

BT01XW Bluetooth Communication Protocol

Version	Reviser	Time	Revision
V10	Zhiming Zhong	2020.10.29	1.Add lighting and beeping function.

Catalogue

1. Broadcast Data 3

1.1	Broadcast Data Format.....	3
1.2	Broadcast Data Detail Introduction.....	4
1.2.1	Broadcast Type.....	4
1.2.2	Manufacturer Custom Data	4
1.2.3	Device Name.....	4
1.2.4	Service UUID	4
2.	Service Custom&UUID	4
2.1	Battery Info Service.....	5
2.2	Data Transmission Service	5
3.	Communication Format.....	5
3.0	Instruction Set.....	6
3.1	Get Device MAC.....	6
3.2	Get Device Version.....	7
3.3	Get Device Battery	7
3.4	Get Device Time.....	7
3.5	Set Device Time	8
3.6	Exchange Random Number	8
3.7	Authentication.....	9
3.8	Get Device Name.....	9
3.9	Set Device Name	10
3.10	Get Device Config.....	错误!未定义书签。

3.11 Set Device Config 错误!未定义书签。

3.12 Shut Down Device 10

3.13 Read Temperature ADC 错误!未定义书签。

3.14 Statistic Offline Data 错误!未定义书签。

3.15 Read Offline Data 错误!未定义书签。

3.16 Clear Offline Data 错误!未定义书签。

4. Authentication 12

1. Broadcast Data

1.1 Broadcast Data Format

Broadcast data is a combination of multiple data, each of which has a unique identifier, the specific format is as follows:

Function	Byte	Description
packet length	1 Byte	packet identification length(1) + packet data length(N)
packet identification	1 Byte	used to distinguish different types of data
packet data	N Byte	

Bracelets use the following signs:

Identification	Ble Standard	Description
0x01	broadcast type	packet identification length(1) + packet data length(N)
0xFF	manufacturer custom data	user broadcast data
0x09	device name	device name
0x02	service UUID	service UUID

1.2 Broadcast Data Detail Introduction

1.2.1 Broadcast Type

The broadcast type is 0x06, indicating that this device is a BLE device.

Current packet length	Identification	Data
0x02	0x01	0x06

1.2.2 Manufacturer Custom Data

The manufacturer custom data is user broadcast data.

Current packet length	Identification	Data
0x08	0xFF	0x22 + 0x07 + 5 byte variable data

1.2.3 Device Name

The default device name is eBracelet.

Current packet length	Identification	Data
0x0A	0x09	eBracelet

1.2.4 Service UUID

The default UUID is 0xFF00.

Current packet length	Identification	Data
0x03	0x02	0x00 0xFF

2. Service Custom&UUID

The bracelet includes the following services:

Service	UUID	Description
Generic Access	0x1800	Bluetooth specification for indicating device name information
Generic Attribute	0x1801	Bluetooth specification
DIS device info	0x180A	Use to indicate manufacturer, model, firmware version, etc.
BAS battery info	0x180F	Used to indicate battery level
Data transmission	0xFF00	Used to send and receive data between the mobile APP and the bracelet
OTA	0x1911	Used to update the bracelet

2.1 Battery Info Service

The UUID of Battery Info Service is 0x180F, contains a service characteristic value:

Characteristic	Battery Level
uuid	0x2A19
Access	Read & Notify
Description	The battery rating is used to indicate the use of the device's battery, the level range is 0%~100%, 0% indicates that the battery is fully charged and 100% indicates that the battery is fully charged.

2.2 Data Transmission Service

The UUID of Data Transmission Service is 0xFF00, contains 4 service characteristic values:

Characteristic	SPP-DATA-ModuleToPhone
uuid	0xFF01
Access	Notify
Description	For sending data to the mobile APP ; Module → Phone The phone APP needs to enable this feature value, then the module can send data to the phone APP

Characteristic	SPP-DATA-PhoneToModule
uuid	0xFF02
Access	Write and Write without response
Description	For sending data to the module ; Phone → Module

3. Communication Format

Data frame is the basic unit of data transmission. Each frame is composed of command code, data length, data field and check code. Each field is composed of several bytes. The specific frame format is as follows:

Field	Code	Bytes	Description
Code	CMD	1	See the following description
Packet length	DLC	1	The byte length of packet data
Packet	DATA	N	Complete the function with command code, variable

BT01XW Bluetooth Communication Protocol

data			length, see the following description
Check code	SUM	1	The sum of all the bytes above

3.0 Instruction Set

Function	Code	Length	Data	Description
get device MAC	0x01	01	01	
get device version	0x02	01	01	
get device battery	0x03	01	01	
get device time	0x04	01	01	
set device time	0x05	04	unix timestamp	
exchange random number	0x06	17	random number	Ssed for identity authentication, random is 8 bytes, the remaining 9 bytes of data are reserved
authentication	0x07	17	authentication data	Authentication data is 16 bytes, the remaining 1 byte of data is reserved
get device name	0x08	01	01	
set device name	0x09	variable	device name	The maximum length of the device name is 17 bytes, using ascii encoding
shut down device	0x0C	01	01	
lighting device	0x14	02	duration	seconds
beeping device	0x15	02	duration	seconds

3.1 Get Device MAC

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x01
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x81
DLC	1	= 0x06
DATA	6	Device MAC is 6 bytes, high byte is first

SUM	1	Check code, the sum of all the bytes above.
-----	---	---

3.2 Get Device Version

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x02
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x82
DLC	1	N(Variable)
DATA	N	Version, ASCII encoding
SUM	1	Check code, the sum of all the bytes above.

3.3 Get Device Battery

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x03
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x83
DLC	1	= 0x03
DATA0	1	Battery voltage high byte, hex
DATA1	1	Battery voltage low byte, hex
DATA2	1	Battery percent, range from 1-100(estimated)
SUM	1	Check code, the sum of all the bytes above.

Note: Battery is 2 hex, unit is mV,

Example: DATA0 = 0x10, DATA1 = 0x23, battery voltage =0x1023 dec value is 4131mV

3.4 Get Device Time

Phone → Module

Code	Byte Count	Description
------	------------	-------------

BT01XW Bluetooth Communication Protocol

CMD	1	= 0x04
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x84
DLC	1	= 0x04
DATA	4	unix timestamp, high byte is first, 4 bytes
SUM	1	Check code, the sum of all the bytes above.

3.5 Set Device Time

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x05
DLC	1	= 0x04
DATA	4	unix timestamp, high byte is first, 4 bytes
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x85
DLC	1	= 0x01
DATA	1	= 0x00 Success = 0x01 Fail
SUM	1	Check code, the sum of all the bytes above.

3.6 Exchange Random Number

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x06
DLC	1	= 0x11
DATA0	8	Random random_m, 8 bytes
DATA1	9	Reserved
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x86

BT01XW Bluetooth Communication Protocol

DLC	1	= 0x11
DATA0	8	Random random_s, 8 bytes
DATA1	9	Reserved
SUM	1	Check code, the sum of all the bytes above.

3.7 Authentication

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x07
DLC	1	= 0x11
DATA0	16	Authentication data, AuthData
DATA1	1	Reserved
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x87
DLC	1	= 0x11
DATA0	16	Reserved
DATA1	1	= 0x00 Authentication succeed = 0x01 Authentication failed, not exchange random = 0x02 Authentication failed, authentication data is error = 0x03 Authentication failed, the number of errors has exceeded the limit. Please reconnect = 0x04 Authentication is succeed already, no need to authenticate again
SUM	1	Check code, the sum of all the bytes above.

3.8 Get Device Name

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x08
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x88

BT01XW Bluetooth Communication Protocol

DLC	1	N(Variable)
DATA	N	Device name, ADCII encoding, the maximum length is 17 bytes
SUM	1	Check code, the sum of all the bytes above.

3.9 Set Device Name

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x09
DLC	1	N(Variable)
DATA	N	Device name, ADCII encoding, the maximum length is 17 bytes
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x89
DLC	1	= 0x01
DATA	1	= 0x00 Success = 0x01 Fail
SUM	1	Check code, the sum of all the bytes above.

3.10 Shut Down Device

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x0C
DLC	1	= 0x01
DATA	1	= 0x01
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x8C
DLC	1	= 0x01
DATA	1	= 0x01 Power down device in one second
SUM	1	Check code, the sum of all the bytes above.

3.11 Open Device Light

Phone → Module

BT01XW Bluetooth Communication Protocol

Code	Byte Count	Description
CMD	1	= 0x14
DLC	1	= 0x02
DATA0	1	= 0x00 reserved
DATA1	1	lighting time, unit: second range of value: 1-250, default value is 5
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x94
DLC	1	= 0x01
DATA	1	= 0x00 Success = 0x01 Fail
SUM	1	Check code, the sum of all the bytes above.

3.12 Open Device Beep

Phone → Module

Code	Byte Count	Description
CMD	1	= 0x15
DLC	1	= 0x02
DATA0	1	= 0x00 reserved
DATA1	1	beeping time, unit: second range of value: 1-250, default value is 5
SUM	1	Check code, the sum of all the bytes above.

Module → Phone

Code	Byte Count	Description
CMD	1	= 0x95
DLC	1	= 0x01
DATA	1	= 0x00 Success = 0x01 Fail
SUM	1	Check code, the sum of all the bytes above.

4. Authentication

In order to prevent device parameters from being modified illegally, it is necessary to authenticate before modifying device parameters. Only after successful authentication, the following operation can be done: Set device time, Set devcie name, Shut down device.

Identity authentication uses AES encryption(ECB mode, PKCS7Padding), first the APP and the bracelet exchange random numbers (random_m and random_s), then the APP merges the 2 random numbers, and perform AES calculation to obtain identity authentication data: AuthData=AES (devicekey, random_m + random_s) , the devicekey is the preset key(Please contact the manufacturer to get the devicekey). The same calculation is done on the device side, if the authentication data is consistent, the authentication is successful.

Identity authentication process:

