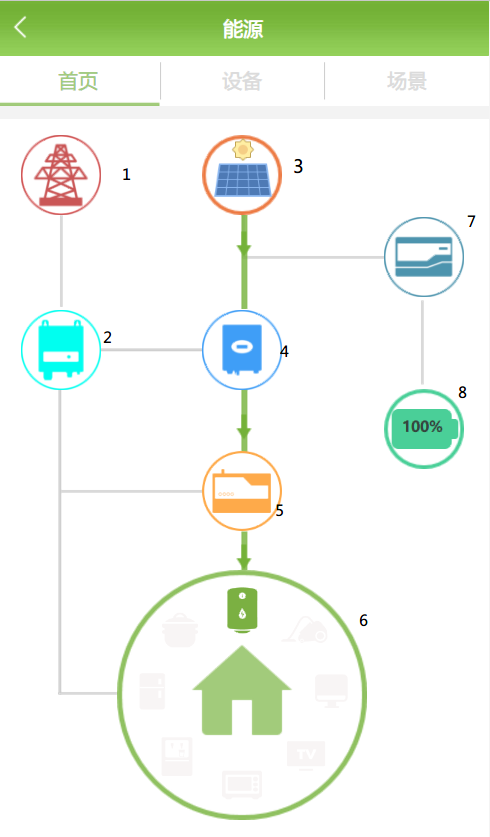


图4：能源导图图标

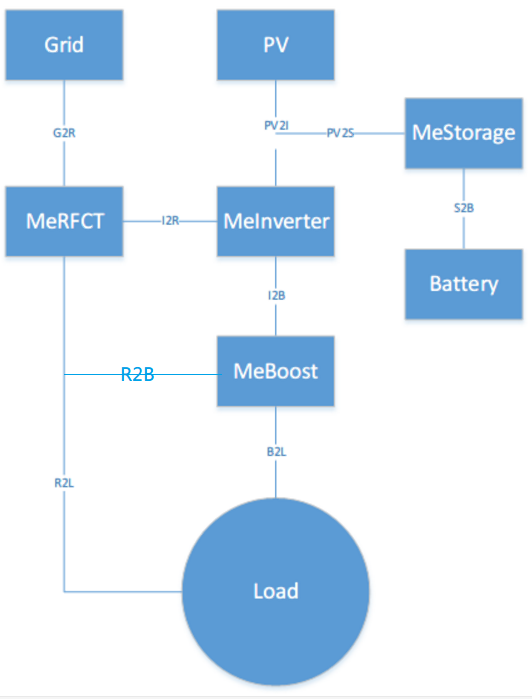


图5：能源导图连线

【能源图谱页面概况接口：/services/energy】

返回数据：

{

"icons":{

"Grip":1, // 1:空闲,2:取电，3：发电

"MeRFCT":1, // 0:无设备,1:设备正常，2：设备异常

"PV":0, // 0:无设备,1:设备正常

"MeInverter":0, // 0:无设备,1:设备正常, 2:设备异常

"MeStroage":0, // 0:无设备,1:设备正常, 2:设备异常

"MeStorageBattery":80, //设备充电百分比

"MeBoost":1, // 0:无设备,1:设备正常, 2:设备异常

"Load":0, // 0:无负载,1:仅热水器负载, 2:仅其他家庭负载，3：全负载

},

"lines":{

"G2R":0, // 0:空闲,1:电网取电，2：太阳能发电

"R2L":0, // 0:空闲,1:电网取电，2：太阳能取电

"PV2I":0, // 0:空闲,1：太阳能发电

"PV2S":0, // 0:空闲,1:太阳能充电，2：放电

"S2B":0, // 0:空闲,1:太阳充电，2：放电

"I2B":0, // 0:空闲,1:太阳能取电，

"R2B":0, // 0:空闲,1:电网取电，

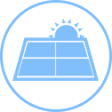
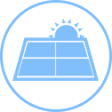
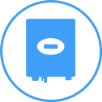
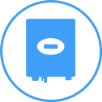
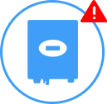
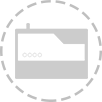
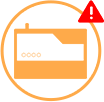
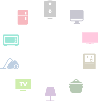
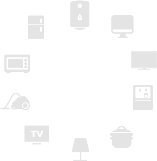
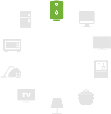
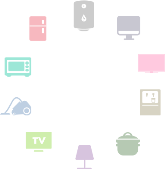
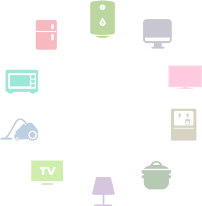
"B2L":0, // 0:空闲,1:电网取电，2：太阳能取电

"I2R":0, // 0:空闲,1：太阳能发电

}

### 主页

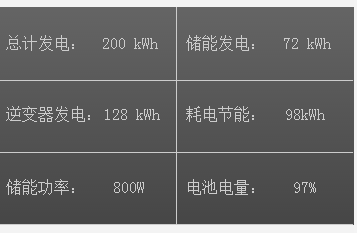
#### 设备图标说明

1. 电网图标：
   1. 0：，空闲状态，既没有取电，也没有发电
   2. 1：取电状态，向电网取电
   3. 2：发电状态，太阳能发电有多余点，输送到电网
2. 逆流传感器（MeRFCT）图标：
   1. 0：，无该设备，用户点击图标提示用户添加MeRFCT。
   2. 1：，设备正常：，用户点击进入设备详情页面。
   3. 2：，设备异常，用户点击提示用户设备异常，检测设备。
3. 太阳能电池板图标：
   1. *0：**，设备不存*在
   2. 1：，设备正常
4. 逆变器（MeInverter）图标：
   1. 0：，设备不存在，用户点击提示用户添加逆变器。
   2. 1：，设备正常，用户点击图标进入设备详情页面。
   3. 2：，设备异常用户点击图标提示用户设备异常，请检测设备。
5. 热水箱控制器（MeBoost）图标：
   1. 0：，设备不存在，用户点击提示用户热水箱控制器。
   2. 1：，设备正常，用户点击图标进入设备详情页面。
   3. 2：设备异常，用户点击图标提示用户设备异常，请检测设备。
6. 家庭负载图标：
   1. 0：（全灰图标），无负载工作
   2. 1：仅热水箱负载工作
   3. 2：其他负载工作
   4. 3：（全彩图标），全负载工作

#### 连线说明

1. G2R：电网到逆流传感器
   1. 0：空闲，灰色连线，无方向
   2. 1：电网取电，红色连线，箭头向下
   3. 2：太阳能发电，绿色连线，箭头向上
2. R2L：逆流传感器到负载
   1. 0：空闲，灰色连线，无方向
   2. 1：电网取电，红色连线，箭头向下
   3. 2：太阳能发电，绿色连线，箭头向上
3. PV2I：太阳能电池板—逆变器
   1. 0：空闲，灰色连线，无方向
   2. 1：太阳能发电，绿色连线，箭头向下
4. PV2S：太阳能电池板—储能机
   1. 0：空闲，灰色连线，无方向
   2. 1：太阳能充电，绿色连线，箭头向右
   3. 2：储能机放电，绿色连线，箭头向左
5. S2B：储能机—蓄电池
   1. 0：空闲，灰色连线，无方向
   2. 1：充电，绿色连线，箭头向下
   3. 2：放电，绿色连线，箭头向上
6. I2B：逆变器—热水控制器
   1. 0：空闲，灰色连线，无方向
   2. 1：太阳能取电，绿色连线，箭头向下
7. R2B：逆流传感器—热水控制器
   1. 0：空闲，灰色连线，无方向
   2. 1：电网取电，红色连线，箭头向右
8. B2L：热水控制器—热水箱
   1. 0：空闲，灰色连线，无方向
   2. 1：电网取电，红色连线，箭头向下
   3. 2：太阳能取电，绿色连线，箭头向下
9. I2R：逆变器—逆流传感器
   1. 0：空闲，灰色连线，无方向
   2. 1：太阳能发电，绿色连线，箭头向左

#### 指标说明



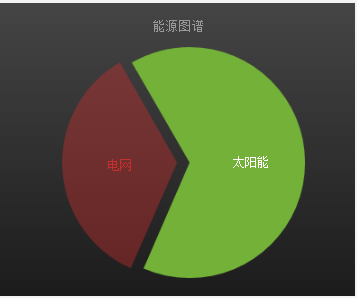
1. 总计发电：MeInverter<Device.type.id == 040B01000004 >.extra.items. totalEnergy +

MeStorage <Device.type.id == 040B01000005 >.extra.items. eChargeTotal

1. 储能发电：MeStorage <Device.type.id == 040B01000005 >.extra.items. eChargeTotal
2. 逆变器发电：MeInverter <Device.type.id == 040B01000004 >.extra.items. totalEnergy
3. 耗电节能：
4. 储能功率：MeStorage <Device.type.id == 040B01000005 >.extra.items. pCharge
5. 电池电量：MeStorage<Device.type.id == 040B01000005 >.extra. capacity.

#### 能源图谱说明

1. 总的能源图谱：太阳能与电网

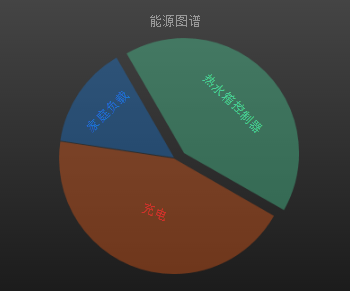


电网：MeRFC< Device.type.id == 040B01000001>.extra. toUser

太阳能：MeInverter <Device.type.id == 040B01000004 >.extra.items. totalEnergy

1. 太阳能能源图谱：太阳能与电网

用户点击太阳能切换到太阳能能源图谱



可能消耗太阳能的设备：

热水箱控制器：MeBoost< Device.type.id == 040B01000005 >extra.items. energySaved

家庭负载：MeInverter< Device.type.id == 040B01000004 >extra.items. totalEnergy - MeBoost< Device.type.id == 040B01000005 >extra.items. energySaved

充电：

### 设备页



能源设备包括：逆变器（040B01000004），储能机（040B01000005），热水器（040B08040004），逆流传感器（040B01000001）

### 场景页

//todo

## 照明服务详情页面

### 主页



回去设备分组信息，调用接口：

URL：（GET）/m-home/rest/v1/services/lighting

返回数据：

[

{

"locationName": "Ground Floor, Living Room", //区域名称

"locationIcon": "ic\_living\_room.png", //区域图标

"on": 1, //开启的设备数

"off": 2, //关闭的设备数

"devices": [//设备的UUID

"xxxx-device-uuid1-xxxx",

"xxxx-device-uuid2-xxxx",

"xxxx-device-uuid3-xxxx"

]

},

{

"locationName": "Ground Floor, Cookroom",

"locationIcon": "ic\_cookroom.png",

"on": 1,

"off": 0,

"devices": [

"xxxx-device-uuid4-xxxx"

]

}

]

1.区域名称： 返回数据的locationName。

2.区域图标：返回数据的locationIcon.

3.区域开关状态：on/(on+off)

4.设备控制：在照明服务的首页，设备的控制将按照区域为执行单位

* 如果用户未选择区域，用户点击界面下部的全开/全关按钮，则将关闭所有设备

调用接口：（POST）/m-home/rest/v1/devices/me/actions

body: {

"devices":["xxxx-device-uuid-xxxx"], //所有接收器的uuid

"command": {

"cmdName": " set\_action",

"cmdCode": "0001",

"paramters": {

" cmdData": 256

}

}

}

* 首次点击区域图标为选中状态（选中格背景颜色加深）
  + 选中区域后，如果用户点击界面下部的全开/全关按钮，则将关闭该区域内所有设备。例如用户选择了客厅，全关，那个将关闭客厅的3个照明设备。

调用接口：（POST）/m-home/rest/v1/devices/me/actions

body: {

"devices":["xxxx-device-uuid-xxxx"], //该区域所有接收器的uuid

"command": {

"cmdName": " set\_action",

"cmdCode": "0001",

"paramters": {

" cmdData": 256

}

}

}

* + 如果用户再次点击选中区域，则跳转到该服务的设备列表页面，但是只展示该区域的设备（不是所有设备）。例如：用户选择了客厅，然后再次点击选中的客厅则，进入设备列表，但是只展示客厅的3个照明设备。



### 设备页



照明设备包括：

1. 零火单路接受器（040B09050101），

2. 零火双路接受器（040B09050102），

3. 零火三路接受器（040B09050103），

4. 单火单路接受器（040B09050111），

5. 单火双路接受器（040B09050112），

6. 单火三路接受器（040B09050113），

7. 挥手感应零火单路接受器（040B09050201），

8. 挥手感应零火双路接受器（040B09050202），

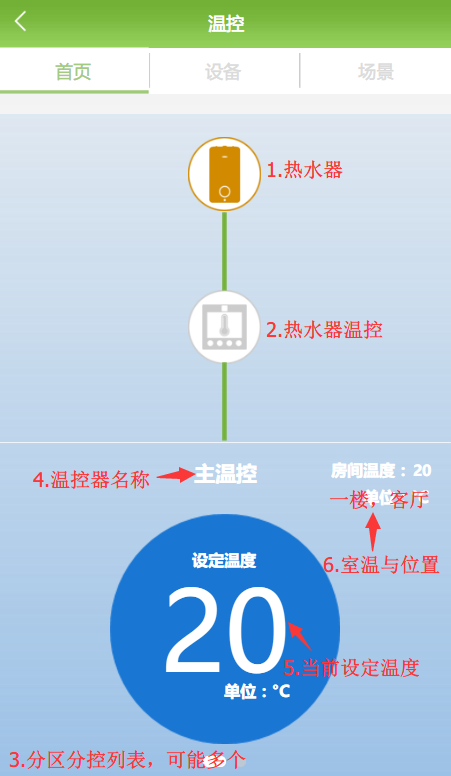
9. 挥手感应零火三路接受器（040B09050203），

### 场景页

//todo

## 温控服务详情页面（#）

### 主页

1.热水箱：水箱有2中状态，一种是加热状态QQ图片20170821220758.png，一种是空闲状态

判断的依据是：【有且只有一个热水箱温控器】

if(MeHotwater< Device.type.id == 040B08080001>extra.items. status === 2){

处于加热状态

}

else if(MeHotwater < Device.type.id == 040B08080001>extra.items. status === 4){

处于空闲状态

}

2. 热水器温控器：如果用户没有添加热水器温控器（Device.type.id == 040B08080001）则改图标为灰色图标，用户点击提示用户添加温控设备，如果存在图标高亮，用户点击进入温控器详情页面。

3. 分区温控列表：如果存在多个温控器则可以左右滑动选择温控器【请查看原型】。

4. 温控器名称：MeThermostat < Device.type.id == 050608070001>.name

5. 室温与位置：

* 房间温度：MeThermostat < Device.type.id == 050608070001>extra.items. dis\_temp
* 位置：MeThermostat < Device.type.id == 050608070001>location.locationName

6：设定温度：MeThermostat < Device.type.id == 050608070001>extra.items. temp\_heat

7：温度设置游标：初始值为当前温度设定值。

* 温度设定：范围[5, 37]

游标选定后延迟（3秒）发送温度设定请求，避免频繁请求

接口调用：/m-home/rest/v1/devices/me/{deviceId}/actions

body：{

"command": {

"cmdName": " set\_temperature ",

"cmdCode": "0002",

"paramters": {

"heat\_mode":3 //当前设备的工作模式，1:离家模式，2：自动模式，3:手动模式

"temp\_heat":"c28.0" //温度值

}

}

}

8：开关控制：【开关控制这块增加模式切换功能，与温控详情页面一致，如下图】



接口调用：/m-home/rest/v1/devices/me/{deviceId}/actions

开关操作：

body：{

"command": {

"cmdName": " set\_onoff",

"cmdCode": "0001",

"paramters": {

"mode": 0 //0:OFF，1：ON

}

}

}

模式切换：

body：{

"command": {

"cmdName": " set\_mode",

"cmdCode": "0002",

"paramters": {

" heat\_mode": 1 //1:离家模式，2：自动模式，3:手动模式

}

}

}

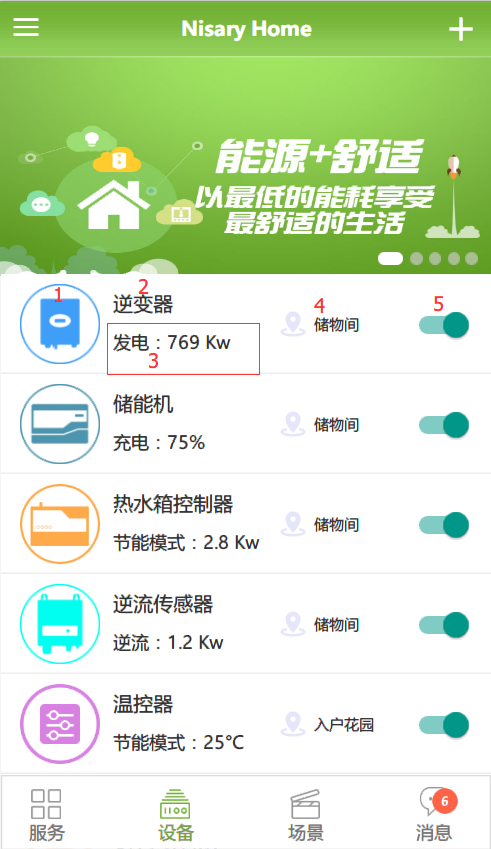
### 设备页



设备列表：MeThermostat（050608070001）

### 场景页

# 设备列表页面



设备列表中的每个设备分为5个要素：

1. 图标：根据设备的icon字段得到设备的图标，如果没有则为设备的默认图标type.icon。

2. 名称：更具设备的name字段得到设备名称。

3. 设备状态：

对于所有设备

（1）如果为关闭状态（status.switch==”OFF”）,显示为【关闭/OFF】。图标为灰色图标

（2）如果为异常状态（status.switch==”ERR”），显示【异常/ERR】。图标为异常图标

（3）如果为开启状态（status.switch==”ON”），显示【开启/ON】。图标为正常图标

以下类型设备在开启状态特殊处理

* 逆变器（040B01000004）：
  + 中文版：【发电：<extra.items. currentPower/1000>kW】
  + 英文版：【electricity: <extra.items. currentPower/1000>kW】
* 储能机（040B01000005）：
  + If(<extra.items. status ==0>)
    - 中文版:：【待机：<extra.items. capacity>%】
    - 英文版:：【Standby：<extra.items. capacity>%】
  + If(<extra.items. status ==1>)
    - 中文版:：【充电：<extra.items. capacity>%】
    - 英文版:：【Charge：<extra.items. capacity>%】
  + If(<extra.items. status ==2>)
    - 中文版:：【放电：<extra.items. capacity>%】
    - 英文版:：【Discharge：<extra.items. capacity>%】
* 热水箱控制器（040B08040004）：
  + If(<extra.items.mode> == 1)
    - 中文版：【节能模式：<extra.items. power/1000>kW】
    - 英文版：【ECO：<extra.items. power/1000>kW】
  + If(<extra.items.mode> == 2)
    - 中文版：【手动模式：<extra.items. power/1000>kW】
    - 英文版：【MAN：<extra.items. power/1000>kW】
  + If(<extra.items.mode> == 3)
    - 中文版：【加热完毕：<extra.items. power/1000>kW】
    - 英文版：【FIN：<extra.items. power/1000>kW】
  + If(<extra.items.mode> == 4)
    - 中文版：【关闭】
    - 英文版：【OFF】
  + If(<extra.items.mode> == 5)
    - 中文版：【调功：<extra.items. power/1000>kW】
    - 英文版：【ADJ：<extra.items. power/1000>kW】
* 逆流传感器（040B01000001）：
  + If(<extra.items.direct> == 16)
    - 中文版：

【取电：<extra.items. effectiveVolt\*extra.items. effectiveVolt/100000>kW】

* + - 英文版：

【Use：<extra.items. effectiveVolt\*extra.items. effectiveVolt/100000>kW】

* + If(<extra.items.direct> == 17)
    - 中文版：

【发电：<extra.items. effectiveVolt\*extra.items. effectiveVolt/100000>kW】

* + - 英文版：

【Generation：<extra.items. effectiveVolt\*extra.items. effectiveVolt/100000>kW】

* 温控器（050608070001）：
  + If(<extra.items. heat\_mode> == 1)
    - 中文版：

【离家：<extra.items. dis\_temp>℃】

* + - 英文版：

【Away：<extra.items. dis\_temp>℃】

* + If(<extra.items. heat\_mode> == 2)
    - 中文版：

【自动：<extra.items. dis\_temp>℃】

* + - 英文版：

【Auto：<extra.items. dis\_temp>℃】

* + If(<extra.items. heat\_mode> == 3)
    - 中文版：

【手动：<extra.items. dis\_temp>℃】

* + - 英文版：

【Manual：<extra.items. dis\_temp>℃】

* 开关面板（040B09051001，040B09051002，040B09051113， 040B09051114）：
  + - 中文版：

【已关联：<X>设备】

* + - 英文版：

【Associated：<X> device】

X的取值为与开关设备关联的设备数量：

X = SUM(distinct(extra. associated. deviceId))

/\*distinct表示去重，不同键可能关联相同设备，简单的说就是取

extra. associated. deviceId数组的并集的长度 \*/

示例：

{

"extra":{

"associated":[

{

"rocker": "B", //B键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//关联了4个设备

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx"

"xxxx-device4-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "A", //A键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//关联了4个设备（但是有2个与B键重复的）

"xxxx-device2-uuid-xxxx", //与B键重复

"xxxx-device4-uuid-xxxx", //与B键重复

"xxxx-device5-uuid-xxxx"

"xxxx-device6-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

}

]

}

}

计算：

extra. associated. deviceId并集 =

[

“xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx",

"xxxx-device4-uuid-xxxx",

"xxxx-device5-uuid-xxxx",

"xxxx-device6-uuid-xxxx"

]

X = 并集.length() = 6

4. 位置：根据设备的location.locationName的到设备位置信息。

5. 设备开关状态：ON:开，OFF：关， ERR：异常，除了ON以外，其他（OFF，ERR）都是OFF。（MeRFCT和开关面板类，太阳能设备不支持开关操作）

设备开关操作，调用接口：

URL:(POST) /m-home/rest/v1/devices/me/{deviceId}/actions

BODY：（支持开关操作的设备如下：）

1. MeBoost（040B08040004）: 热水控制器

1. 关：

{

"command": {

"cmdName": "set\_mode ",

"cmdCode": "0002",

"paramters": {

"mode": "OFF"

}

}

}

1. 开：

{

"command": {

"cmdName": "set\_mode ",

"cmdCode": "0002",

"paramters": {

"mode": "ECO" //默认打开MeBoost都进入ECO模式

}

}

}

2. MeThermostate(050608070001): 温控器

1. 关：

{

"command": {

"cmdName": " set\_onoff",

"cmdCode": "0001",

"paramters": {

" status\_onoff": 0

}

}

}

1. 开：

{

"command": {

"cmdName": " set\_onoff",

"cmdCode": "0001",

"paramters": {

" status\_onoff": 1

}

}

}

3. MeReceiver-LN-1: 单路接收器（040B09050101）

1. 关：

{

"command": {

"cmdName": "set\_action",

"cmdCode": "0001",

"paramters": {

"cmdData": 256

}

}

}

1. 开：

{

"command": {

"cmdName": "set\_action",

"cmdCode": "0001",

"paramters": {

"cmdData": 257

}

}

}

4. MeWallPanel-LN-3: 三路零火带挥手感应接收器（040B09050203）

1. 关：

{

"command": {

"cmdName": "set\_action",

"cmdCode": "0001",

"paramters": {

"cmdData": 1792

}

}

}

1. 开：

{

"command": {

"cmdName": "set\_action",

"cmdCode": "0001",

"paramters": {

"cmdData": 1799

}

}

}

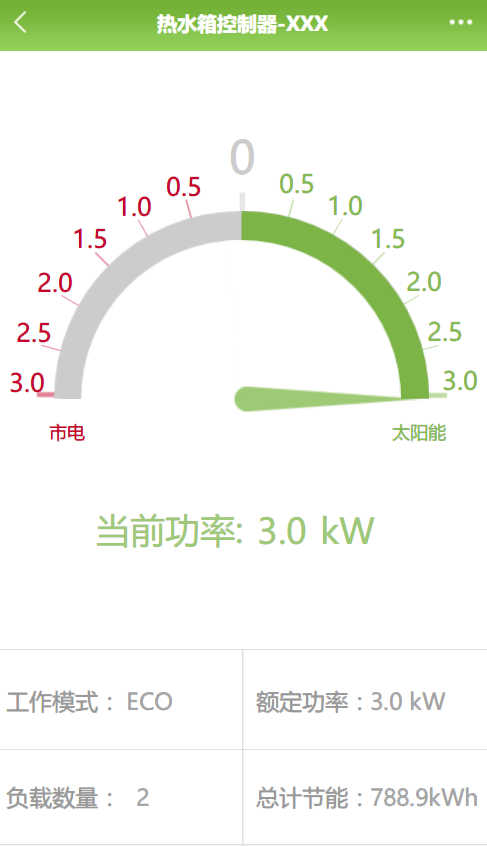
# 设备详情页面

进入设备详情页面

首先调用接口：（GET）/m-home/rest/v1/devices/me/{deviceId} 获取设备的详细信息，主要是为了获取设备的静态参数和一些状态数据。

然后调用轮询接口：（GET）/m-home/rest/v1/devices/me/{deviceId}/items获取设备的运行时参数，都是一些实时的可变参数。

## MeBoost详情页面



【调用接口：查询设备所有实时指标/m-home/rest/v1/devices/me/{deviceId}/items】

1. 当前功率：如果模式为手动模式：MAN，则表示用的“市电”，如果节能模式：ECO则表示用的“太阳能”。

当前功率=<MeBoost>.extra.items.power

1. 工作模式：<MeBoost>.extra.items.mode
   1. 0x0001: ECO，节能模式
   2. 0x0002: MAN，节能模式
   3. 0x0003: FIN，加热完毕
   4. 0x0004:OFF，关闭状态
   5. 0x0005:ADJ，节能模式
   6. 0xFFFX: [0xFFF1, 0xFFF2, 0xFFF3, 0xFFF4]异常状态
2. 额定功率：<MeBoost>.extra. nominalPower
3. 负载数量：<MeBoost>.extra. loadType
   1. 0x00:未连接负载
   2. 0x10:单路可控硅连接
   3. 0x11:单路继电器负载连接
   4. 0x20: 双路负载连接
   5. 0xFF:无效连接

1. 总计节能：<MeBoost>.extra.items. energySaved

【调用接口：获取指定设备的报表数据/m-home/rest/v1/datas/reports/me/{deviceId}】

dailyReport：返回24个点值

monthlyReport：返回的点数已实际月的天数为准

yearlyReport：返回12个点值

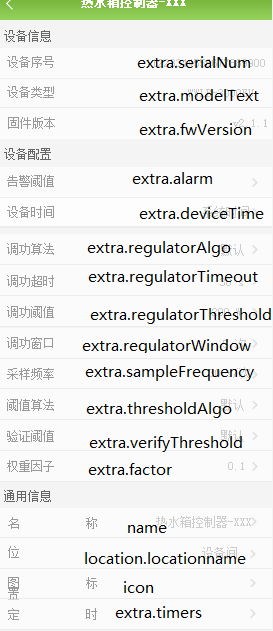
allReport：返回用户自使用该系统到现在的以年为单位的点值，例如用户2012年开始使用该系统则返回6个点值[2012,2013,2014,2015,2016,2017]

1. 日功率走势图：[dailyReport.power]
2. 能耗统计报表：日，月，年，全部（2个指标在同一坐标系）

节能数据：[<dailyReport,monthReport,yearReport,allReport >. energySaved]

用电数据：[<dailyReport,monthReport,yearReport,allReport >. energyUsed]

设备信息详情：



【图标取消,MeBoost不能更改图标】

## MeRFCT详情页面



**设备实时指标**：

1. 太能能发电：用电（MeRFCT.extra.items.direct ==16）,发电（MeRFCT.extra. items.direct ==17）

功率计算：(MeRFCT.extra. items. effectiveVolt\* MeRFCT.extra. items. effectiveCurrent)/100（单位：Kw）

2. 总计用电：MeRFCT.extra. items. toUser （单位：kWh）

3. 总计发电：MeRFCT.extra. items. toGrid（（kWh）：kWh）

**报表数据：**

【调用接口：获取指定设备的报表数据/m-home/rest/v1/datas/reports/me/{deviceId}】

dailyReport：返回24个点值

monthReport：返回的点数已实际月的天数为准

yearReport：返回12个点值

allReport：返回用户自使用该系统到现在的以年为单位的点值，例如用户2012年开始使用该系统则返回6个点值[2012,2013,2014,2015,2016,2017]

1. 日功率走势图：

取电功率：[dailyReport.toUserPower]

输电功率：[dailyReport.toGridPower]

2. 能源统计报表：日，月，年，全部（2个指标在同一坐标系）

发电数据：[<dailyReport,monthReport,yearReport,allReport >. toGrid]

去电数据：[<dailyReport,monthReport,yearReport,allReport >. toUser]

**设备信息详情**



【图标与定时器取消不做，不能更改图标也没有定时操作】

## MePanel详情页面

### MePanel1(单开无线无缘开关面板)详情页面



1. 已关联设备：已经实现与开关面板管理的接收器

数据来源：<MePanel1>.extra. associated[

{

"rocker": "B", //对于单键的开关面板，按键标识固定都是B

"action": "press", //固定都是press，标识按下

"deviceId":[“xxxx-device-uuid-xxxx”], //所有被关联设备的UUID

"command": { //控制命令【目前占时忽略】

"name": "",

"option": "",

"parameters": []

}

}

]

由于单键开关只有一个键，所以associated数组只会有一个元素。其中的deviceId就是被关联的设备。因此根据deviceId这个数组就能从现有的设备列表中把这些接收器（MeReceiver）找出来。然后根据设备的location将对应的图标显示出来【这个逻辑跟“照明服务”（2.2.1）里面的设备图标显示一样】。

1. 修改关联设备：修改开关面板管理的设备



选择开关面板要关联的接收器，目前包括：

1. 单路零火：MeReceiver-LN-1(040B09050101)，

2. 三路零火带挥手感应：MeWallPanel3-LN-3(040B09050203)），

这些设备都从现有的缓存设备列表中过滤得到，如果已经关联的标记为“已选择”。点击“完成”，提交修改：

调用接口：

[URL:(PUT)/m-home/rest/v1/devices/me/{deviceId}](URL:(PUT)/m-home/rest/v1/devices/me/%7bdeviceId%7d)

BODY: {

"extra.associated":[

{

"rocker": "B", //对于单键的开关面板，按键标识固定都是B

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx"

"xxxx-device4-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略】

"name": "",

"option": "",

"parameters": []

}

}

]

}

1. 清除设备：清除开关面板关联的设备

调用接口：

[URL:(PUT)/m-home/rest/v1/devices/me/{deviceId}](URL:(PUT)/m-home/rest/v1/devices/me/%7bdeviceId%7d)

BODY: {

"extra.associated":[ ] //设置为空数组

}

### MePanel2(双开无线无缘开关面板)详情页面



双开面板相当于是2个单开面板，区别就是单开只有一个B键，而双开有AB两个键，点击B键显示与B键关联的设备，修改关联设备与清除关联设备都是针对B键，同理对A键的处理类似，逻辑完全和3.3.1中对单开的处理一样。这里在数据上举例说明一下：

{

"extra":{

"associated":[

{

"rocker": "B", //B键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx"

"xxxx-device4-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "A", //A键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device2-uuid-xxxx",

"xxxx-device4-uuid-xxxx",

"xxxx-device6-uuid-xxxx"

"xxxx-device7-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

}

]

}

}

### MePanel3(三开无线无缘开关面板)详情页面



三间开关面板与两开开关面板类似，只是有3个键C,B,A

在数据上举例说明一下：

{

"extra":{

"associated":[

{

"rocker": "C", //C键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "B", //B键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx"

"xxxx-device4-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "A", //A键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device2-uuid-xxxx",

"xxxx-device4-uuid-xxxx",

"xxxx-device6-uuid-xxxx"

"xxxx-device7-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

}

]

}

}

### MePanel4(四开无线无缘开关面板)详情页面



四键开关面板的页面逻辑和双键，三键类似，只不过有4个键（按坐标命名）：BI，AI，BO，AO。

在数据上举例说明一下：

{

"extra":{

"associated":[

{

"rocker": "BI", //BI键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "AI", //AI键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "BO", //BO键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device1-uuid-xxxx",

"xxxx-device2-uuid-xxxx",

"xxxx-device3-uuid-xxxx"

"xxxx-device4-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

},

{

"rocker": "AO", //AO键的关联关系

"action": "press", //固定都是press，标识按下

"deviceId":[//所有被关联设备的UUID

"xxxx-device2-uuid-xxxx",

"xxxx-device4-uuid-xxxx",

"xxxx-device6-uuid-xxxx"

"xxxx-device7-uuid-xxxx"

],

"command": { //控制命令【目前占时忽略,默认以下空数据】

"name": "",

"option": "",

"parameters": []

}

}

]

}

}

## 温控器详情页面



**设备状态：**

1.锁定状态：<暂时不支持，隐藏>

2.加热状态:<Device>.extra.items. status，4：表示加热状态，2：表示非加热状态，如果是处于加热状态，加热图标闪烁。

3.当前室温：<Device>.extra.items. dis\_temp，格式（c28.5）表示28.5摄氏度

4.温度设置值：<Device>.extra.items. temp\_heat，格式（c28.5）表示28.5摄氏度

5.温度设置游标：范围[5,37]

游标选定后延迟（3秒）发送温度设定请求，避免频繁请求：

接口调用：/m-home/rest/v1/devices/me/{deviceId}/actions

body：{

"command": {

"cmdName": " set\_temperature ",

"cmdCode": "0003",

"paramters": {

"heat\_mode":3 //当前设备的工作模式，1:离家模式，2：自动模式，3:手动模式

"temp\_heat":"c28.0" //温度值

}

}

}

6.工作模式：<Device>.extra.items. heat\_mode， 1：离家模式，2：自动模式，3：恒温模式

接口调用：/m-home/rest/v1/devices/me/{deviceId}/actions

body：{

"command": {

"cmdName": " set\_mode",

"cmdCode": "0002",

"paramters": {

" heat\_mode": 1 //1:离家模式，2：自动模式，3:手动模式

}

}

}

7.当前设置模式：显示当前设备模式图标

8.开关：开关设备

接口调用：/m-home/rest/v1/devices/me/{deviceId}/actions

body：{

"command": {

"cmdName": " set\_onoff",

"cmdCode": "0001",

"paramters": {

"status\_onoff": 0 //0:OFF，1：ON

}

}

}

**数据报表：（只有一个温度的日月年报表）**

【调用接口：获取指定设备的报表数据/m-home/rest/v1/datas/reports/me/{deviceId}】

dailyReport：返回24个点值

monthReport：返回的点数已实际月的天数为准

yearReport：返回12个点值

allReport：返回用户自使用该系统到现在的以年为单位的点值，例如用户2012年开始使用该系统则返回6个点值[2012,2013,2014,2015,2016,2017]

温度报表数据：[<dailyReport,monthReport,yearReport,allReport>.temperature]

**设备信息详情**



## 接收器详情页面

### 单路接收器详情页面



设备控制，调用接口：

URL：（POST）/devices/me/{deviceId}/actions

BODY:

{

"cmdCode": " 0001 ",

"cmdName": " set\_action ",

"parameters": {

“cmdData”:257 //OFF:256, ON:257

}

}

设备配置信息：



## 逆变器详情页面



**实时指标：**

1. 当前功率：extra.items. currentPower
2. 日电量：extra.items. dayTotalEnergy
3. 总电量：extra.items. totalEnergy
4. 最大功率：extra.items. powerMax
5. 收益(￥)：extra.items. totalEnergy\*0.29

**报表数据：**

【调用接口：获取指定设备的报表数据/m-home/rest/v1/datas/reports/me/{deviceId}】

dailyReport：返回24个点值

monthReport：返回的点数已实际月的天数为准

yearReport：返回12个点值

allReport：返回用户自使用该系统到现在的以年为单位的点值，例如用户2012年开始使用该系统则返回6个点值[2012,2013,2014,2015,2016,2017]

1. 日功率走势图：[dailyReport.currentPower]

2. 能源统计报表：日，月，年，全部（2个指标在同一坐标系）

发电数据：[<dailyReport,monthReport,yearReport,allReport >. totalEnergy]

**设备信息详情**



## 储能机详情页面



**实时指标：**

1. 耗电功率：
2. 光伏功率：
3. 电网功率：
4. 最大功率：

**报表数据：**

【调用接口：获取指定设备的报表数据/m-home/rest/v1/datas/reports/me/{deviceId}】

dailyReport：返回24个点值

monthReport：返回的点数已实际月的天数为准

yearReport：返回12个点值

allReport：返回用户自使用该系统到现在的以年为单位的点值，例如用户2012年开始使用该系统则返回6个点值[2012,2013,2014,2015,2016,2017]

1. 日功率走势图：[dailyReport.currentPower]

2. 能源统计报表：日，月，年，全部（2个指标在同一坐标系）

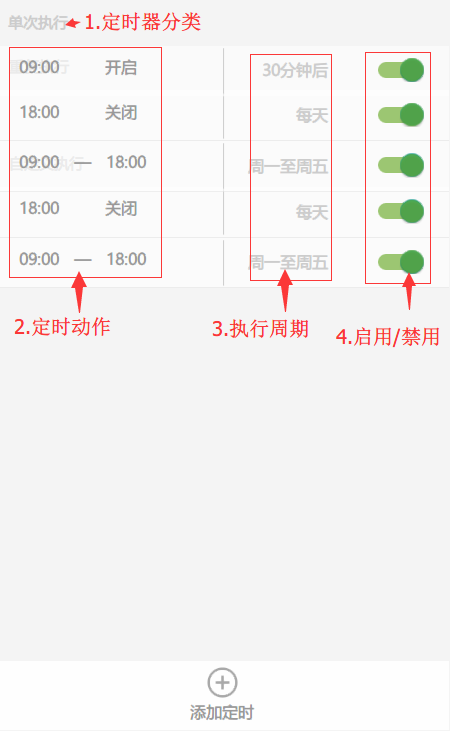
发电数据：[<dailyReport,monthReport,yearReport,allReport >. totalEnergy]

**设备信息详情**



## \*定时开关详情页面（温控，MeBoost）

### 定时开关列表页面



获取定时器列表接口：

URL：/m-home/rest/v1/flows/me/device/{deviceId}

RESPONSE：

{

"weiwiz": {

"MHome": {

"response": {

"automatics": [

{

"timerId": "xxxx-timer-uuid-xxxx",

"actionTime":"09:00 开启",

"actionCycle":"周一到周五",

"enable": true

}

]

},

"errorId": "",

"errorMsg": ""

}

}

}

* + - 1. **定时器分类**：【取消分类，直接列表显示】
      2. **定时动作**：取值<actionTime>。
      3. **执行周期:** 取值< executeCycle >
      4. **启用禁用标志：**取值<enable>

**启用/禁用定时器接口：**

URL:(POST) /m-home/rest/v1/flows/me/device/{deviceId}/{flowId}/actives

BODY:

{

"automatic": {

"enable": true, // true:启用定时器，false：禁用定时器

"commands": [ //\*对命令的启用/禁用，对于定时器可以忽略这个字段

{

"cmdName": "open",

"cmdCode": "0001",

"enable": true

}

]

}

}

### 修改定时器详情页面

#### 定时器主页面



获取定时器详情接口：

URL：/m-home/rest/v1/flows/me/device/{deviceId}/{flowId}

RESPONSE：

{

"weiwiz": {

"MHome": {

"response": {

"automatic": {

"timerId": "xxxx-timer-uuid-xxxx",

"mode": "PARALLEL",

"name": "timer",

"interval": 0,

"between": [

"09:00",

"18:00"

],

"weekday": [

1,

2,

3,

4,

5,

6,

0

],

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"cmd": {

"cmdCode": "0001",

"cmdName": "open",

"parameters": {}

},

"enable": true

},

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"cmd": {

"cmdCode": "0002",

"cmdName": "close",

"parameters": {}

},

"enable": true

}

],

"enable": true

}

},

"errorId": "",

"errorMsg": ""

}

}

}

* + - 1. 执行计划描述：

If(weekday.size ==0) then 执行一次

else if(weekday==[1,2,3,4,5,6,0]) then 每天

else if(weekday==[1,2,3,4,5]) then 工作日

else if(weekday==[6,0]) then 周末

else 自定义

【注意如果是“执行一次”，隐藏前面的“重复”】

* + - 1. 开启时间：取值between[0]，如果between[0] == “”，表示“未设置”

* + - 1. 关闭时间：取值between[1]，如果between[1] == “”，表示“未设置”
      2. 取消修改：如果存在变动，提示用户“是否放弃修改”
      3. 提交修改：调用接口

URL:(PUT) /m-home/rest/v1/flows/me/device/{deviceId}/{timerId}

BODY:

{

"timerId":"xxxx-timer-uuid-xxxx"

"mode": "PARALLEL",

"name": "每天",

"enable":true,

"interval": 0,

"between": ["08:00", "12:00"],

"weekday": [0, 1, 2, 3, 4, 5, 6],

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"enable":true,

"cmd": {

"cmdCode": "0001",

"cmdName": "open",

"parameters": {}

}

}

]

}

#### 修改执行计划页面



#### 修改执行时间页面



* + - 1. 时间选择：选择执行时间点
      2. 清除：清除后时间点设置为空字符串 ””

### 添加定时器详情页面



* + - 1. 执行计划：默认为每天，weekday=[1,2,3,4,5,6,0]
      2. 执行时间：默认为未设置，时间点值为“”
      3. 取消：如果存在修改，提示用户是否放弃添加
      4. 确定：提交定时器

调用接口：

URL：(POST) /m-home/rest/v1/flows/me/device/{deviceId}

BODY:

{

"mode": "PARALLEL",

"name": "timer",

"enable": true,

"interval": 0,

"between": ["08:00", "12:00"],

"weekday": [0, 1, 2, 3, 4, 5, 6],

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"enable": true,

"cmd": {

"cmdCode": "0001",

"cmdName": "open",

"parameters": {}

}

}

]

}

**【添加定时器模板】**

{

"mode": "PARALLEL", //默认值

"name": "timer", //默认值

"enable": true, //默认值

"interval": 0, //默认值

"between": ["", ""], //默认值，开启/关闭都是未设置

"weekday": [0, 1, 2, 3, 4, 5, 6], //默认值，每天

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx", //设备UUID

"deviceType": "device-type-id", //设备类型ID

"enable": true, //默认值

"cmd": {

<具体设备开启操作命令>

}

}，

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "device-type-id",

"enable": true,

"cmd": {

<具体设备关闭操作命令>

}

}

]

}

不同类型设备的开启关闭命令列表：

* + - 1. MeThermostat(050608070001) : 温控器
         1. 开启：

{

"cmdName":"set\_onoff",

"cmdCode":"0001",

"parameters":{

"status\_onoff":1

}

}

* + - * 1. 关闭

{

"cmdName":"set\_onoff",

"cmdCode":"0001",

"parameters":{

"status\_onoff":0

}

}

* + - 1. MeReceiver(040B09050101，040B09050111，040B09050201) :单路接收器
         1. 开启：

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":257

}

}

* + - * 1. 关闭

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":256

}

}

* + - 1. MeReceiver(040B09050102，040B09050112，040B09050202) :双路接收器
         1. 开启：

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":771

}

}

* + - * 1. 关闭

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":768

}

}

* + - 1. MeReceiver(040B09050103，040B09050113，040B09050203) :三路接收器
         1. 开启：

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":1799

}

}

* + - * 1. 关闭

{

"cmdName":" set\_action ",

"cmdCode":"0001",

"parameters":{

"cmdData":1792

}

}

# 消息页面

## 系统消息

### 消息列表



* 进入系统消息页面调用消息查询接口：

**URI**：（GET）/m-home/rest/v1/events/me

Query Parameter：offset= latest

【返回用户下所有设备的最后一条消息,每个设备只有一条】

消息返回：

{

"weiwiz": {

"MHome": {

"response": {

"events": [

{

"index": "66", //序列号

"timestamp": "2015-10-12T06:32:50.441Z",

"userUuid": "d022d760-61b4-11e5-a1f7-07f1ad4ed417",

"ownerUuid": "d1f7be90-629d-11e5-a1f7-07f1ad4ed417",

"deviceUuid": "7ddb2230-631e-11e5-a1f7-07f1ad4ed417",

"deviceName": "TV",//设备名称

"deviceType": "040300101001",//设备类型决定消息列表图标

"eventTag": "SWITCH\_CONTROLLER\_STATE\_CHANGED",

"eventLevel": 0,

"eventDescription": "switch on",//消息内容

"handled": false

}

]

},

"errorId": "",

"errorMsg": ""

}

}

}

* 设备消息列表由3个要素组成：

1. 设备图标：设备图标取决于设备的类型（deviceType），每个设备类型对应一个默认图标。

消息列表里面有2类特殊的消息：场景消息和系统消息

**场景消息**：由场景被执行为产生的消息，场景可能包含5个设备类型：

0016080B0000：用户自定义场景，对应图标【待定】

0016080B0001：系统场景-回家，对应图标E:\项目文档\MeApp\03-切图\添加场景\系统场景-回家.png

0016080B0002：系统场景-离家，对应图标E:\项目文档\MeApp\03-切图\添加场景\系统场景-离家.png

0016080B0003：系统场景-起床，对应图标E:\项目文档\MeApp\03-切图\添加场景\系统场景-起床.png

0016080B0004：系统场景-睡觉，对应图标E:\项目文档\MeApp\03-切图\添加场景\系统场景-睡觉.png

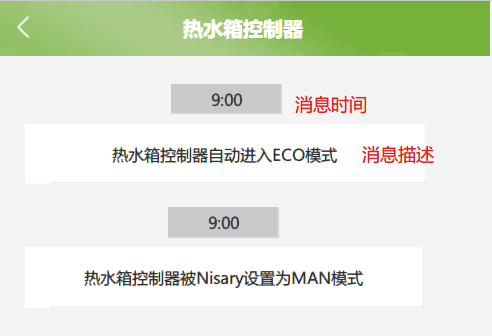
【图标可参见场景管理页面】

**系统消息**：由系统发送给用户消息，因此该消息的deviceType为用户的类型ID：060A08000000，因此只要deviceType==060A08000000就是系统消息，对应图标: 

2. 设备名称：名称由该设备最后一条消息的名称（deviceName）决定，因为一个设备可能被重命名过，以最近的名称为准。

3. 最近消息描述：最近消息描述就是该设备最后一条消息的描述（eventDescription）

### 设备消息列表



**URI**：（GET）/m-home/rest/v1/events/me/{deviceId}

Query Parameter：

offset=<last\_index>, 偏移参考序号

limit=20, 分页窗口大小

forward = true，true向前翻页（index<offset），false：向后翻页(index>offset)

例如：offset = 100&&limit=10&&forward=true, 返回结果是95 <= index < 100

设备消息列表是改设备的所有消息的倒序排列，每条消息由2个元素组成：

* + - 1. 消息时间：消息产生的时间（timestamp）
      2. 消息内容：消息的具体描述（eventDescription）

## 用户消息（暂时不做）

# 场景页面

## 场景列表页

****

调用场景查询接口：

URL：(GET) /m-home/rest/v1/flows/me/

接口返回：

{

"weiwiz": {

"MHome": {

"response": {

"automatics": [

{

"sceneId": "xxxx-scene-uuid-xxxx",

"mode": "PARALLEL",

"name": "回家",

"type": "0016080B0001",

"description": "一切都为你准备好了",

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"cmd": {

"cmdCode": "0001",

"cmdName": "open",

"parameters": {}

}

}

]

}

]

},

"errorId": "",

"errorMsg": ""

}

}

}

1. 回家，离家，起床，睡觉这4个系统预置场景是根据返回列表里面的type来判断是否已经配置过了，如果没有配置就类似原型中的“睡觉”场景，图标灰色，用户点击提示用户去配置
   1. 0016080B0000:用户自定义场景【多个】
   2. 0016080B0001:系统场景-回家【唯一】
   3. 0016080B0002:系统场景-离家【唯一】
   4. 0016080B0003:系统场景-起床【唯一】
   5. 0016080B0004:系统场景-睡觉【唯一】

## 执行场景



场景执行接口：

URL：(POST) /m-home/rest/v1/flows/me/{flowId}/actions

【flowId即场景的id】

## 编辑场景



### 编辑场景控制命令列表：



### 编辑命令

不同设备支持不同的命令

* 1. MeBoost：040B08040004
     1. OFF 关闭设备
     2. ECO 设置为节能模式
     3. MAN 设置为手动模式

命令封装为：

{

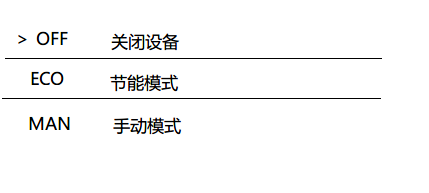
"cmdCode": " 0002 ",

"cmdName": " set\_mode ",

"parameters": {“mode”:”ECO”}//根据用户的选择

}

示意草图：



* 1. MeReceiver-LN-1: 040B09050101
     1. OFF 关闭
     2. ON 打开

命令封装为：

{

"cmdCode": " 0001 ",

"cmdName": " set\_action ",

"parameters": {

“cmdData”:257 //OFF:256, ON:257

}

}

MeWallPanel-LN-3: 040B09050203

MeWallPanel-SL-3: 040B09050113

* + 1. OFF 全部关闭
    2. ON 全部打开

命令封装为：

{

"cmdCode": " 0001 ",

"cmdName": " set\_action ",

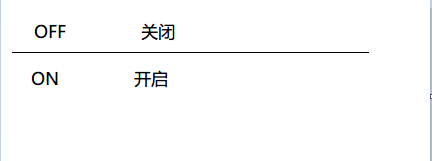
"parameters": {

“cmdData”: 1792 // OFF:1792, ON:1799

}

}

示意草图：



* 1. MeThermostat(WiFi): 050608070001
     1. OFF 关闭
     2. ON 开启

命令封装为：

{

"cmdCode": " 0001 ",

"cmdName": " set\_onoff ",

"parameters": {

“status\_onoff”: 0 // OFF:0, ON:1

}

}

* + 1. AWAY 外出
    2. AUTO 自动模式
    3. MANUAL 恒温模式

命令封装为：

{

"cmdCode": " 0002 ",

"cmdName": " set\_mode ",

"parameters": {

“heat\_mode”: 0 // AWAY:1, AUTO:2， MANUAL：3

}

}

* + 1. SET 设置温度，select选择框[5, 37]

命令封装为：

{

"cmdCode": " 0003 ",

"cmdName": " set\_temperature ",

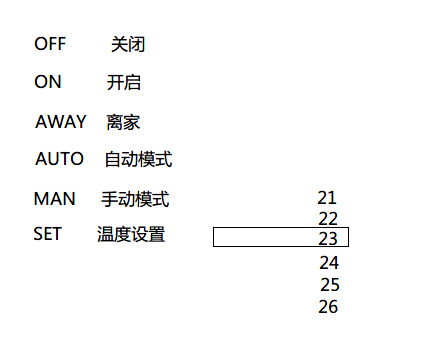
"parameters": {

“temp\_heat”: "c28.0" // 根据用户选择为准

}

}

示意草图：



### 添加设备控制命令

从现有设备列表中选择要进行操作的设备，并指定执行的控制命令，目前只支持这几类设备：

* 1. MeBoost：040B08040004
  2. MeReceiver-LN-1: 040B09050101
  3. MeWallPanel-LN-3: 040B09050203
  4. MeWallPanel-SL-3: 040B09050113
  5. MeThermostat(WiFi): 050608070001



设备支持的命令类似上面5.3.2中描述

### 添加延时器（暂时不做）

### 编辑确认提交

场景编辑完成后提交数据：

URL：（PUT）/m-home/rest/v1/flows/me/{flowId}

BODY：

{

"sceneId": "xxxx-scene-uuid-xxxx",

"mode": "PARALLEL",

"name": "回家",

"type": "0016080B0000",

"description": "一切都为你准备好了",

"commands": [

{

"uuid": " afa02f31-1eba-45f8-988c-41c14f14df95",

"deviceType": " 040B08040004",

"cmd": {

"cmdCode": "0001",

"cmdName": " set\_mode ",

"parameters": {“mode”:”ECO”}

}

}

]

}

## 添加场景

![C:\Users\jacky\AppData\Roaming\Tencent\Users\418626849\QQ\WinTemp\RichOle\[K_B2S3](V)W9NR70FFCADX.png](data:image/png;base64,) 



### 自定义场景

【添加自定义场景和5.3章节的编辑场景页面逻辑基本一致，不在累述】

场景编辑完成后提交数据：

URL：（POST）/m-home/rest/v1/flows/me/

BODY：

{

"mode": "PARALLEL",//场景模式目前固定为PARALLEL，并行模式

"name": "回家",

/\*

\*0016080B0000:用户自定义场景

\*0016080B0001:系统场景-回家

\*0016080B0002:系统场景-离家

\*0016080B0003:系统场景-起床

\*0016080B0004:系统场景-睡觉

\*/

"type": "0016080B0000",

"description": "一切已经为你准备好了",

"commands": [

{

"uuid": "xxxx-device-uuid-xxxx",

"deviceType": "",

"cmd": {

"cmdCode": "0001",

"cmdName": " set\_mode ",

"parameters": {“mode”:”ECO”}

}

}

]

}

### 定时场景（暂不实现）

### 自动化场景（暂不实现）

## 删除场景

1. 删除场景：

URL：（DELETE）/m-home/rest/v1/flows/me/{flowId}

1. 是否推送结果通知：

URL：（POST）/m-home/rest/v1/flows/me/{flowId}/settings?notify=true

【true：推送，false：不推送】

# 侧边栏

## 用户页面



获取用户信息：

URL：（GET）/m-home/rest/v1/users/me

修改用户信息：

URL：（PUT）/m-home/rest/v1/users/me

BODY：{

"name": "",

"icon": "",

"extra": {

"email": "",

"language": "en"

}

}

### 修改密码



修改密码接口：

URL：（POST）/m-home/rest/v1/system/settings/user/forgot/{userId}

BODY：

{

"password": "new-password",

"old\_password":"old-password",

"captcha": "1234"

}

old\_password:通过旧密码更改密码，

captcha:通过验证码更改密码，二者二选一

## 设置页面



获取用户设置：

URL：(GET) /m-home/rest/v1/users/me/settings

设置用户设置：

URL：(POST) /m-home/rest/v1/users/me/settings

BODY:

{

"name": "notify\_scene",//设置项名称

"value": "true" //设置项值

}

## 帮助与反馈

### 主页面



获取帮助信息接口：

URL：(GET)/m-home/rest/v1/system/settings/wrappers/:markid/helps

RESPONSE:

{

"device":[

{

"type":"040B08040004",

"name":"热水箱控制器",

"FAQ":[

{

"question":"设备如何安装？",

"answer":"这样。。。这样。。。。"

}

]

}

]

"other":[

{

"type":"061608000000",

"name":"场景",

"FAQ":[

{

"question":"什么是场景？",

"answer":"场景就是。。。"

}

]

}

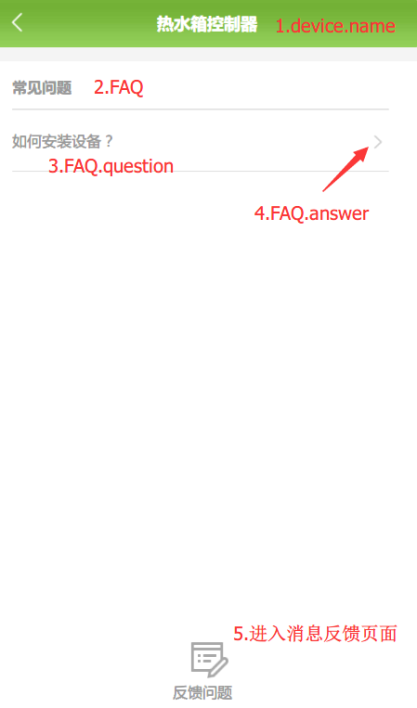
]

}

关于Other中设备类型与icon的对应关系：

1. 场景：
   1. Type：061608000000
   2. Icon： 
2. 应用体验：
   1. Type：061808000000
   2. Icon： C:\Users\jacky\AppData\Roaming\Tencent\Users\418626849\QQ\WinTemp\RichOle\8C{E(F8CDJUQ[){N~GB@VL1.png
3. 服务：
   1. Type：061908000000
   2. Icon：C:\Users\jacky\AppData\Roaming\Tencent\Users\418626849\QQ\WinTemp\RichOle\[SD47%U54U$IGT4X$)$D`RV.png

### 设备帮助页面



* + - 1. 标题：以设备名称作为title
      2. 常见问题：中文（常见问题）/ 英文（FAQ）
      3. 问题描述：FAQ.question
      4. 问题回答：点击展开问题回答，FAQ.answer

### 消息反馈页面



* + - 1. 标题：以设备名称作为title
      2. 联系方式：提示用户输入email地址
      3. 反馈消息内容：用户输入反馈消息内容
      4. 提交：调用接口

URL：（POST）/m-home/rest/v1/messages/me

BODY：{

"type":"<device\_type\_id>", //设备类型ID

"name":"<device\_name>", //设备类型名称

"description": "反馈信息", //反馈信息内容

"email": "邮箱地址" //用户联系方式

}

### 反馈消息列表页面



查询消息反馈列表(按index降序排列)：

URL：(GET) /m-home/rest/v1/messages/me

RESPONSE:

[{

"index":"1", //序列号

"timestamp": "2015-10-12T06:32:50.441Z",//消息反馈时间戳

"type":"<device\_type\_id>", //设备类型ID

"name":"<device\_name>", //设备类型名称

"description": "反馈信息", //反馈消息内容

"feedback":"回复" //反馈消息回复内容

}]

* + - 1. 反馈消息内容：取值<description>,
      2. 设备名称：取值<name>
      3. 消息反馈时间：取值<timestamp>
      4. 反馈消息回复内容：取值<feedback>