

## 佩林 Buffalo 串口通讯协议

苏州佩林网络科技有限公司

日期	版本	释放	标注
2015.9.28	V1.0	Kevin	First created
2016.2.29	V1.1	Kevin	Add CMD(0x0D)
2016.3.04	V1.11	Reggie	Modify Command
2016.4.29	V1.2	Reggie	Add CMD(0x0E)
2016.5.04	V1.3		Add CMD(0x0F)
2017.3.30	V1.4		Add CMD(0x10) Add Event(0x08) Add CMD(0x11) Add CMD(0xF3)
2018.10.31	V1.5		Add CMD(0x12)
2018.11.15	V1.6 SW:1.3.11		Modify CMD(0x12) Add CMD(0x13) Add CMD(0x14) Add Event(0x09) Add Event(0x0A)

## Table of Contents

1. Serial Interface(串口设置).....	3 -
1.1. 串口设定.....	3 -
1.2. 数据包格式.....	3 -
2. Protocol (串口协议).....	4 -
2.1. Command [Type = 0x01](命令).....	4 -
2.1.1.OpCode(操作码).....	4 -
1) Set Pairing Mode [0x01] (进入配对模式).....	4 -
2) Set Pairing Mode Cancel [0x02] (取消配对模式).....	4 -
3) Set Disconnected [0x03] (设置模块断开连接).....	5 -
4) Get Local Device Name [0x04] (获取设备名称).....	5 -
5) Get Local BD Address [0x05] (获取 Buffalo 的蓝牙地址).....	5 -
6) Get Firmware Version [0x06] (获取 Buffalo 当前的固件版本号).....	6 -
7) Send Data [0x0B] (发送数据).....	6 -
8) Set Local Name [0x0C] (设置蓝牙本地名称).....	6 -
9) Get System State [0x0D] (获取系统状态).....	7 -
10) SET GPIO [0x0E] (设置模块 GPIO 状态).....	7 -
11) SET_Scan [0x10](设置扫描).....	7 -
12) SET Baudrate [0x0F] (设置串口波特率).....	8 -
13) SET_Deepsleep[0x11](设置睡眠唤醒模式).....	8 -
14)SET_RF_TX_Power[0xF3](设置 RF 发射功率).....	9 -
15) SET_LPM_WAKEUP_IO[0x12](蓝牙唤醒后, 设置唤醒主机的引脚和模式).....	9 -
16) SET_LPM_ADV[0x13]^ (设置蓝牙心跳模式睡眠间隔).....	9 -
17) SET_LPM_ADV[0x14]^ (设置蓝牙心跳模式连接间隔).....	10 -
2. Reserved [Type = 0x02](预留).....	11 -
2.3. Response [Type = 0x03](回应).....	11 -
2.3.1. Response Error Code(错误代码).....	11 -
2.4. Event [Type = 0x04](事件).....	12 -
2.4.1. OpCode.....	12 -
1) BLE Startup [0x01] (Buffalo 初始化完成).....	12 -
2) Pairing Status [0x02] (Buffalo 的配对信息).....	12 -
3) Local Device Name [0x03] (Buffalo 上报设备名称).....	12 -
4) Local BD Address [0x04] (Buffalo 上报蓝牙地址).....	12 -
5) Local Firmware Version [0x05] (Buffalo 上报固件版本号).....	13 -
6) Master send data package [0x06] (Buffalo 上报数据).....	13 -
7) Baudrate Return [0x07] (Buffalo 上报串口波特率).....	13 -
8) Event_ADV_Report[0x08](Buffalo 上报扫描到的信息).....	13 -
9) .EVENT_LPM_ADV_TIMEOUT [0x09](Buffal 上报心跳模式广播间隔超时信息).....	13 -
10) EVENT_LPM_CON_TIMEOUT [0x0A](Buffal 上报心跳模式连接间隔超时信息).....	13 -
3.Message Sequence Example (举例说明命令顺序).....	14 -

## 1. Serial Interface(串口设置)

通过串口发送数据包来管理 BLE 连接。

### 1.1. 串口设定

NO	NAME	CONFIG
1	Baud Rate	9600
2	Data Bit	8Bit
3	Stop	1Bit
4	Parity Bit	None

### 1.2. 数据包格式

Command	Host send to Buffalo, to manage the BLE connections
Reserved	N/A
Response	Buffalo send to Host, to response the command packet from Host
Event	Buffalo send to Host, to report BLE connection event to Host

Command(命令):MCU 发送命令给 Buffalo, 管理蓝牙 BLE 的连接。

Reserved(预留): N/A 未使用

Response(回应):Buffalo 发送给主机, MCU 发的每个命令都会有一个回应。

Event(事件): Buffalo 发送给主机, Event 是模块收到了 APP/APK 的数据或蓝牙有任何变化(如连接断开/连上)后, Buffalo 上报给 MCU 的。

Packet format 数据包格式

LSB(小端)

MSB(大端)

Header	Type	Length	OpCode	Payload	Checksum
--------	------	--------	--------	---------	----------

Payload 的内容是可变的,它取决于操作码 OpCode。

Table 1 <Packet Format>

<b>Header</b> 数据起始位	0x77
<b>Type</b> 数据格式	Type of Packet 数据格式 Command 0x01 Reserved 0x02 Response 0x03 Event 0x04
<b>Length</b> 数据长度	Length of OpCode + Payload (长度=操作码+数据负载)
<b>OpCode</b> 操作码	Selected Buffalo Function (选择需要的 Buffalo 功能)
<b>Payload</b> 数据负载	Detail data of each OpCode (功能数据包的详细内容)
<b>Checksum</b> 校验码	Check the validity Packet header (校验数据包有效性) Checksum = Header ^ Type ^ Length ^ OpCode ^ Payload

## 2. Protocol (串口协议)

### 2.1. Command [Type = 0x01](命令)

Command control BLE Command : 0x01

0x01 代表 MCU 发出的命令

\* Packet Format : Command

LSB(小端)

MSB(大端)

Header	Type	Length	OpCode	Payload	Checksum
0x77 (Fix)	0x01 (Fix)	Variable	Variable	Variable	Variable

\* Indication: Header, Type, Length, OpCode, Payload, Checksum in Sequence mark as H, T, L, O, P, C

#### 2.1.1.OpCode(操作码)

##### 1) Set Pairing Mode [0x01] (进入配对模式)

设置 Buffalo 打开广播

Command					
H	T	L	O	P	C
77	01	01	01	NA	76
Response					
H	T	L	O	P	C
77	03	02	01	Err(1 byte)	Checksum

##### 2) Set Pairing Mode Cancel [0x02] (取消配对模式)

设置 Buffalo 关闭广播

Command					
H	T	L	O	P	C
77	01	01	02	NA	75
Response					
H	T	L	O	P	C
77	03	02	02	Err(1 byte)	Checksum

**3) Set Disconnected [0x03] (设置模块断开连接)**

设置 Buffalo 断开选定 Handle 对应的手机的连接

Command					
H	T	L	O	P	C
77	01	03	03	Handle	Checksum
Response					
H	T	L	O	P	C
77	03	02	03	Err(1 byte)	Checksum

Handle: the connection handle, 2 Bytes.

Buffalo 可以同时连接 2 个手机，每个手机都有一个自己连接的 Handle.

连接成功后 Handle 的值，Buffalo 会通过 EVENT .2) Pairing Status [0x02] (Buffalo 的配对信息) 上报给 MCU.

**4) Get Local Device Name [0x04] (获取设备名称)**

获取 Buffalo 的设备名称

Command					
H	T	L	O	P	C
77	01	01	04	NA	73
Response					
H	T	L	O	P	C
77	03	02	04	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	Length	03	Payload	Checksum

**5) Get Local BD Address [0x05] (获取 Buffalo 的蓝牙地址)**

Buffalo 会以 ASCII 码的格式上报蓝牙地址

Command					
H	T	L	O	P	C
77	01	01	05	NA	72
Response					
H	T	L	O	P	C
77	03	02	05	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	Length	04	Payload	Checksum

举例说明：蓝牙地址是 00 18 31 84 A1 32

Command: 77 01 01 05 72

Response: 77 03 02 05 01 72

Event: 77 04 0D 04 30 30 31 38 33 31 38 34 41 31 33 32 Checksum

**6) Get Firmware Version [0x06] (获取 Buffalo 当前的固件版本号)**

获得 Buffalo 当前烧录的固件版本号

Command					
H	T	L	O	P	C
77	01	01	06	NA	71
Response					
H	T	L	O	P	C
77	03	02	06	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	Length	05	Payload	Checksum

举例说明: 当前版本为: S20150506(ASCII 53 32 30 31 35 30 35 30 36)

Command: 77 01 01 06 71

Response: 77 03 02 06 01 71

Event: 77 04 0A 05 53 32 30 31 35 30 35 30 36 Checksum

**7) Send Data [0x0B] (发送数据)**

发送数据给主机(APP/APK)

Command					
H	T	L	O	P	C
77	01	Length	0B	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	0B	Err(1 byte)	Checksum

Payload: handle of connection(2 bytes) + user data(ex: 0A CE 09 )

Payload: 2bytes 的 handle+用户数据(ex:0A CE 09)

解释:Buffalo 可以同时连接 2 个手机, 每个手机有一个不同的 Handle,发送数据时必须选定 2 个有效 Handle 中的一个。

Payload 数据负载最大长度为 22Bytes.

**8) Set Local Name [0x0C] (设置蓝牙本地名称)**

用户设置完本地名称后, Buffalo 将会以这个名称开启广播, APP/APK 可以收搜到所设置名称的蓝牙设备。

这个指令必须在 Buffalo 复位或者刚上电初始化的阶段, 必须在“Set Pairing Mode”指令之前完成。

Command					
H	T	L	O	P	C
77	01	Length	0C	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	0C	Err(1 byte)	Checksum

Payload:以 ASCII 码格式的本地名称。

Payload 数据负载最大长度为 16Bytes.

**9) Get System State [0x0D] (获取系统状态)**

Buffalo 收到此命令后，MCU 会收到 Buffalo 包含当前状态的 Response。

Command					
H	T	L	O	P	C
77	01	01	0D	NA	7A
Response					
H	T	L	O	P	C
77	03	06	0D	Payload	Checksum

Payload:广播状态(0 关; 1 开)+handle1+ handle2

解释:Handle1 代表手机 1, Handle2 代表手机 2

**10) SET GPIO [0x0E] (设置模块 GPIO 状态)**

MCU 通过发送这条指令来实现对 Buffalo GPIO 的控制(输出高或低)。

Command					
H	T	L	O	P	C
77	01	Length	0E	Payload <sup>(1)</sup>	Checksum
Response					
H	T	L	O	P	C
77	03	02	0E	Err(1 byte)	Checksum

Payload: Level(bit7)| GPIO\_Number(bit0~6),

Bit7 为电平控制(1 高电平, 0 低电平), Bit0~6 为 GPIO 的选择。

Payload 的最大长度为 8Bytes. 使用者一次可以同步最多控制 8 个 GPIO.

Bit0~6	0000	0000	0000	0000	0000	0000	0000	0000	0001	0001	0001	0001
	000	001	010	011	100	101	110	111	000	001	010	011
GPIO	DIO0	DIO1	DIO2	DIO3	DIO4	DIO5	DIO6	DIO7	DIO8	DIO9	DIO10	DIO11

举例说明 CMD :0x 77 01 04 0E 00 81 82 7F

00 P0 output L

81 P1 output H

82 P2 output H

**11) SET\_Scan [0x10](设置扫描)**

MCU 通过发送这条指令来实现对 Buffalo SCAN 的管理。

Command					
H	T	L	O	P	C
77	01	02	10	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	10	Err(1 byte)	Checksum

Payload: UINT 8 SET\_Scan/ 00 ---- NO\_SCAN ,01---- LOW\_SCAN ,02---HIGH\_SCAN

描述:打开 HIGH\_SCAN 30 秒后自动结束并进入 LOW-SCAN.

打开 LOW-SCAN 300 秒后自动结束并进入 NO\_SCAN.

举例说明:CMD :0x 77 01 02 10 00 64, 设置 Buffalo 进入 No\_SCAN 状态.

**12) SET Baudrate [0x0F] (设置串口波特率)**

MCU 通过发送这条指令来设置 Buffalo 的串口波特率。

Command					
H	T	L	O	P	C
77	01	05	0F	Payload <sup>(1)</sup>	Checksum
Response					
H	T	L	O	P	C
77	03	02	0F	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	05	07	Payload	Checksum

Payload: UINT 32 Baudrate 9600~1500000

举例说明:设置 Buffalo 的波特率为 115200. (0x 00 01 C2 00)

CMD:0x 77 01 05 0F 00 C2 01 00 BF

MCU 发送这条指令给 Buffalo 之后, Buffalo 会用未设置之前的串口波特率(默认 9600)给 MCU Response 一个值(0x 77 03 02 0F 01 78), 接着在 100ms 后用新设定的串口波特率 115200 给 MCU 再 Response 一个值(0x 77 04 05 07 00 C2 01 00 B2)。

Note:

(1): Note: Default baudrate of Buffalo is 9600

Buffalo 默认串口波特率为 9600bps

**13) SET\_Deepsleep[0x11](设置睡眠唤醒模式)**

Buffalo 会在收到 MCU 发来的指令 100ms 后进入睡眠模式。

Command					
H	T	L	O	P	C
77	01	03	11	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	11	Err(1 byte)	Checksum

Payload(2 Bytes): UINT8 Wake\_up\_IO\_pin, UINT8 Wake\_up\_IO\_level

Wake\_up\_IO\_pin: 选择哪个 GPIO 来唤醒模块。

Bit0~7	00001001	00001010	00001011	00001100	00001101
GPIO	P9	P10	P11	P12 <sup>(a)</sup>	P13 <sup>(a)</sup>

Wake\_up\_IO\_level:

00: 给 GPIO 一个低电平来唤醒模块。

01: 给 GPIO 一个高电平来唤醒模块。

(a): P12\ P13 是 OD 结构的 GPIO, 内部不能自己上拉。

举例说明:CMD :0x 77 01 03 11 0B 00 6F // 0B 00 :GPIO11 低电平可以唤醒 Buffalo.

Buffalo 收到这条指令 100ms 后进入睡眠模式, MCU 可以通过给 Buffalo 的 GPIO11 一个低电平来唤醒 Buffalo.



**14) SET\_RF\_TX\_Power[0xF3](设置 RF 发射功率)**

MCU 可以通过发送这条指令来设置 Buffalo 的 RF 发射功率。

Command					
H	T	L	O	P	C
77	01	02	F3	Payload	Checksum
Response					
H	T	L	O	P	C
77	01	02	F3	Err(1 byte)	Checksum

Payload(1 Byte): UINT8:RF\_TX\_Power

Bit0~7	00000000	00000001	00000010	00000011	00000100	00000101	00000110	00000111
GPIO	-14dBm	-11 dBm	-8 dBm	-5 dBm	-2 dBm	+2dBm	+4 dBm	+8dBm

举例说明:CMD :0x 77 01 02 F3 07 80.

设置 Buffalo 的发射功率为 8dBm.

**15) SET\_LPM\_WAKEUP\_IO[0x12](蓝牙唤醒后, 设置唤醒主机的引脚和模式)**

Command					
H	T	L	O	P	C
77	01	03	12	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	12	Err(1 byte)	Checksum

Payload(2 Bytes):

UINT8 Wake\_up\_MCU\_IO, 设定唤醒主机的引脚 IO0~IO14

(DIO7,DIO8,DIO11,DIO12,DIO13 不建议配置)

UNIT8 Wake\_up\_MCU\_duration, 设定唤醒主机的引脚高电平脉冲时间,

持续时间: Wake\_up\_duration\*10ms

**16) SET\_LPM\_ADV[0x13]^ (设置蓝牙心跳模式睡眠间隔)**

Command					
H	T	L	O	P	C
77	01	05	13	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	13	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	01	09	N	7B

Payload(4 Bytes):

UINT16 Interval,心跳模式广播间隔  $0.625 \times \text{Interval ms}$  间隔越大, 功耗越低

UINT16 Timeout,心跳模式广播持续时间,单位秒, 超时会通过 CMD[0x12] 设定唤醒 MCU, 并发送事件 EVENT\_LPM\_ADV\_TIMEOUT[0x09]

**17) SET\_LPM\_ADV[0x14]^ (设置蓝牙心跳模式连接间隔)**

Command					
H	T	L	O	P	C
77	01	05	14	Payload	Checksum
Response					
H	T	L	O	P	C
77	03	02	14	Err(1 byte)	Checksum
Event					
H	T	L	O	P	C
77	04	01	0A	N	78

Payload(4 Bytes):

UINT16 Interval,心跳模式连接间隔  $1.25 * \text{Interval ms}$  间隔越大,功耗越低

UINT16 Timeout,心跳模式连接广播持续时间,单位秒,超时会通过 CMD[0x12] 设定唤醒 MCU,并发送事件 EVENT\_LPM\_CON\_TIMEOUT[0x0A],

A: Interval : 0x0020~0x4000, interval \*0.625 ms, 间隔越大, 功耗越低, 但是越难被连接, 客户自己衡量, 建议 1s.

## 2. Reserved [Type = 0x02](预留)

Not in use

## 2.3. Response [Type = 0x03](回应)

Response 是 Buffalo 通知 MCU 已经成功收来自 MCU 的指令。

MCU 发送完命令后可以查看 Buffalo 的 Response 来确定 Buffalo 是否在执行。

\* Packet Format : Response

LSB				MSB	
Header	Type	Length	OpCode	Payload	Checksum
0x77 (Fix)	0x03 (Fix)	Variable	Variable	Variable	Variable

### 2.3.1. Response Error Code(错误代码)

Error	Remark
0x01	ERR_NONE //Command received and processing(没有错误)
0x02	ERR_LENGTH_FAIL // Length of command is error(命令长度错误)
0x03	ERR_INVALID_FAIL // Command invalid error(命令无效)

## 2.4. Event [Type = 0x04](事件)

Event 是模块收到了 APP/APK 的数据或蓝牙有任何变化(如连接断开/连上)后, 模块发给 MCU 的。

Event 数据包的 Opcode 对应了 Command 所指定的功能。

\* Packet Format : Event

LSB					MSB
Header	Type	Length	OpCode	Payload	Checksum
0x77 (Fix)	0x04 (Fix)	Variable	Variable	Variable	Variable

### 2.4.1. OpCode

#### 1) BLE Startup [0x01] (Buffalo 初始化完成)

Event					
H	T	L	O	P	C
77	04	02	01	Payload	Checksum

Payload: 0x01 初始化完成; 0x02 初始化失败.

#### 2) Pairing Status [0x02] (Buffalo 的配对信息)

主机连接或者断开 Buffalo.上报这个 Event 给 MCU.

Event					
H	T	L	O	P	C
77	04	Length	02	Payload	Checksum

Payload: Connected: 0x01 + connection Handle + master BD\_ADDR

连接:0x01+所连接手机的 Handle+手机的蓝牙地址

Disconnected: 0x02 + connection handle

断开:0x02+所连接手机的 Handle

#### 3) Local Device Name [0x03] (Buffalo 上报设备名称)

上报 Buffalo 的设备名称给 MCU.

Event					
H	T	L	O	P	C
77	04	Length	03	Payload	Checksum

Payload: Buffalo 设备名称以 ASCII 码形式给出。

#### 4) Local BD Address [0x04] (Buffalo 上报蓝牙地址)

上报 Buffalo 的蓝牙地址给 MCU.

Event					
H	T	L	O	P	C
77	04	Length	04	Payload	Checksum

Payload: Buffalo 蓝牙地址以 ASCII 码形式给出。

**5) Local Firmware Version [0x05] (Buffalo 上报固件版本号)**

上报 Buffalo 的固件版本号给 MCU.

Event					
H	T	L	O	P	C
77	04	Length	05	Payload	Checksum

Payload: Buffalo 固件版本号以 ASCII 码形式给出。

**6) Master send data package [0x06] (Buffalo 上报数据)**

Buffalo 把接收到的数据上报给 MCU.

Event					
H	T	L	O	P	C
77	04	Length	06	Payload	Checksum

Payload: connection handle(2 bytes) + data from BLE master

所连接手机的 Handle+手机发过来的数据

**7) Baudrate Return [0x07] (Buffalo 上报串口波特率)**

上报 Buffalo 的串口波特率给 MCU.

Event					
H	T	L	O	P	C
77	04	05	07	Payload	Checksum

Payload: UINT 32 Baudrate 9600~1500000. The Baudrate which set by MCU.

E.g. SET Buffalo Baudrate to 115200. (0x 00 01 C2 00)

**8) Event\_ADV\_Report[0x08](Buffalo 上报扫描到的信息)**

Event					
H	T	L	O	P	C
77	04	Length	08	Payload	Checksum

Payload: UINT8 Event type

INT8 Rssi

UINT8 Address[6]

UINT8 Data[n]

**9) .EVENT\_LPM\_ADV\_TIMEOUT [0x09](Buffal 上报心跳模式广播间隔超时信息)**

Event					
H	T	L	O	P	C
77	04	01	09	N	7B

Payload: NO

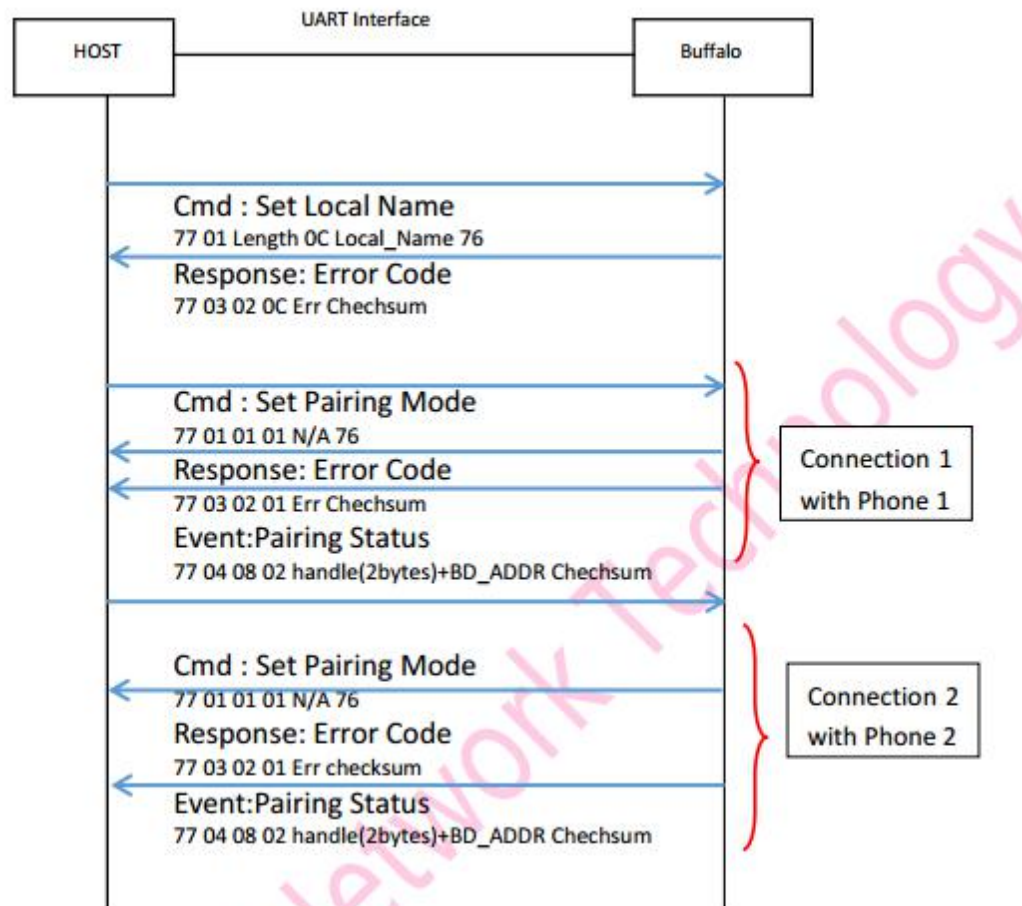
**10) EVENT\_LPM\_CON\_TIMEOUT [0x0A](Buffal 上报心跳模式连接间隔超时信息)**

Event					
H	T	L	O	P	C
77	04	01	0A	N	78

Payload: NO

### 3.Message Sequence Example (举例说明命令顺序)

设置进入配对模式命令 0x01 去被 Phone1 和 Phone2 连接



注意:Buffalo 作为蓝牙 Peripheral 工作，可以同时连接 2 个作为 Central 的手机。

对于连接 1:MCU 设置 Buffalo 进入配对模式，Buffalo 将会上报一个包含了 Phone1 蓝牙地址的 2-bytes 的 handle1 给 MCU 当 Phone1 连接上 Buffalo 后。

对于连接 2:MCU 再次设置 Buffalo 进入配对模式，Buffalo 将会再次上报一个包含了 Phone2 蓝牙地址的 2-bytes 的 handle2 给 MCU 当 Phone2 连接上 Buffalo 后。

Handle1 和 Handle2 是两个不同的值。