橙家智能科技有限公司

**[Home Smart Box Wi-Fi Device Protocol]**

Revision: v00.00.01

Release Date: 2016-03-19

Document Revision

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Author** | **Description** |
| 00.00.01 | 2016-03-19 |  | Initial created. |
|  |  |  |  |

Table of Contents

[**1** **Scope** 4](#_Toc446247514)

[**2** **Requirements** 5](#_Toc446247515)

[**3** **Reference** 6](#_Toc446247518)

[**4** **Conventions** 7](#_Toc446247519)

[**5** **Scenario** 8](#_Toc446247520)

[**5.1** **Smart Configure Stage** 8](#_Toc446247521)

[**5.2** **Device Discover Stage** 8](#_Toc446247522)

[**5.3** **Normal Stage** 8](#_Toc446247523)

[**6** **Protocol** 9](#_Toc446247524)

[**6.1** **Device Discover (0x8901)** 9](#_Toc446247525)

[**6.2** **Device Discover Response (0x8902)** 9](#_Toc446247526)

[**6.3** **Get Info (0x8911)** 9](#_Toc446247527)

[**6.4** **Get Info Response (0x8912)** 10](#_Toc446247528)

[**6.5** **Get Status (0x8921)** 10](#_Toc446247529)

[**6.6** **Get Status Response (0x8922)** 11](#_Toc446247530)

[**6.7** **Set Status (0x8923)** 11](#_Toc446247531)

[**6.8** **Status Changed (0x8924)** 12](#_Toc446247532)

[**6.9** **Event (0x8931)** 12](#_Toc446247533)

[**6.10** **Set Action (0x8941)** 12](#_Toc446247534)

[**6.11** **Keep Alive (0x8951)** 13](#_Toc446247535)

[**6.12** **Result (0x89A1)** 13](#_Toc446247536)

[**7** **Definition** 13](#_Toc446247537)

[**7.1** **Device Class** 13](#_Toc446247538)

[**7.2** **Device Interface** 14](#_Toc446247539)

[**7.3** **Device Status** 14](#_Toc446247540)

[**7.3.1** **STATUS\_BM** 15](#_Toc446247541)

[**7.3.2** **STATUS\_TLV** 16](#_Toc446247542)

[**7.4** **Device Event** 16](#_Toc446247543)

[**7.4.1** **EVENT\_BM** 17](#_Toc446247544)

[**7.4.2** **EVENT\_TLV** 17](#_Toc446247545)

[**7.5** **Device Action** 17](#_Toc446247546)

[**7.5.1** **ACTION\_BM** 18](#_Toc446247547)

[**7.5.2** **ACTION\_TLV** 18](#_Toc446247548)

[**7.6** **Error Code** 18](#_Toc446247549)

[**8** **Encryption** 20](#_Toc446247550)

1. **Scope**

本文档用于描述HSB和Device之间的通信方式和协议

* HSB (Home Smart Box)：家庭智能盒子，是智能家居的核心，下文简称BOX
* Device：智能设备

1. **Requirements**

4. **Reference**
5. **Conventions**

本文档描述的通信协议全部使用小端字节序(Little Endian)，以及UDP通信方式

本文档部分协议中采用TLV(Type Length Value)的方式来编码，具体参考Definition

本文档中协议格式有一些相同的约定，如下描述

|  |  |  |
| --- | --- | --- |
| CMD | | LEN |
| FIELD1 | FIELD2 | RESERVED |

Packet头部由16 Bit的CMD和16 Bit的LEN组成，整个Packet按4 Byte对齐，不足部分补0

CMD – Packet的类型

LEN – Packet的长度

RESERVED – 为了扩展预留的字段，或者仅仅是为了对齐

1. **Scenario**
   1. **Smart Configure Stage**
   2. **Device Discover Stage**

BOX启动之后，开始发送UDP广播包Device Discover到端口19001；Device收到之后回复Device Discover Response，这样BOX可以得到Device的IP地址，随后进入Normal Stage。

Device Discover

Device Discover Resp

Device

BOX

* 1. **Normal Stage**

Device在19001端口监听，BOX在19002端口监听

通信有二种方式，

1. BOX向Device发送消息，Device回应消息，大部分消息遵循此方式
2. Device主动发送消息给BOX，如状态改变、传感器触发等
3. **Protocol**
   1. **Device Discover (0x8901)**

Purpose: BOX寻找局域网内的Device

Reply: Device Discover Response

Direction: BOX to Device

DIP: 局域网广播地址

(例如，BOX IP：192.168.1.100，Netmask: 255.255.255.0则广播地址为192.168.1.255)

DPORT: 19001

SIP: ANY

SPORT: ANY

32

16

0

|  |  |  |
| --- | --- | --- |
| CMD (0x8901) | | LEN |
| Ver Minor | Ver Major | RESERVED |

Ver Major – BOX软件主版本号

Ver Minor – BOX软件二级版本号

* 1. **Device Discover Response (0x8902)**

Purpose: Device回应来自BOX的Device Discover

Reply: None

Direction: Device to BOX

DIP: “Device Discover” Packet的SIP

DPORT: “Device Discover” Packet的SPORT

SIP: ANY

SPORT: ANY

|  |  |  |
| --- | --- | --- |
| CMD (0x8902) | | LEN |
| Ver Minor | Ver Major | RESERVED |
| CLASS | | INTERFACE |
| MAC[4] | | |
| MAC[2] | | RESERVED |
| STATUS\_BM | | |
| EVENT\_BM | | |
| ACTION\_BM | | |

Ver Major – Device软件主版本号

Ver Minor – Device软件二级版本号

* 1. **Get Info (0x8911)**

Purpose: BOX获取Device的信息

Reply: Get Info Response

Direction: BOX to Device

|  |  |
| --- | --- |
| CMD (0x8911) | LEN |

* 1. **Get Info Response (0x8912)**

Purpose: Device回应来自BOX的Get Info消息

Reply: None

Direction: Device to BOX

|  |  |
| --- | --- |
| CMD (0x8912) | LEN |
| CLASS | INTERFACE |
| MAC[4] | |
| MAC[2] | RESERVED |
| STATUS\_BM | |
| EVENT\_BM | |
| ACTION\_BM | |

CLASS – 表示Device类型，参考Definition章节

INTERFACE – 表示Device通信接口，参考Definition章节

MAC[4] – MAC地址前4个字节

MAC[4] – MAC地址后2个字节

STATUS\_BM – 表示Device支持哪些状态类型，参考Definition章节

EVENT\_BM – 表示Device支持哪些事件类型，参考Definition章节

ACTION\_BM – 表示Device支持哪些动作类型，参考Definition章节

* 1. **Get Status (0x8921)**

Purpose: BOX获取Device的状态

Reply: Get Status Response

Direction: BOX to Device

|  |  |
| --- | --- |
| CMD (0x8921) | LEN |
| STATUS\_BM | |

STATUS\_BM – 表示Device支持哪些状态类型。如果想获取所有状态，可以简单的设置为全1

* 1. **Get Status Response (0x8922)**

Purpose: Device回应来自BOX的Get Status消息

Reply: None

Direction: Device to BOX

|  |  |
| --- | --- |
| CMD (0x8922) | LEN |
| STATUS TLV | |

Status TLV参考Definition章节

* 1. **Set Status (0x8923)**

Purpose: BOX设置Device的状态

Reply: Result

Direction: BOX to Device

|  |  |
| --- | --- |
| CMD (0x8923) | LEN |
| STATUS TLV | |

Status TLV参考Definition章节

Device收到Set Status消息之后，只需要回复Result消息；当Status发生改变之后，Device还是要用”Status Changed”消息通知BOX

* 1. **Status Changed (0x8924)**

Purpose: Device通知BOX，某些状态发生了改变

Reply: Result

Direction: Device to BOX

|  |  |
| --- | --- |
| CMD (0x8924) | LEN |
| STATUS TLV | |

Status TLV参考Definition章节

* 1. **Event (0x8931)**

Purpose: Device通知BOX某些事件发生了

Reply: Result

Direction: Device to BOX

|  |  |
| --- | --- |
| CMD (0x8931) | LEN |
| EVENT TLV | |

Event TLV参考Definition章节

* 1. **Set Action (0x8941)**

Purpose: BOX设置Device进行某个动作

Reply: Result

Direction: BOX to Device

|  |  |
| --- | --- |
| CMD (0x8941) | LEN |
| ACTION TLV | |

Action TLV参考Definition章节

* 1. **Keep Alive (0x8951)**

Purpose: BOX以UDP广播的方式查看Device连接状态，Device回复BOX

Reply: Keep Alive

Direction: BOX to Device，Device reply to BOX

|  |  |
| --- | --- |
| CMD (0x8951) | LEN |

* 1. **Result (0x89A1)**

Purpose: 回应来自对方的消息，仅表示收到

Reply: None

Direction: BOX to Device，Device to BOX

|  |  |
| --- | --- |
| CMD (0x89A1) | LEN |
| ERROR\_CODE | RESERVED |

Error Code参考Definition章节

1. **Definition**
   1. **Device Class**

|  |  |  |
| --- | --- | --- |
| ID | Device Class | Description |
| 0 | HSB\_DEV\_CLASS\_SOCKET | 插座 |
| 1 | HSB\_DEV\_CLASS\_SWITCH | 开关 |
| 2 | HSB\_DEV\_CLASS\_LAMP | 灯 |
| 3 | HSB\_DEV\_CLASS\_SENSOR | 传感器 |
| 4 | HSB\_DEV\_CLASS\_REMOTE\_CONTROL | 遥控器 |
| 5 | HSB\_DEV\_CLASS\_CAMERA | 摄像头 |
| 6 | HSB\_DEV\_CLASS\_SOUND\_BOX | 音箱 |
| 7 | HSB\_DEV\_CLASS\_DOORBELL | 门铃 |
| 8 | HSB\_DEV\_CLASS\_ENTRANCE\_GURAD | 门禁 |
| 9 | HSB\_DEV\_CLASS\_TV | 电视 |
| 10 | HSB\_DEV\_CLASS\_AIR\_CONDITIONING | 空调 |
| 11 | HSB\_DEV\_CLASS\_CURTAIN | 窗帘 |
| 12 | HSB\_DEV\_CLASS\_FRIDGE | 冰箱 |
| 13 | HSB\_DEV\_CLASS\_AIR\_CLEANER | 空气净化器 |
| 14 | HSB\_DEV\_CLASS\_WATER\_HEATER | 热水器 |
| 15 | HSB\_DEV\_CLASS\_OVEN | 烤箱 |
| 16 | HSB\_DEV\_CLASS\_MICROWAVE\_OVEN | 微波炉 |
| 17 | HSB\_DEV\_CLASS\_LAMPBLACK | 油烟机 |
| 18 | HSB\_DEV\_CLASS\_FAN | 风扇 |
| 19 | HSB\_DEV\_CLASS\_WARMING\_OVEN | 暖炉 |
| 20 | HSB\_DEV\_CLASS\_HEATING | 暖气 |
| 21 | HSB\_DEV\_CLASS\_STB | 机顶盒 |
| 22 | HSB\_DEV\_CLASS\_BRACELET | 手环 |
| 23 | HSB\_DEV\_CLASS\_CUP | 杯子 |
| 24 | HSB\_DEV\_CLASS\_ROUTER | 路由器 |
| 25 | HSB\_DEV\_CLASS\_WASHER | 洗衣机 |
| 26 | HSB\_DEV\_CLASS\_ELECTRIC\_COOKER | 电饭煲 |
| 27 | HSB\_DEV\_CLASS\_FAX | 传真机 |
| 28 | HSB\_DEV\_CLASS\_PRINTER | 打印机 |

Table 1 Device Class Definition

* 1. **Device Interface**

|  |  |  |
| --- | --- | --- |
| ID | Device Interface | Description |
| 0 | HSB\_DEV\_INTERFACE\_WIFI | WIFI |
| 1 | HSB\_DEV\_INTERFACE\_ZIGBEE | Zigbee |
| 2 | HSB\_DEV\_INTERFACE\_BLUETOOTH | Blutooth |
| 3 | HSB\_DEV\_INTERFACE\_IR | 红外 |
| 4 | HSB\_DEV\_INTERFACE\_NFC | NFC |
| 5 | HSB\_DEV\_INTERFACE\_AM | AM |
| 6 | HSB\_DEV\_INTERFACE\_FSK | FSK |

Table 2 Device Interface Definition

* 1. **Device Status**

本协议规定：凡是可以设置、并且可以被获得的变量称为“可控状态(Status)”

凡是只能被获得，但是不能被设置的变量称为“只读状态(RO Status)”

例如，On/Off状态、灯光亮度等，可以被读取和设置的，属于可控状态；

门窗传感器的开合，可以被读取但不能被设置，属于只读状态

但是本协议并不定义具体状态是什么，实际的状态请参考具体产品的Spec。

HSB软件的驱动层，会根据Device Spec和本协议，把这里的状态定义和实际的状态定义对应起来，例如，

HSB\_DEV\_STATUS\_1 -> HSB\_DEV\_STATUS\_BRIGHTNESS

HSB\_DEV\_STATUS\_2 -> HSB\_DEV\_STATUS\_VOLUME

HSB\_DEV\_RO\_STATUS\_0 -> HSB\_DEV\_STATUS\_DW\_SENSOR

|  |  |  |
| --- | --- | --- |
| ID | Device Status | Description |
| 0 | HSB\_DEV\_STATUS\_ON\_OFF | 开启/关闭状态 |
| 1 | HSB\_DEV\_STATUS\_1 |  |
| 2 | HSB\_DEV\_STATUS\_2 |  |
| 3 | HSB\_DEV\_STATUS\_3 |  |
| 4 | HSB\_DEV\_STATUS\_4 |  |
| 5 | HSB\_DEV\_STATUS\_5 |  |
| 6 | HSB\_DEV\_STATUS\_6 |  |
| 7 | HSB\_DEV\_STATUS\_7 |  |
| 8 | HSB\_DEV\_STATUS\_8 |  |
| 9 | HSB\_DEV\_STATUS\_9 |  |
| 10 | HSB\_DEV\_STATUS\_10 |  |
| 11 | HSB\_DEV\_STATUS\_11 |  |
| 12 | HSB\_DEV\_STATUS\_12 |  |
| 13 | HSB\_DEV\_STATUS\_13 |  |
| 14 | HSB\_DEV\_STATUS\_14 |  |
| 15 | HSB\_DEV\_STATUS\_15 |  |
| 16 | HSB\_DEV\_RO\_STATUS\_0 |  |
| 17 | HSB\_DEV\_RO\_STATUS\_1 |  |
| 18 | HSB\_DEV\_RO\_STATUS\_2 |  |
| 19 | HSB\_DEV\_RO\_STATUS\_3 |  |
| 20 | HSB\_DEV\_RO\_STATUS\_4 |  |
| 21 | HSB\_DEV\_RO\_STATUS\_5 |  |
| 22 | HSB\_DEV\_RO\_STATUS\_6 |  |
| 23 | HSB\_DEV\_RO\_STATUS\_7 |  |
| 24 | HSB\_DEV\_RO\_STATUS\_8 |  |
| 25 | HSB\_DEV\_RO\_STATUS\_9 |  |
| 26 | HSB\_DEV\_RO\_STATUS\_10 |  |
| 27 | HSB\_DEV\_RO\_STATUS\_11 |  |
| 28 | HSB\_DEV\_RO\_STATUS\_12 |  |
| 29 | HSB\_DEV\_RO\_STATUS\_13 |  |
| 30 | HSB\_DEV\_RO\_STATUS\_14 |  |
| 31 | HSB\_DEV\_RO\_STATUS\_15 |  |

Table 3 Device Status Definition

* + 1. **STATUS\_BM**

STATUS\_BM是一个uint32\_t类型的数字，它的每个bit分别和Table 3 Device Status Definition 中的32种状态对应，用于表示Device支持的所有状态类型。

* + 1. **STATUS\_TLV**

用于表示状态，以(Type，Length，Value)的方式定义，如下图所示，

Type和Length各占一个Byte，Type参考Table 3 Device Status Definition，Length表示Value占用的Byte个数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type | Length | Value (Byte 0) | …… | Value (Byte N) |

* 1. **Device Event**

本协议规定：除了状态改变之外，Device主动通知HSB的事件，定义为Device Event

但是本协议并不定义具体Event是什么，实际的Event请参考具体产品的Spec。

HSB软件的驱动层，会根据Device Spec和本协议，把这里的Event定义和实际的Event定义对应起来，例如，

HSB\_DEV\_EVENT\_1 -> HSB\_DEV\_EVENT\_GAS\_SENSOR

HSB\_DEV\_EVENT\_2 -> HSB\_DEV\_EVENT\_IR\_SENSOR

|  |  |  |
| --- | --- | --- |
| ID | Device Event | Description |
| 0 | HSB\_DEV\_EVENT\_0 |  |
| 1 | HSB\_DEV\_EVENT\_1 |  |
| 2 | HSB\_DEV\_EVENT\_2 |  |
| 3 | HSB\_DEV\_EVENT\_3 |  |
| 4 | HSB\_DEV\_EVENT\_4 |  |
| 5 | HSB\_DEV\_EVENT\_5 |  |
| 6 | HSB\_DEV\_EVENT\_6 |  |
| 7 | HSB\_DEV\_EVENT\_7 |  |
| 8 | HSB\_DEV\_EVENT\_8 |  |
| 9 | HSB\_DEV\_EVENT\_9 |  |
| 10 | HSB\_DEV\_EVENT\_10 |  |
| 11 | HSB\_DEV\_EVENT\_11 |  |
| 12 | HSB\_DEV\_EVENT\_12 |  |
| 13 | HSB\_DEV\_EVENT\_13 |  |
| 14 | HSB\_DEV\_EVENT\_14 |  |
| 15 | HSB\_DEV\_EVENT\_15 |  |
| 16 | HSB\_DEV\_EVENT\_16 |  |
| 17 | HSB\_DEV\_EVENT\_17 |  |
| 18 | HSB\_DEV\_EVENT\_18 |  |
| 19 | HSB\_DEV\_EVENT\_19 |  |
| 20 | HSB\_DEV\_EVENT\_20 |  |
| 21 | HSB\_DEV\_EVENT\_21 |  |
| 22 | HSB\_DEV\_EVENT\_22 |  |
| 23 | HSB\_DEV\_EVENT\_23 |  |
| 24 | HSB\_DEV\_EVENT\_24 |  |
| 25 | HSB\_DEV\_EVENT\_25 |  |
| 26 | HSB\_DEV\_EVENT\_26 |  |
| 27 | HSB\_DEV\_EVENT\_27 |  |
| 28 | HSB\_DEV\_EVENT\_28 |  |
| 29 | HSB\_DEV\_EVENT\_29 |  |
| 30 | HSB\_DEV\_EVENT\_30 |  |
| 31 | HSB\_DEV\_EVENT \_31 |  |

Table 4 Device Event Definition

* + 1. **EVENT\_BM**

EVENT\_BM是一个uint32\_t类型的数字，它的每个bit分别和Table 4 Device Event Definition 中的32种Event对应，用于表示Device支持的所有Event类型。

* + 1. **EVENT\_TLV**

用于表示Event，以(Type，Length，Value)的方式定义，如下图所示，

Type和Length各占一个Byte，Type参考Table 4 Device Event Definition，Length表示Value占用的Byte个数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type | Length | Value (Byte 0) | …… | Value (Byte N) |

* 1. **Device Action**

本协议规定：BOX对Device的指令，定义为Device Action

不同于Set Status，Device Action仅仅指不会影响Status的动作，例如，

打开开关，会影响到ON\_OFF Status，属于Set Status

发出警报，没有对应的Status，属于Device Action

但是本协议并不定义具体Action是什么，实际的Action请参考具体产品的Spec。

BOX软件的驱动层，会根据Device Spec和本协议，把这里的Action定义和实际的Action定义对应起来，例如，

HSB\_DEV\_ACTION\_1 -> HSB\_DEV\_ACTION\_ALARM

HSB\_DEV\_ACTION\_2 -> HSB\_DEV\_ACTION\_BLINK

|  |  |  |
| --- | --- | --- |
| ID | Device Action | Description |
| 0 | HSB\_DEV\_ACTION\_0 |  |
| 1 | HSB\_DEV\_ACTION\_1 |  |
| 2 | HSB\_DEV\_ACTION\_2 |  |
| 3 | HSB\_DEV\_ACTION\_3 |  |
| 4 | HSB\_DEV\_ACTION\_4 |  |
| 5 | HSB\_DEV\_ACTION\_5 |  |
| 6 | HSB\_DEV\_ACTION\_6 |  |
| 7 | HSB\_DEV\_ACTION\_7 |  |
| 8 | HSB\_DEV\_ACTION\_8 |  |
| 9 | HSB\_DEV\_ACTION\_9 |  |
| 10 | HSB\_DEV\_ACTION\_10 |  |
| 11 | HSB\_DEV\_ACTION\_11 |  |
| 12 | HSB\_DEV\_ACTION\_12 |  |
| 13 | HSB\_DEV\_ACTION\_13 |  |
| 14 | HSB\_DEV\_ACTION\_14 |  |
| 15 | HSB\_DEV\_ACTION\_15 |  |
| 16 | HSB\_DEV\_ACTION\_16 |  |
| 17 | HSB\_DEV\_ACTION\_17 |  |
| 18 | HSB\_DEV\_ACTION\_18 |  |
| 19 | HSB\_DEV\_ACTION\_19 |  |
| 20 | HSB\_DEV\_ACTION\_20 |  |
| 21 | HSB\_DEV\_ACTION\_21 |  |
| 22 | HSB\_DEV\_ACTION\_22 |  |
| 23 | HSB\_DEV\_ACTION\_23 |  |
| 24 | HSB\_DEV\_ACTION\_24 |  |
| 25 | HSB\_DEV\_ACTION\_25 |  |
| 26 | HSB\_DEV\_ACTION\_26 |  |
| 27 | HSB\_DEV\_ACTION\_27 |  |
| 28 | HSB\_DEV\_ACTION\_28 |  |
| 29 | HSB\_DEV\_ACTION\_29 |  |
| 30 | HSB\_DEV\_ACTION\_30 |  |
| 31 | HSB\_DEV\_ACTION\_31 |  |

Table 5 Device Action Definition

* + 1. **ACTION\_BM**

ACTION\_BM是一个uint32\_t类型的数字，它的每个bit分别和Table 5 Device Action Definition中的32种Action对应，用于表示Device支持的所有Action类型。

* + 1. **ACTION\_TLV**

用于表示Action，以(Type，Length，Value)的方式定义，如下图所示，

Type和Length各占一个Byte，Type参考Table 5 Device Action Definition，Length表示Value占用的Byte个数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type | Length | Value (Byte 0) | …… | Value (Byte N) |

* 1. **Error Code**

|  |  |  |
| --- | --- | --- |
| ID | Error Code | Description |
| 0 | HSB\_E\_OK | OK |
| 1 | HSB\_E\_INVALID\_MSG | 非法消息 |
| 2 | HSB\_E\_NOT\_SUPPORT | 不支持 |
| 3 | HSB\_E\_BAD\_PARAM | 参数错误 |
| 4 | HSB\_E\_NO\_MEMORY | 内存不够 |
| 5 | HSB\_E\_OTHERS | 其他错误 |
| 6 | HSB\_E\_UNDEFINED\_6 | 未定义6 |
| 7 | HSB\_E\_UNDEFINED\_7 | 未定义7 |
| 8 | HSB\_E\_UNDEFINED\_8 | 未定义8 |
| 9 | HSB\_E\_UNDEFINED\_9 | 未定义9 |
| 10 | HSB\_E\_UNDEFINED\_10 | 未定义10 |
| 11 | HSB\_E\_UNDEFINED\_11 | 未定义11 |
| 12 | HSB\_E\_UNDEFINED\_12 | 未定义12 |
| 13 | HSB\_E\_UNDEFINED\_13 | 未定义13 |
| 14 | HSB\_E\_UNDEFINED\_14 | 未定义14 |
| 15 | HSB\_E\_UNDEFINED\_15 | 未定义15 |

Table 6 Error Code Definition

1. **Encryption**

使用BASE64方式加密和解密数据，参考：

http://baike.baidu.com/link?url=yJM50u-3b2m-fn8KdG0C0hwAbcYPcAZ8vehOqhPYU8gBo2HrZCWWX\_dk9YZPAmX9eiMkoJ0oMETzs8JuDo1VP\_