

移动机器人软件 API 手册

型号：X2BOT V3.0

（圆柱标准版）



北京智能佳科技有限公司

2025 年 7 月

目 录

总体介绍	4
适用环境:	5
协议形式和连接方式:	5
接口使用说明:	5
1. 机器人移动功能	6
指令:	6
参数:	6
接口描述:	6
示例:	7
2. 移动取消功能	7
指令:	7
参数:	7
接口描述:	8
示例:	8
3. 获取机器人当前全局状态	8
指令:	8
参数:	8
接口说明:	8
示例:	9
4. 插入新的 marker 点位	10
指令:	10
参数:	10
接口说明:	10
示例:	10
5. 获取 marker 点位列表	11
指令:	11
参数:	11
接口说明:	11
示例:	11
6. 机器人直接控制指令	13
指令:	13
参数:	13
接口说明:	13
示例:	14
7. 机器人急停控制指令	14
指令:	14
参数:	14
接口说明:	14
示例:	15
8. 机器人初始位置校定	15
指令:	15
参数:	15
接口说明:	15
示例:	16
9. 请求机器人实时数据	16

指令:	16
参数:	16
接口说明:	16
示例:	17
10. 机器人主动通知	17
接口说明:	17
通知列表:	17
字段解释:	19
示例:	20
11. 设置参数	20
指令:	20
参数:	20
接口说明:	20
示例:	21
12. 获取参数	21
指令:	21
参数:	21
接口说明:	21
示例:	21
13. 无线网络接口	22
13.1 获取机器人当前可用的 WiFi 列表	22
13.2 连接 WiFi	23
13.3 获取当前连接的 WiFi	24
13.4 获取机器人 IP 和无线网卡地址	25
14. 地图接口	26
14.1 获取地图列表	26
14.2 设置当前地图	27
14.3 获取当前地图	27
15. 关机重启接口	28
16. 软件更新接口	29
16.1 获取当前软件版本	29
16.2 检查更新	30
16.3 更新	31
a. 状态代码返回说明（针对返回数据的 status）	33
b. 错误消息返回说明（针对返回数据的 error_message）	33

总体介绍

X2BOT V3.0 是一套具有成熟稳定的工业级移动平台，具有感知环境和定位导航能力的产品级轮式机器人开发平台，采用六轮运动方式，驱动轮带有悬挂系统，满足在不同路面下运行。该平台配有惯性导航和激光模块，一站式 SLAM 工具包，快速完成工作环境部署，可以在未知环境下的自主导航与避障，并具有自动回充电功能。软件提供 API 接口和建图网页，可以方便调取机器人运动信息，是开发各种商业服务机器人完整的底层运动平台，也是科研院所研究轮式机器人以及多机器人协作理想平台。有些操作需要用到 API，需结合智能佳提供的 API 文档一起使用。

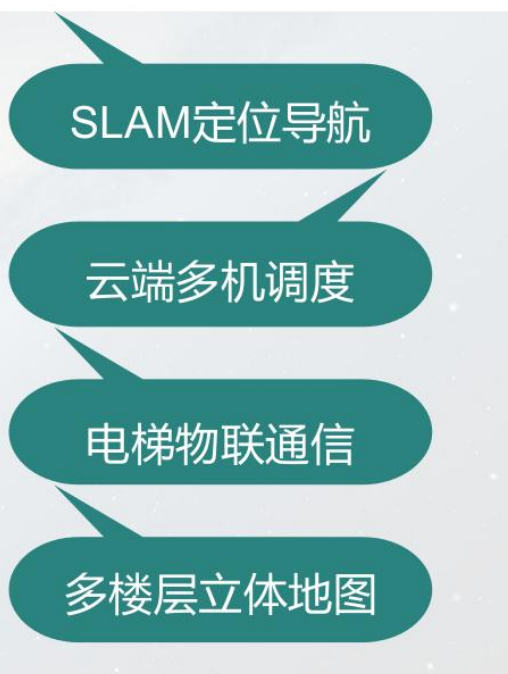
适合用途

- 商用轮式服务机器人开发；
- 教学与科研平台实验；
- 多机器人协作开发；

• 产品定位 •



• 核心功能 •



适用环境:

- 基于 TCP socket 的网络通信。

协议形式和连接方式:

协议形式: 接口内容采用类 url 的字符串格式, 接口反馈数据统一采用 **json** 格式, 同级字段间**没有**前后关系, 解析时请使用 json 的标准解析方式。

连接方式一: TCP 连接

网络设置: TCP 客户端

服务器 IP 地址:

- a) 192.168.10.10(这个是底盘主机的静态 IP 地址, TCP 客户端主机需要通过路由器等设备与底盘主机建立局域网连接, 有相同网段 192.168.10.*)
- b) 底盘通过 API 连接上局域网 WiFi, 然后通过 API 获取的局域网 IP。

服务器端口: 31001

连接方式二: 串口连接

需要底盘提供串口接口

建议使用 **socket** 连接方式, 因为串口通信有一定概率使数据传输错误。

接口使用说明:

通用解释:

- 通用参数“uuid”
 - 如不做特殊说明, 以下指令中均可携带 uuid 作为可选参数, server 端执行完指令后在 **response** 中将其原样返回。
- 对于 json 中“type”字段的解释:
 - “response” 表示对一条指令的响应
 - “callback” 表示由 server 端以一定频率发送的数据
 - “notification” 表示由 server 端发起的主动通知

示例:

客户端发送字符串: "/api/move?marker=target_name&uuid=123456"

返回:

```
{
  "type": "response",    /* package 类型 */
  "command": "/api/move", /* 源指令 */
  "error_message": "",   /* 错误消息 */
  "status": "OK",        /* 执行结果 */
  "uuid": "123456"       /* 用户定义的 uuid */
}
```

1. 机器人移动功能

指令:

/api/move

参数:

名称	说明	是否必选	备注
marker	目标点位代号	与 location 必选其一	优先级低于 location
location	x<地图中 x 轴坐标>,y<地图中 y 轴坐标>,theta<地图中相对 theta 值>	与 marker 必选其一	

接口描述:

可使机器人从当前位置移动至地图中已经标定的某个目标点。

协议中对于移动指令部分包括两大主要的形式:

- 对于需要使机器人在运动过程中使用自动导航和避障功能的, 请使用此接口;
- 如果需要使用诸如手柄遥控功能的, 请使用[接口 6](#), 会直接操控机器人两轮的速度;

在使用此接口之前还有必要学习扫图工作, 需要使用者提前安排(智能佳提供一套通过局域网连接机器人, 然后登入浏览器进行扫图的方案, 具体操作参考《智能佳移动机器人用户使用手册》)。

参数中的 location 的坐标是相对于“地图”坐标系的, 由于直接调用(x、y、theta)在使用中不够直观和方便, 智

能佳提供“事先标锚点，而后使机器人重返锚点”的工作形式。之后“锚点”统一用 **marker** 代称，设置和查询已设锚点的方式见[接口 4](#)、[接口 5](#)。

机器人采用自动导航和避障的过程中，机器人会自动规划路径和调节速度，过程中不需要操作者干预。操作者可以监听 **robot_status**(机器人整体和行进状态)，或者等待相应的 **notification**，以了解任务完成情况和整机状态。

示例:

发送:

```
/api/move?marker=target_name  
// 调用移动接口, 移动至代号为"target_name"的目标点  
/api/move?location=15.0,4.0,1.5707963  
// 调用移动接口, 移动至 location(15.0, 4.0,  $\pi/2$ )的目标点
```

返回:

```
{  
  "type": "response",  
  "command": "/api/move",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

2.移动取消功能

指令:

```
/api/move/cancel
```

参数:

N/A

接口描述:

使机器人主动放弃当前正在执行的移动任务，成功取消后可使机器人进入新的待命状态。

在机器人执行[接口 1](#)-机器人移动命令过程中，如果需要终止机器人当前的移动状态，可以调用此接口。机器人会在接收“移动取消”命令之后，原地停止，等待再次的 **move** 指令。

示例:

发送:

```
/api/move/cancel  
// 取消当前正在进行的移动指令
```

返回:

```
{  
  "type": "response",  
  "command": "/api/move/cancel",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

3.获取机器人当前全局状态

本接口作为兼容用户以前的软件版本保留， 新用户推荐使用[接口 9](#)——请求机器人实时数据。

指令:

```
/api/robot_status
```

参数:

N/A

接口说明:

获取机器人当前全局状态，包括移动任务的状态。

除配合[接口 1](#) 周期监听 **move** 的反馈之外，也可从此接口中监听机器人整机的其他信息，包括“是否处于充电

状态”、“是否处于急停状态”、“所剩电量百分比”、“当前相对于地图的位置”、“当前楼层”，具体见示例。
建议调用频率为 1-2HZ，可以实时监控任务状态，作为流程控制的逻辑判断。

示例：

发送：

```
/api/robot_status  
// 获得机器人当前的全局状态
```

返回：

```
{  
  "type": "response",  
  "command": "/api/robot_status",  
  "uuid": "",  
  "status": "OK",  
  "error_message": "",  
  "results": {  
    "move_target": "target_name",  
    "move_status": "running",  
    "move_retry_times": 3, //此次数每增加1, 表示机器人进行了新一轮的路径重试; 路径规划一次性成功此值默认为0  
  
    "charge_state": true, //true->充电中状态。false->未充电状态。  
    "soft_estop_state": true, // 通过API 接口设置的急停状态  
    "hard_estop_state": false, // 通过硬件急停按钮设置的急停状态  
    "estop_state": true, // hard_estop_state || soft_estop_state  
    "power_percent": 100, //电量百分比, 单位: %  
    "current_pose": {  
      "x": 11.0,    // 单位: m  
      "y": 11.0,    // 单位: m  
      "theta": 0.5, //单位: rad  
    }  
    "current_floor": 16  
  }  
}
```

返回结果字段解释：

结果字段	解释
move_target	目标点别名(marker)

结果字段	解释
move_status	任务状态(running/idle/failed/succeeded/canceled)

4.插入新的 marker 点位

指令:

```
/api/markers/insert
```

参数:

名称	说明	是否必选	备注
name	点位名字	必选	如果 name 已经标记过, 则更新坐标。
type	点位类型: 0(一般点位), 11 (充电桩)。	可选	默认为 0

接口说明:

在当前地图中标记点位, 让机器人记住这个点位 (marker)。

调用此接口时, 机器人会捕捉当前相对于地图坐标系的(x,y,theta), 并自动把捕捉到的锚点位置, 存储在机器人的锚点列表中, 以随时响应 move 的调用。如果不使用充电功能, 接口中的 type 可以不设置。

示例:

发送:

```
/api/markers/insert?name=205_room  
//调用该接口, 将机器人当前位置设为marker 点位, 名字为 205_room。默认为1层一般点位。  
/api/markers/insert?name=charge_dock_2&type=11  
//调用该接口, 将机器人当前位置设为marker 点位, 名字为 charge_dock_2, 类型为充电桩。
```

返回:

操作成功时返回:

```
{  
  "type": "response",  
  "command": "/api/markers/insert",
```

```
"uuid": "",  
"status": "OK",  
"error_message": ""  
}
```

5. 获取 marker 点位列表

指令:

```
/api/markers/query_list
```

参数:

N/A

接口说明:

获取机器人在某场景地图中的所有点位(markers)信息。

通过[接口 4](#)设置的锚点位置，可以通过此接口进行 json 格式的锚点列表查阅。**results** 内容以字典形式查阅，其中目前包括信息有“楼层”、“点位坐标”、“显示名称”和“类型”。

location 中的数据一般用户无需使用，它以四元数的方式保存机器人的角度，可以转化成前文阐述的坐标系中的 **theta**

示例:

发送:

```
/api/markers/query_list
```

返回:

```
{  
  "type": "response",  
  "command": "/api/markers/query_list",  
  "uuid": "",  
  "status": "OK",  
  "error_message": "",  
  "results":{  
    www.bjrobot.com
```

```
"meeting_room": {
  "floor": 1,
  "pose": {
    "orientation": {
      "w": -0.899999976158142,
      "x": 0,
      "y": 0,
      "z": -0.430000007152557
    },
    "position": {
      "x": -8.57999992370605,
      "y": 6.3600001335144,
      "z": 0
    }
  },
  "marker_name": "meeting_room",
  "key": 0
},
"205_room": {
  "floor": 1,
  "pose": {
    "orientation": {
      "w": -0.899999976158142,
      "x": 0,
      "y": 0,
      "z": -0.430000007152557
    },
    "position": {
      "x": -8.57999992370605,
      "y": 6.3600001335144,
      "z": 0
    }
  },
  "marker_name": "205_room",
  "key": 0
}
}
```

// "marker_name"表示点位名称。"key"表示点位类型: 0(一般点位), 11(充电桩)。“floor”表示点位所属楼层。“pose”表示具体坐标点。

6. 机器人直接控制指令

指令：

/api/joy_control

参数：

名称	说明	是否必选	备注
angular_velocity	角速度	必选	
linear_velocity	线速度	必选	

接口说明：

对机器人进行部分的直接控制，如自转或停止（此控制优先级高于 move 指令），返回 succeeded 表示机器人已成功接收并开始运行此指令

接口 1 已提供了机器人带路径规划和自动避障的移动功能，相对而言，此接口提供人工控制的遥控移动功能。遥控的方式为控制机器人行进的线速度和角速度，其单位分别为 m/s 和 rad/s；通过两值设置可以自行控制转弯半径等机器人运动表现形式

机器人运动方向和角速度、线速度符号说明如下：

线速度	角速度	机器人行进表现	备注
0	正	机器人原地左转	机器人角速度设值范围为 (-1.0 ~ 1.0)rad/s
0	负	机器人原地右转	
正	0	机器人前进	机器人线速度设值范围为 (-0.5 ~ 0.5)m/s
负	0	机器人后退	
正	正	机器人前进且左转	
正	负	机器人前进且右转	
负	正	机器人倒车且左转	
负	负	机器人倒车且右转	

如果传入的参数超出限定范围，模块内部会按最大预设工作

从安全方面考虑，单个命令的持续时间为 0.5 秒，但输入频率可以大于 2Hz。连续的指令发送可使机器人运动连贯，比如把命令映射到手柄控制键位上

示例：

发送：

```
/api/joy_control?angular_velocity=0.5&linear_velocity=0.2  
// 调用移动接口,机器人以角速度 0.5rad/s 逆时针转动，同时以线速度 0.2m/s 前进。
```

返回：

```
{  
  "type": "response",  
  "command": "/api/joy_control",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

7. 机器人急停控制指令

指令：

```
/api/estop
```

参数：

名称	说明	是否必选	取值范围	备注
flag	进入或退出急停模式	必选	true/false	

接口说明：

使机器人进入自由停止模式，机器人可被推动。

机器人硬件平台提供了一个急停控制按钮，按动此按钮可使机器人进行自动停车状态。除此硬件按钮之外，软件接口等价的提供一个急停控制指令，软件开启急停模式之后，也可使机器人进入自由停止模式。需要注意的是，软件和硬件的急停接口是分别控制机器人急停状态的，不可以相互解锁。

示例：

发送：

```
/api/estop?flag=true //进入急停模式  
/api/estop?flag=false //退出急停模式
```

返回：

```
{  
  "type": "response",  
  "command": "/api/estop",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

8. 机器人初始位置校定

指令：

```
/api/position_adjust
```

参数：

名称	说明	是否必选	取值范围	备注
marker	用以校定位置的 marker 名	必选	已经标定的 marker 点	

接口说明：

该接口用于机器人初始化自身位置，可将机器人推至已存入 marker 列表的任一位置，然后用指定的 marker 的坐标校定机器人当前位置。

示例:

发送:

```
/api/position_adjust?marker=001  
//告知机器人当前处于代号为001的marker点位上
```

返回:

```
{  
  "type": "response",  
  "command": "/api/position_adjust",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

9.请求机器人实时数据

本接口至少需要软件版本 v0.4.0

指令:

```
/api/request_data
```

参数:

名称	说明	是否必选	取值范围	备注
topic	请求的实时数据类型	必选	目前仅接受"robot_status"类型	
switch	开启/停止数据发送	必选	on/off	默认关闭状态
frequency	发送频率	可选	[1, 10]	默认 2HZ

接口说明:

请求 server 端以一定频率发送 topic 类型的数据。当请求成功后, server 端会以一定频率发送数据给请求的 client。

示例:

发送:

```
/api/request_data?topic=robot_status&switch=on&frequency=1  
// 请求 server 以 1HZ 的频率发送机器人全局状态
```

返回:

```
{  
  "type": "response",  
  "command": "/api/request_data",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

成功请求后, server 以 1HZ 的频率发送 **topic** 为 robot_status 的数据, type 为 **callback**

```
{  
  "type": "callback",  
  "topic": "robot_status",  
  "results":  
  {  
    ... // 详见接口 3 里“results”的内容和解释  
  }  
}
```

10. 机器人主动通知

接口说明:

机器人主动发出通知给所有连接的 client, 客户端在收到通知后可以进行相应的动作, 比如语音播报等。

注: 不建议作为流程控制的逻辑判断, 因为网络等原因可能会造成收不到通知。建议通过[接口 3](#) 获取机器人当前全局状态获取实时状态作为流程的逻辑判断。

通知列表:

code	description	备注	level	支持软件版本
通用移动任务:				
"01001"	The move task is started.	移动任务开始了。	info	0.4.0
"01002"	The move task is finished.	移动任务完成了。	info	0.4.0
"01003"	The move task is failed.	移动任务失败了。	info	0.4.0
"01004"	The move task is canceled.	移动任务被取消了。	info	0.4.0
"01005"	The move task is retried.	移动重试。	info	0.4.0
"01010"	Start to leave charging pile.	开始离开充电桩。	info	0.5.2
"01011"	Succeed to leave charging pile.	离开充电桩成功了。	info	0.5.2
"01012"	Failed to leave charging pile.	离开充电桩失败了。	warning	0.5.2
"01020"	Start to auto dock to charging pile.	开始自动停靠到充电桩。	info	0.5.2
"01021"	Succced to auto dock to charging pile.	自动停靠到充电桩上成功了。	info	0.5.2
"01022"	Failed to auto dock to charging pile.	自动停靠到充电桩上失败了。	warning	0.5.2
"01023"	Failed to to receive valid data.	回充电失败原因 1。	warning	0.5.10
"01024"	Failed to find any feature around the robot.	回充电失败原因 2。	warning	0.5.10
"01025"	Failed to catch power status.	回充电失败原因 3。	warning	0.5.10
"01026"	Failed to catch infrared signal.	回充电失败原因 4。	warning	0.5.10
电梯任务:				
"04000"	Start to go to lift.	开始去电梯门口。	info	0.5.2
"04001"	Succeed to go to lift.	去电梯门口成功了。	info	0.5.2
"04002"	Failed to go to lift.	去电梯门口失败了。	info	0.5.2
"04010"	Start to call lift.	开始呼叫电梯。	info	0.5.2
"04011"	Succeed to call lift.	呼叫电梯成功了。	info	0.5.2

code	description	备注	level	支持软件版本
"04013"	Call lift more than 3 minutes.	呼叫电梯超过 3 分钟了还没来。	warning	0.5.2
"04020"	Start to take lift.	开始乘坐电梯。	info	0.5.2
"04021"	Succeed to take lift.	乘坐电梯成功，准备出电梯。	info	0.5.2
"04023"	Take lift more than 3 minutes."	乘坐电梯超过 3 分钟了还没到。	warning	0.5.2
"04030"	Start to enter lift.	开始进电梯。	info	0.5.2
"04031"	Succeed to enter lift.	进电梯成功。	info	0.5.2
"04032"	Failed to enter lift.	进电梯失败。	info	0.5.2
"04040"	Start to avoid lift.	进电梯失败，开始回避电梯。	info	0.5.2
"04041"	Finish to avoid lift.	回避电梯成功。	info	0.5.2
"04050"	Start to exit lift.	开始出电梯。	info	0.5.2
"04051"	Succeed to exit lift.	出电梯成功了。	info	0.5.2
"04052"	Failed to exit lift.	出电梯失败了。	warning	0.5.2
"04060"	Start to back to lift.	出电梯失败，开始回到电梯。	info	0.5.2
"04061"	Succeed to back to lift.	回到电梯成功了。	info	0.5.2
"04062"	Failed to back to lift.	回到电梯失败了。	warning	0.5.2
状态相关:				
"02000"	Poweroff notice.	将关机断电	"info"	0.5.6
"02001"	Charge status on.	未充电状态=>充电状态	"info"	0.5.6
"02002"	Charge status off.	充电状态=>未充电状态	"info"	0.5.6
"02003"	Estop on.	未急停状态=>急停状态	"info"	0.5.6
"02004"	Estop off.	急停状态=>未急停状态	"info"	0.5.6

字段解释:

字段	说明	备注
----	----	----

字段	说明	备注
code	通知的具体代码	后续将扩展新的通知，可以作为执行逻辑的参考
description	代码的解释	不能作为参考，可作为打印信息
level	通知的级别	限定为 info/warning/error 类型，每个 code 的划分以后可能会改
备注	中文解释，仅存在于文档内	

示例：

```
{
  "type": "notification",
  "code": "01001"
  "level": "info"
  "description": "The move task is started."
}
```

11.设置参数

本接口至少需要软件版本 v0.5.2

指令：

```
/api/set_params
```

参数：

名称	说明	是否必选	取值范围	备注	软件版本
max_speed_ratio	机器人最大行进速度百分比	可选	[0.3, 1.0]	小于 0.3 取 0.3，大于 1.0 取 1.0	v0.5.2

接口说明：

设置机器人的参数。

需要注意的是：调用此接口后无论是否成功，返回的 json 中“status”字段都是“OK”。如果想确定参数是否设置成功，请调用接口 12 获取参数获取参数后查看参数的当前值。

示例:

发送:

```
/api/set_params?max_speed_ratio=0.5  
// 设置机器人最大行进速度为0.5
```

返回:

```
{  
  "type": "response",  
  "command": "/api/set_params",  
  "uuid": "",  
  "status": "OK",  
  "error_message": ""  
}
```

12. 获取参数

本接口至少需要软件版本 v0.4.2

指令:

```
/api/get_params
```

参数:

N/A

接口说明:

获取机器人可设置的参数列表以及当前值。各版本支持的参数见接口 11 设置参数。

示例:

发送:

```
/api/get_params  
// 获取参数列表和当前值
```

返回:

```
{
  "type": "response",
  "command": "/api/get_params",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": {
    "max_speed_ratio": 0.5
  }
}
```

13.无线网络接口

13.1 获取机器人当前可用的 WiFi 列表

本接口至少需要软件版本 v0.5.2

指令:

```
/api/wifi/list
```

参数:

N/A

接口说明:

获取机器人当前可用的 WiFi 列表，返回中包含 SSID 和信号强度。

示例:

发送:

```
/api/wifi/list
```

返回:

```
{
```

```
"type": "response",  
"command": "/api/wifi/list",  
"uuid": "",  
"status": "OK",  
"error_message": "",  
"results": {  
  "SSID1": 50,  
  "SSID2": 30,  
  "SSID3": 80,  
}  
}
```

13.2 连接 WiFi

本接口至少需要软件版本 v0.5.2

指令:

```
/api/wifi/connect
```

参数:

名称	说明	是否必选	取值范围	备注	软件版本
SSID	WiFi 的 SSID	必选	当前的环境 WiFi		v0.5.2
password	WiFi 密码	可选	SSID 对应	如果已经连接过，可以不填	v0.5.2

接口说明:

使机器人连接到环境 WiFi 或切换当前连接的环境 WiFi。

示例:

发送:

```
/api/wifi/connect?SSID=SSID1&password=123456
```

返回:

```
{  
  "type": "response",
```

```
"command": "/api/wifi/connect",  
"uuid": "",  
"status": "OK",  
"error_message": "",  
}
```

13.3 获取当前连接的 WiFi

本接口至少需要软件版本 v0.5.2

指令:

```
/api/wifi/get_active_connection
```

参数:

N/A

接口说明:

获取机器人当前连接的 WiFi 的 SSID.

示例:

发送:

```
/api/wifi/get_active_connection
```

返回:

```
{  
  "type": "response",  
  "command": "/api/wifi/get_active_connection",  
  "uuid": "",  
  "status": "OK",  
  "error_message": "",  
  "results": "SSID1" // 如果当前未连接到WiFi, 则results为空"  
}
```

13.4 获取机器人 IP 和无线网卡地址

本接口至少需要软件版本 v0.5.2

指令：

```
/api/wifi/info
```

参数：

N/A

接口说明：

获取机器人当前通过环境 WiFi 分配到的 IP 地址和无线网卡的物理地址。

示例：

发送：

```
/api/wifi/info
```

返回：

```
{
  "type": "response",
  "command": "/api/wifi/info",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": {
    "IPaddr": "192.168.XXX.XXX", // 如果未连接WiFi 则为空""
    "HWaddr": "xx:xx:xx:xx:xx:xx", // 连接WiFi 网卡的物理地址
  },
}
```

14.地图接口

14.1 获取地图列表

本接口至少需要软件版本 v0.5.2

指令:

```
/api/map/list
```

参数:

N/A

接口说明:

获取机器人中所有的地图名称和楼层。

示例:

发送:

```
/api/map/list
```

返回:

```
{
  "type": "response",
  "command": "/api/wifi/info",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": {
    "map_name_1": [1,2,3,4,5], // 地图map_name_1中有1-5层
    "map_name_2": [10],
  },
}
```

14.2 设置当前地图

本接口至少需要软件版本 v0.5.2

指令：

```
/api/map/set_current_map
```

参数：

名称	说明	是否必选	备注	软件版本
hotel_id	地图名	必选		v0.5.2
floor	楼层	必选		v0.5.2

接口说明：

设置机器人当前地图。

注:设置成功后会重启 *water* 服务，所以有可能收不到 *response*。

示例：

发送：

```
/api/map/set_current_map?hotel_id=map_name_1&floor=5
```

返回：

```
{
  "type": "response",
  "command": "/api/map/set_current_map",
  "uuid": "",
  "status": "OK",
  "error_message": "",
}
```

14.3 获取当前地图

本接口至少需要软件版本 v0.5.2

指令:

```
/api/map/get_current_map
```

参数:

N/A

接口说明:

获取机器人当前地图。

示例:

发送:

```
/api/map/get_current_map
```

返回:

```
{
  "type": "response",
  "command": "/api/map/get_current_map",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": {
    "hotel_id": "map_name_1",
    "floor": "5"
  },
}
```

15. 关机重启接口

本接口至少需要软件版本 v0.5.10

指令:

```
/api/shutdown
```

参数:

名称	说明	是否必选	取值范围	备注	软件版本
reboot	关机后是否重启	可选	true/false	缺省为 false	v0.5.10

接口说明:

调用接口关闭或者重新启动机器人,在电源关闭之前会有通知发出(见[接口 10](#) 机器人主动通知), 在通知发出后 10s 电源关闭。如果是重启的话, 重新上电与电源关闭间会有 5s 间隔。

注:可能会收不到 response。

示例:

发送:

```
/api/shutdown // 关机
```

或者

```
/api/shutdown?reboot=true // 重启
```

返回:

```
{
  "type": "response",
  "command": "/api/shutdown",
  "uuid": "",
  "status": "OK",
  "error_message": "",
}
```

16.软件更新接口

16.1 获取当前软件版本

本接口至少需要软件版本 v0.5.5

指令:

```
/api/software/get_version
```

参数:

N/A

接口说明:

获取当前软件版本。

示例:

发送:

```
/api/software/get_version
```

成功返回:

```
{
  "type": "response",
  "command": "/api/software/get_version",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": "x.x.x",    // 当前软件版本
}
```

失败返回:

```
{
  "type": "response",
  "command": "/api/software/get_version",
  "uuid": "",
  "status": "UNKNOWN_ERROR",
  "error_message": "xxx",    // 失败原因
}
```

16.2 检查更新

本接口至少需要软件版本 v0.5.5

指令:

```
/api/software/check_for_update
```

参数:

N/A

接口说明:

检查是否有新版本可以更新，返回当前版本、最新版本和是否可以更新。

注:

1. 本功能需要机器人接入网络，否则返回失败。
2. 本接口是阻塞执行。

示例:

发送:

```
/api/software/check_for_update
```

成功返回:

```
{
  "type": "response",
  "command": "/api/software/check_for_update",
  "uuid": "",
  "status": "OK",
  "error_message": "",
  "results": {
    "version_latest": "x.x.x", // 当前版本
    "version_current": "x.x.x", // 最新版本
    "enable_update": bool, // 能否更新, true->能更新, false->不能更新
  },
}
```

失败返回:

```
{
  "type": "response",
  "command": "/api/software/check_for_update",
  "uuid": "",
  "status": "UNKNOWN_ERROR",
  "error_message": "xxx", // 错误消息
}
```

16.3 更新

本接口至少需要软件版本 v0.5.5

www.bjrobot.com

指令：

```
/api/software/update
```

参数：

N/A

接口说明：

更新软件到**最新版本**。

注：

1. 本功能需要机器人接入网络，否则返回失败。
2. 本接口是阻塞执行。
3. 升级成功后将自动重启软件服务，所有 tcp socket 需要重新连接。

示例：

发送：

```
/api/software/update
```

成功返回：

```
{
  "type": "response",
  "command": "/api/software/update",
  "uuid": "",
  "status": "OK",
  "error_message": "",
}
```

失败返回：

```
{
  "type": "response",
  "command": "/api/software/update",
  "uuid": "",
  "status": "UNKNOWN_ERROR",
  "error_message": "xxx",    // 错误消息
}
```

a. 状态代码返回说明（针对返回数据的 status）

返回码	定义描述
OK	表示响应包含有效的 results
INVALID_REQUEST	表示提供的请求无效。这一状态的常见原因包括参数或参数值无效
REQUEST_DENIED	表示系统已拒绝您的请求
UNKNOWN_ERROR	表示由于服务器发生错误而无法处理请求。如果您重试一次，请求可能会成功

b. 错误消息返回说明（针对返回数据的 error_message）

如果状态代码不是 OK，在响应对象包含附加的 error_message 字段中更详细地说明了给定状态代码背后的原因。

注：此字段不保证始终有内容，并且其内容可能会更改。

generated by haroopad