

类别	内容
关键词	BLE 透传 指令
摘要	透传模块指令格式及使用介绍

修订历史

版本	日期	原因
V0.9.00	2015/06/11	创建文档
V1.0.00	2015/10/11	初始版本
V1.0.01	2016/05/16	添加蓝牙的发送和接收的 UUID; 添加重要参数的获取指令; 添加主动断开连接的指令; 添加获取当前连接状态的指令; 添加修改和获取配对码的指令
V1.0.02	2019/03/08	公司名称变更, 更新文档模板
V1.0.03	2020/01/08	替换文档模板
V1.0.04	2020/12/17	更新文档模板

目 录

1. 产品简介.....	1
1.1 本文档适用范围.....	1
1.2 概述.....	1
1.3 主要特点.....	1
2. 引脚定义.....	2
3. 串口透传协议说明.....	4
3.1 串口透传协议说明.....	4
3.2 操作细节及要点.....	4
3.3 服务的 UUID.....	6
3.3.1 ServiceUUID.....	6
3.3.2 主机端发送 UUID.....	6
3.3.3 主机端接收 UUID.....	6
3.4 模块传输加密传输.....	6
4. 串口 AT 指令.....	8
4.1 AT 指令介绍.....	8
4.1.1 连接间隔设定.....	8
4.1.2 获取连接间隔时间.....	8
4.1.3 模块重命名.....	8
4.1.4 获取模块名称指令.....	9
4.1.5 波特率设定.....	9
4.1.6 获取串口波特率指令.....	9
4.1.7 获取物理地址 MAC.....	9
4.1.8 模块复位.....	9
4.1.9 广播周期设定.....	9
4.1.10 获取广播周期指令.....	10
4.1.11 发射功率设定.....	10
4.1.12 获取发射功率指令.....	10
4.1.13 数据延时设定.....	10
4.1.14 获取数据延时指令.....	11
4.1.15 模块低功耗指令.....	11
4.1.16 获取软件版本指令.....	11
4.1.17 获取当前模块的连接状态.....	11
4.1.18 断开模块连接指令.....	12
4.1.19 修改配对码指令.....	12
4.1.20 获取配对码的指令.....	12
4.1.21 设置传输加密指令.....	12
4.1.22 获取传输加密指令.....	13
4.2 AT 指令总表.....	13
5. 系统复位与恢复.....	16
5.1 模块复位.....	16
5.2 恢复出厂设置.....	16

6. 主机参考代码.....	17
7. 免责声明.....	18

1. 产品简介

1.1 本文档适用范围

适用于固件版本 V2.04 及以下的 ZLG9021P0-1(C)-TC 模块。可通过 AT 指令“TTM:VER-VERSION”（不含双引号）来获取固件版本号。

1.2 概述

ZLG9021P0-1(C)-TC 模块主要工作在透传模式下。

模块启动后会自动进行广播，已打开特定 APP 的手机会对其进行扫描和对接，连接成功之后就可以通过 BLE 在模块和手机之间进行数据传输。

透传模式下，用户 MCU 可以通过模块的通用串口和移动设备进行双向通讯；用户也可以通过特定的串口 AT 指令，对某些通讯参数进行修改（例如：串口波特率、广播周期等）。

移动设备可以通过 APP 对模块进行写操作，写入的数据将通过串口发送给用户的 MCU；同样的，模块收到来自用户 MCU 串口的数据包后，将自动转发给移动设备。

用户在使用此模块时，只需设计好 MCU 端和手机 APP 端代码即可。

由于 BLE 属于蓝牙 4.0 的范畴，因此要求手机的蓝牙模块是 4.0 以上的版本；iPhone4S 及以后的版本的苹果手机都支持；Android 手机系统版本必须 4.3.1 以上的版本才能够支持；微软的 Win8 以上的系统原生支持 BLE 的功能（有些电脑需要外加 BLE 适配器才能正常使用）。

1.3 主要特点

1. 操作简单，无需任何蓝牙协议栈应用经验；
2. 使用通用串口设计，半双工双向通讯，支持波特率最低 4800bps，最高 115200bps；
3. 默认 20ms 同步间隔，通讯迅速；
4. 支持 AT 指令软件复位模块、获取 MAC 地址；
5. 支持 AT 指令修改蓝牙连接间隔，用于控制不同的转发速率；
6. 支持 AT 指令调整发射功率，更改信号强度；
7. 支持 AT 指令修改广播间隔、设定数据延时(用户 MCU 串口接收准备时间)、修改串口波特率以及修改模块名的操作，这些参数均能掉电保存；
8. 串口数据包长度为 20byte 以下(含 20byte)的任意长度，大包自动分包；
9. 高速透传转发速率高达 5.5K/s 左右；
10. 支持 RESTORE 引脚接地 10s 恢复出厂设置；
11. 可通过串口获取蓝牙连接状态（连接，断线）字串提示；也可以通过 IO 的电平状态获取蓝牙的连接状态；
12. 可通过 I/O 口的电平状态获取模块的低功耗模式；
13. 支持 EN 引脚接地使能模块的功能；
14. 支持 AT 指令或者 EN 引脚控制进入极低功耗的待机模式；
15. BLE 数据发送加密。

2. 引脚定义

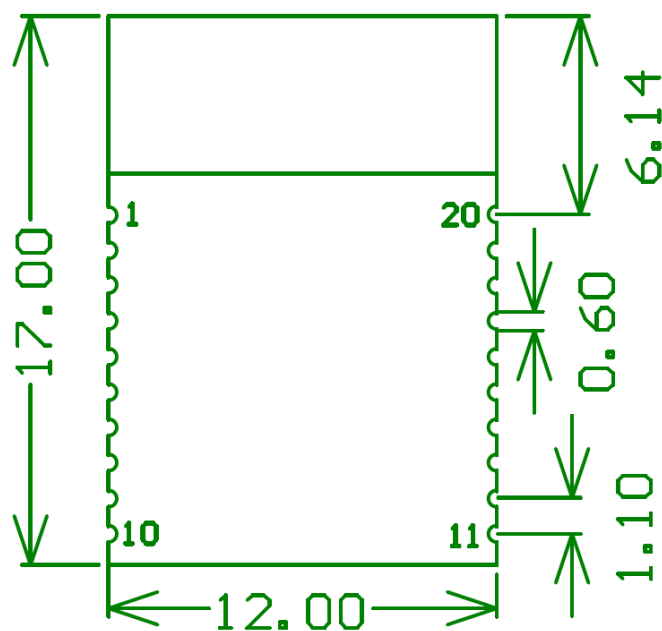


图 2.1 引脚图

表 2.1 引脚说明

引脚	定义	描述
1	NC	普通 IO 口，用户可不连接
2	NC	普通 IO 口，用户可不连接
3	BLKSLP	睡眠指示状态。当模块禁用蓝牙功能，进入极低功耗模式的时候，此引脚输出高电平，其余状态输出 1s 周期的方波
4	BLKCON	蓝牙连接状态。当模块与其他模块或手机连接时，此引脚输出 1s 周期的方波，其余状态输出高电平
5	EN	模块使能引脚， 高阻态输入，使用时不能悬空此引脚 0：使能模块蓝牙功能； 1：模块禁用，并进入极低功耗模式
6	TX	串口 TX 引脚
7	RX	串口 RX 引脚
8	VCC	电源引脚， 需与 VDD 一起接入 3.3V
9	VDD	电源引脚， 需与 VCC 一起接入 3.3V
10	GND	电源地引脚
11	GND	电源地引脚

续上表

引脚	定义	描述
12	BRTS	数据发送请求引脚， 高阻态输入，使用时不能悬空此引脚 0: 模块不再睡眠，串口允许接收数据或命令； 1: 串口不接收数据或命令，且模块在空闲的状态下会进入低功耗模式
13	BCTS	数据输入信号，表示模块有数据将要发送到 MCU 0: 当前模块有数据，提示 MCU 准备接收； 1: 当前模块没有数据
14	NC	普通 IO 口，用户可不连接
15	RESTORE	恢复出厂设置引脚。 当复位模块的 30s 内拉低此引脚，且持续 10s 之后即可恢复出厂设置，此引脚悬空时默认为高电平
16	NC	普通 IO 口，用户可不连接
17	NC	普通 IO 口，用户可不连接
18	NC	普通 IO 口，用户可不连接
19	NC	普通 IO 口，用户可不连接
20	RESET	低电平复位引脚

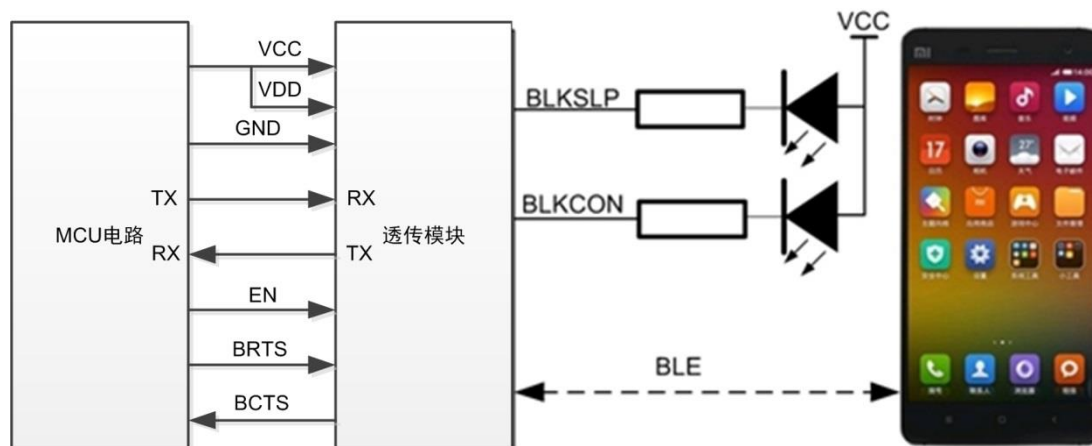


图 2.2 模块透传模式示意图

如图 2.2 所示，其中引脚 BCTS 用于提醒 MCU 当前模块是否有数据，此功能主要用于唤醒 MCU。如果 MCU 无需进入低功耗，或者 MCU 的串口可以唤醒 MCU，可不使用此引脚。

为了降低模块的功耗，此模块中 EN 与 BRTS 引脚被设置为高阻态输入，因此在使用模块的时候必须把这两个引脚接入固定的电平，以免状态错乱。

为避免用户 MCU 的 I/O 和模块 I/O 的输出电平差异导致大电流，建议在模块的输出信号线 TX、BCTS 上串入一小额定流电阻。

3. 串口透传协议说明

3.1 串口透传协议说明

模块的透传模式是指，通过通用串口和用户 MCU 相连，建立用户 MCU 和移动设备之间的双向通讯。用户可以通过串口，使用指定的 AT 指令对串口波特率，BLE 连接间隔等内容进行设置(详见后面第 4 章节)。针对不同的串口波特率以及 BLE 连接间隔，以及不同的发包间隔，模块将会有不同的数据吞吐能力。为协调低速 MCU 的使用，默认波特率为 9600bps；当有大数据量传输，或者高实时性需求的应用中，建议设定为高速串口波特率 115200bps，设置的波特率支持掉电保存；如果有大数据量传输且保证数据传输稳定，则建议使用 38400bps，且在应用层加重传协议进行数据传输。

模块 BLE 连接间隔为 20ms，串口波特率为 115200bps 时，模块具有最高理论转发能力(5.5kbyte/s)。

为了保证数据传输的稳定性，模块可以从串口一次性传输的数据量限定在 200 字节以内，模块会根据数据包大小自动分包发送，每个无线包最大载荷为 20 个字节。移动设备端发往模块的数据包，必须自行分包(1~20 字节/包)发送。模块收到无线包后，会依次转发到主机串口接收端。

3.2 操作细节及要点

1.串口硬件协议：数据位 8 位、无校验位、1 位停止位，串口的波特率可用 AT 指令来修改，见第 4 章节；

2.EN 引脚为高电平，蓝牙模块处于完全睡眠状态(通过 AT 指令让模块进入睡眠状态的模式除外)；EN 置低时，模块会以 200ms 的间隔开始广播，直到与手机对接成功。当 EN 引脚从低电平到高电平跳变，不论模块处于何种状态状态，都会立即进入睡眠模式；

3.连接成功之后，主机(MCU)如有数据发送至 BLE 模块，需将 BRTS 拉低，主机可在约 100us 后开始发送数据。发送完毕之后主机应主动抬高 BRTS，让模块退出串口接收模式；

4.当模块有数据上传请求时，模块会置低 BCTS，最快会在 500us 之后开始发送，直到数据发送完毕。这个延时可以通过 AT 指令进行配置，见第 4 章节。数据发送完毕，模块会将 BCTS 置高；

5.如若主机的 BRTS 一直保持低电平，则蓝牙模块会一直处于串口接收模式，会有较高的功耗(通过 AT 指令让模块进入睡眠模式的状态除外)；

6.在模块连接成功后，会从 TX 给出“TTM:CONNECTED\r\n0”字串，可以根据此字串来确定是否可以正常转发操作。当然也可以使用连接状态提示脚确认是否已经连接。当连接被断开后，会从 TX 发出“TTM:DISCONNCE\r\n0”字串提示，此时指示连接状态的 IO 口输出高电平；

7.模块的蓝牙默认连接间隔为 20ms，如果需要节省功耗采用低速转发模式，需通过 AT 指令调整连接间隔(最长连接间隔 2000ms)，必须重新连接手机后才有效。每个连接间隔最多传输 110 个字节，连接间隔为 T(单位:ms)，那么每秒最高转发速率 V(单位 byte/s)为：

$$V = 110 \times 1000 / T$$

如果模块的蓝牙连接间隔为 20ms，而每个间隔最多传输 110byte，因此理论最高传输能力(转

发速率)为 $110 \times 50 = 5.5 \text{Kbyte/s}$ 。测试表明,转发速率在 4K/s 以下,漏包机率很低。但是为了安全起见,无论是低速或者高速转发大数据量应用时,都建议在上层做校验重传处理,这里推荐使用 Xmodem 传输协议。

表 3.1 为不同条件下的传输速率。

表 3.1 不同条件下的传输速率

通讯参考模式	BLE 连接间隔 T(ms)	理论最高转发能力 V(byte/s) $V = 110 \times 1000 / T$	串口包长度 L(byte)	串口发包间隔 TS(ms) 当 $L \leq 100$ 时, $TS \geq T$ 当 $100 < L \leq 200$ 时, $TS \geq T \times 2$	实际转发速率 V0(byte/s) $V0 = L \times 1000 / TS$	备注
1	20	5.5K	100	$TS \geq T$ 即可,若取 $TS = 20 \text{ms}$	$100 \times 1000 / 20 = 5 \text{K}$	TS 偏小,不推荐
2	20	5.5K	200	$TS \geq T \times 2$ 即可,若取 $TS = 50 \text{ms}$	$200 \times 1000 / 50 = 4 \text{K}$	
3	20	5.5K	200	$TS \geq T \times 2$ 即可,若取 $TS = 60 \text{ms}$	$200 \times 1000 / 60 = 3.3 \text{K}$	
4	20	5.5K	80	$TS \geq T$ 即可,若取 $TS = 35 \text{ms}$	$80 \times 1000 / 30 = 2.6 \text{K}$	
5	20	5.5K	70	$TS \geq T$ 即可,若取 $TS = 30 \text{ms}$	$70 \times 1000 / 30 = 2.3 \text{K}$	
6	20	5.5K	60	$TS \geq T$ 即可,若取 $TS = 30 \text{ms}$	$60 \times 1000 / 30 = 2 \text{K}$	
7	20	5.5K	40	$TS \geq T$ 即可,若取 $TS = 30 \text{ms}$	$40 \times 1000 / 30 = 1.3 \text{K}$	
8	20	5.5K	20	$TS \geq T$ 即可,若取 $TS = 30 \text{ms}$	$20 \times 1000 / 30 = 666 \text{byte}$	

用户可以根据实际应用设计特定的通讯模式,串口包的长度可以设计在 $80 \text{byte} < L < 200 \text{byte}$ 之间(大包传输),根据 BLE 协议有以下关系:

当取 $L < 100$ 时, $TS \geq T$;

当取 $100 < L < 200$ 时, $TS \geq T \times 2$;

满足以上条件的转发模式都是相对安全的，若取 $TS=T$ ， $TS=T \times 2$ ，可以用但不推荐，丢包率相对较高，必须加入校验重发机制。也就是说，当串口包采用 $100\text{byte} < L < 200\text{byte}$ 的大包时，串口数据可以一次性传递给模块，但需要预留模块通过蓝牙信号发送数据的时间，否则会出现追尾现象。如：在连接间隔设置为 $T=20\text{ms}$ 时，如串口数据包长度选择 $L=200$ ，则 TS 必须大于 $T \times 2=40\text{ms}$ ，取 $TS=50\text{ms}$ 是比较合理的选择。

3.3 服务的 UUID

3.3.1 ServiceUUID

```
0x0000FEE900001000800000805F9B34FB
```

3.3.2 主机端发送 UUID

```
0xD44BC439ABFD45A2B575925416129600
```

3.3.3 主机端接收 UUID

主机接收的特征值是 Notify 模式。

```
0xD44BC439ABFD45A2B575925416129601
```

3.4 模块传输加密传输

此模块传输数据时默认是不加密的，可以通过 AT 指令来开启传输加密功能。当模块开启传输加密功能之后，就可以保证传输的数据不被第三方工具破解。

如果某个手机在第一次连接模块的时候，会提示传输加密的对话框，只需点击“配对”即可，不需要输入配对码。如果点击取消配对，则模块立马断开与手机的连接；如果手机忽略了配对信息，即没有点击“配对”也没有点击“取消”，则 30s 后自动断开与手机的连接。在手机确认加密之前，模块与手机之间传输功能会被禁止。



图 3.1 小米手机中提示配对加密请求

如果开启了传输配对功能（详见 4.1.19 修改配对码指令 章节），则需要控制 APP 在确认加密之后在发送配对码。

当手机第一次与模块加密确认之后，后续再连接模块就不需要重新确认。

如果模块开启了传输加密功能，则只能跟手机相连，无法跟 BLE 的主机模块连接；如果模块开启了传输加密功能，且手机已经配对，然后模块通过拉低 RESTORE 引脚恢复出厂设置，则有可能导致某些手机连接不上模块。此时只需打开手机系统中的蓝牙配置界面，找

到对应的蓝牙设备取消配对即可。



图 3.2 取消蓝牙配对

4. 串口 AT 指令

4.1 AT 指令介绍

以“TTM:”开头的字符串会当成 AT 指令进行解析并执行，并从串口原样返回，之后会追加输出执行结果，“TTM:OK\r\n0”或“TTM:ERR\r\n0”等；不以“TTM:”开头的串口数据包，将被视为透传数据。

在发送 AT 指令的时候，要保证 EN 引脚和 BRTS 引脚接入 GND 电平。

4.1.1 连接间隔设定

模块与手机相连时，会定时进行同步，保证手机与模块一直处于连接状态，且数据交互操作也是在同步的时候进行的；当 BLE 连接间隔较短，则平均功耗会升高；如果 BLE 连接间隔较长，则数据传输的速率会降低。

向串口 RX 输入以下字符串，设定 BLE 连接间隔：“TTM:CIT-X”其中 X=“20”、“50”、“100”、“200”、“300”、“400”、“500”、“1000”、“1500”、“2000”，单位 ms，例如，连接间隔设置为 20ms：

```
TTM:CIT-20
```

在执行完此指令之后，会从串口 TX 得到以下确认：

```
TTM:ERR\r\n0
```

表示更改超时，修改失败；

```
TTM:OK\r\n0
```

表示更改成功，当下一次连接设备的时候，就以新的连接间隔运行。

这个连接间隔设定的成功与否取决于移动设备对连接间隔的限制。

不同的手机的实际连接间隔数据也可能不一样，如魅族手机，其实际连接间隔时间比设置的时间少 25% 左右；iPhone 手机在连接的时候先是 30ms 周期运行 1 分钟，然后切换成设置的连接周期。

此连接间隔掉电不保存，并且更改的指令只有在重新连接后有效。如果当前模块正与手机相连的时候更改连接间隔，那么其连接间隔时间不会立马修改。

4.1.2 获取连接间隔时间

可以使用如下指令来获取连接间隔时间：

```
TTM:CIT-?
```

模块会从 TX 发送“TTM:CIT-20\r\n0”返回连接间隔时间；如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.3 模块重命名

向串口 RX 输入以下字符串，“-”以后为模块名，长度为 15 个字节以内，例如：

```
TTM:REN-ZLGMCU
```

如果在修改名字的过程中，顺便把模块的 MAC 的部分字段现在在名字里边，则可以调用以下指令。修改完成之后，模块会更改为“Name(xxxxxx)”模式显示出更改后的名字以及模块的 MAC 地址后 3 个字节的 Hex 码，便于 iOS 系统识别模块：

```
TTM:RNI-ZLGMCU
```

指令发送完成之后，模块会从 TX 发送“TTM:OK\r\n0”确认，如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

此名称掉电保存。

4.1.4 获取模块名称指令

可以使用如下指令来获取模块名称：

```
TTM:REN-?
```

或者：

```
TTM:RNI-?
```

模块会从 TX 发送“TTM:REN-name\r\n0”返回模块名称，如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.5 波特率设定

串口波特率影响着串口的传输速率与稳定性，用户可以根据不同的条件设置好合适的波特率。

向串口 RX 输入以下字符串，“-”后参数为新波特率，见 AT 指令表，如：

```
TTM:BPS-115200
```

之后会从 TX 收到“TTM:OK\r\n0”确认。如果设置值不在选项中，或者指令格式不正确，则返回“TTM:ERR\r\n0”。

此参数掉电保存。

4.1.6 获取串口波特率指令

可以使用如下指令来获取串口波特率：

```
TTM:BPS-?
```

模块会从 TX 发送“TTM:BPS-115200\r\n0”返回串口波特率；如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.7 获取物理地址 MAC

向串口 RX 输入以下字符串：

```
TTM:MAC-?
```

会从 TX 收到：

```
TTM:MAC-xxxxxxxxxxx\r\n0
```

字符串后面“xxxxxxxxxxx”为 6 字节模块蓝牙地址。

指令格式不正确，则返回“TTM:ERR\r\n0”。

4.1.8 模块复位

向串口 RX 输入以下字符串：

```
TTM:RST-SYSTEMRESET
```

会让模块强制软复位一次。

4.1.9 广播周期设定



向串口 RX 输入以下字符串，设置模块的广播周期， $T=X \times 100\text{ms}$ ，比如设置命令为“TTM:ADP-(2)”，则表示广播周期为 200ms。

```
TTM:ADP-(X)
```

其中 X=“2”、“5”、“10”、“15”、“20”、“25”、“30”、“40”、“50”，其他数据不支持。发送完成后，会返回“TTM:OK\r\n0”确认；若返回“TTM:ERR\r\n0”表明指令格式不正确。

此参数掉电保存。

4.1.10 获取广播周期指令

可以使用如下指令来获取广播周期：

```
TTM:ADP-?
```

模块会从 TX 发送类似“TTM:ADP-(2)\r\n0”字符返回广播周期；如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.11 发射功率设定

BLE 发射功率影响着模块的传输距离，功率越大传输的距离越远，相应的平均功耗就越大。

向串口 RX 输入以下字符串，设置相应的发射功率，单位 dBm。

```
TTM:TPL-(X)
```

其中 X=“-20”、“-18”、“-16”、“-14”、“-12”、“-10”、“-8”、“-6”、“-4”、“-2”、“0”、“2”、“4”，其它参数不支持；发送完成之后，会从 TX 脚收到“TTM:OK\r\n0”确认，并且模块立即使用新的发射功率进行通讯；如果指令格式不对，则会返回“TTM:ERR\r\n0”。

此参数掉电不保存。

4.1.12 获取发射功率指令

可以使用如下指令来获取模块发射功率：

```
TTM:TPL-?
```

模块会从 TX 发送“TTM:TPL-(0)\r\n0”返回模块发射功率；如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.13 数据延时设定

BCTS 引脚用于唤醒外设 MCU，表示当前模块有数据将要通过串口发送出来。由于 MCU 有一定的唤醒时间，因此当 BCTS 拉低到串口发送数据之间需要有一定的延时。这个指令就是设置数据发送的延时的。

向串口 RX 输入以下字符串，设置 BCTS 输出低到串口 TX 输出数据之间的延时，单位 ms。

```
TTM:CDL-X
```

其中 X=“0”、“10”、“20”、“30”之一，如果指令无误，会收到指令：

```
TTM:OK\r\n0
```

确认，如果指令格式不正确，则会返回：

TTM:ERR\r\n0

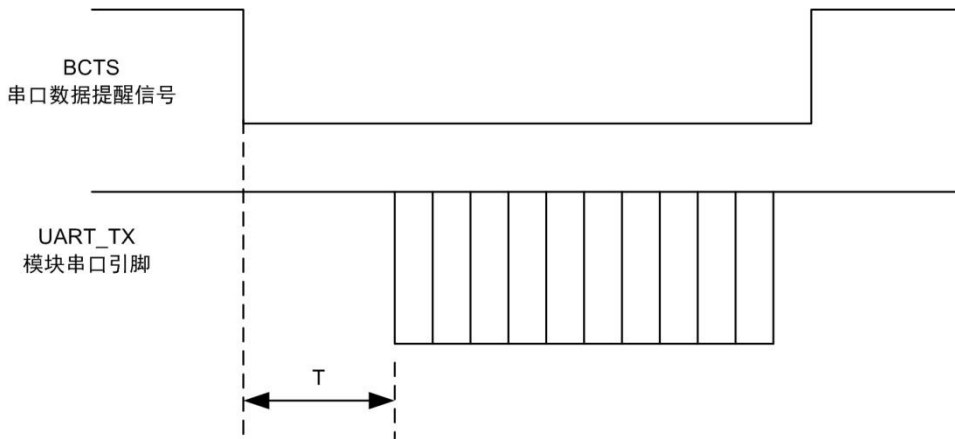


图 4.1 模块串口输出数据延时设定示意图

为了让用户 MCU 有足够的时间从睡眠中唤醒，准备接收，模块提供了这个延时设定，在模块串口有数据发出之前会置低 BCTS，而 BCTS 输出低到模块 TX 输出数据之间的延时由此参数设定。

此参数掉电保存。

4.1.14 获取数据延时指令

可以使用如下指令来获取数据延时：

TTM:CDL-?

模块会从 TX 发送 “TTM:CDL-0\r\n0” 返回数据延时时间；如果指令格式不正确，则会返回 “TTM:ERR\r\n0”。

4.1.15 模块低功耗指令

模块在使用的时候，可以通过 EN 引脚接高电平来让模块处于低功耗模式，也可以通过指令的方式，让模块进入低功耗。两种模式的区分是，如果是 EN 引脚控制低功耗，则无论 BRTS 引脚处于何种电平、何种边沿，都不能唤醒模块，只有当 EN 引脚接 GND 的时候，才能够唤醒；如果指令控制低功耗，则无论此时 EN 和 BRTS 引脚处于何种电平，模块都不会被唤醒，只有 EN 引脚或者 BRTS 引脚进行边沿跳变的时候才能唤醒模块。

向串口 RX 输入以下字串，设置进入掉电模式。

TTM:PWR-POWERDOWN

如果解析成功，则会在串口上返回 “TTM:OK\r\n0” 的字符，过了 2s 之后，模块进入低功耗模式；如果解析失败，则返回 “TTM:ERR\r\n0”。

4.1.16 获取软件版本指令

向串口 RX 输入以下字串，用于获取软件版本号。

TTM:VER-VERSION

如果指令能够顺利解析出来，则会在串口助手上打印出 “TTM:Vx.xx”，其中 “x.xx” 为软件的版本号。

4.1.17 获取当前模块的连接状态

向串口 RX 输入以下字串，用于获取当前模块的连接状态：

```
TTM:CON-CONSTATE
```

如果模块跟手机处于连接状态，则返回：

```
TTM:CONNECT\r\n0
```

如果处于广播模式，则返回：

```
TTM:DISCONNECT\r\n0
```

4.1.18 断开模块连接指令

向串口 RX 输入以下字符串，用于获取当前模块的连接状态：

```
TTM:DSC-DISCON
```

如果解析成功，则会在串口上返回“TTM:OK\r\n0”的字符，且断开模块与手机的连接；如果解析失败，则返回“TTM:ERR\r\n0”。

4.1.19 修改配对码指令

配对码设置指令“TTM:PWD-xxxxxx”，其中“xxxxxx”表示将要修改的配对码（数字类型的字符串），如果此位写“000000”表示取消配对码；例如：

- 1、模块出厂时是不设配对码的，此时想把配对码改为“123456”，则需要键入字符串：

```
TTM:PWD-123456
```

- 2、在此之后，如果想把配对码从“123456”改为“666666”，需要键入指令：

```
TTM:PWD-666666
```

- 3、如果此时想从“666666”取消配对密码，则键入指令：

```
TTM:PWD-000000
```

恢复出厂操作可以强制取消配对密码。

如果模块设置了配对功能，在模块与手机刚连接的 10s 之内通过手机 APP 发送配对码字符。如果配对码错误或者超过 10s 没有发送配对码，则模块会主动断开与手机的连接，这样可以保证模块只能被特有的 APP 连接，防止被别的手机及 APP 软件占用连接。

4.1.20 获取配对码的指令

可以使用如下指令来获取数据延时：

```
TTM:PWD-?
```

模块会从 TX 发送“TTM:PWD-xxxxxx\r\n0”返回系统配对码；如果指令格式不正确，则会返回“TTM:ERR\r\n0”。

4.1.21 设置传输加密指令

当设置模块传输加密功能，则传输的数据不会被第三方工具破解。

向串口 RX 输入以下字符串，则开启传输加密功能：

```
TTM:ENC-TRUE
```

向串口 RX 输入以下字符串，则关闭传输加密功能：

```
TTM:ENC-FALSE
```

如果指令无误，会收到指令：

```
TTM:OK\r\n0
```


如果指令格式不正确，则会返回：

```
TTM:ERR\r\n0
```

4.1.22 获取传输加密指令

可以使用如下指令来获取模块是否开启传输加密功能：

```
TTM:ENC-?
```

如果模块开启传输加密功能，返回指令：

```
TTM:ENC-TRUE\r\n0
```

如果模块开启传输加密功能，则会返回：

```
TTM:ENC-FALSE\r\n0
```

4.2 AT 指令总表

表 4.1 AT 指令表

AT 指令格式	掉电保存	参数说明	可能的回应	含义
TTM:CIT-X (重新连接后有效)	否	X="20"、"50"、"100"、"200"、 "300"、"400"、"500"、"1000"、 "1500"、"2000"设置相应的 BLE 连接间隔，单位 ms	TTM:OK\r\n0 TTM:ERR\r\n0	设置成功 错误参数
TTM:CIT-?	-	获取 BLE 的连接间隔	TTM:CIT-x\r\n0 TTM:ERR\r\n0	获取成功 获取失败
TTM:REN-Name	是	Name: 新模块名，长度 为 15 字节以内的任意字符串	TTM:OK\r\n0 TTM:ERR\r\n0	设置成功 设置失败
TTM:RNI-Name	是	Name: 新模块名，长度 为 15 字节以内的任意字符串。 修改完成之后名字中会显示 MAC 地址后三个字节	TTM:OK\r\n0 TTM:ERR\r\n0	设置成功 设置失败
TTM:REN-? 或 TTM:RNI-?	-	获取模块的名称	TTM:REN-x\r\n0 TTM:ERR\r\n0	获取成功 获取失败
TTM:BPS-X	是	X="4800"、"9600"、"19200"、 "38400"、"57600"、"115200"	TTM:BPSSETAFTER2S \r\n TTM:ERR\r\n0	设置成功 设置失败
TTM:BPS-?	-	获取模块的串口波特率	TTM:BPS-x\r\n0 TTM:ERR\r\n0	获取成功 获取失败
TTM:MAC-?	—	获取 MAC 地址	TTM:MAC-xxxxxxxxxxx\ \r\n0 (xxxxxxxxxxxx 为模块 MAC 地址)	返回 MAC 地址
TTM:RST-SY STEMRESET	—	让模块系统复位	无	复位模块

续上表

AT 指令格式	掉电保存	参数说明	可能的回应	含义
TTM:ADP-(X)	是	X="2"、"5"、"10"、"15"、"20"、 "25"、"30"、"40"、"50" 设置相应的广播周期， $T=X \times 100\text{ms}$	TTM:OK\r\n\0 TTM:ERR\r\n\0	设置广播周期， 如设置为"5"， 则为 500ms
TTM:ADP-?	-	获取广播周期数据	TTM:ADP-(x)\r\n\0 TTM:ERR\r\n\0	获取成功 获取失败
TTM:TPL-(X)	否	X="-20"、"-18"、"-16"、"-14"、 "-12"、"-10"、"-8"、"-6"、"-4"、 "-2"、"0"、"2"、"4" 设置相应的发射功率，单位 dBm	TTM:OK\r\n\0 TTM:ERR\r\n\0	发射功率设定
TTM:TPL-?	-	获取模块的发射功率	TTM:TPL-(x)\r\n\0 TTM:ERR\r\n\0	获取成功 获取失败
TTM:CDL-X	是	X="0"、"10"、"20"、"30" 设置 BCTS 输出低到串口输出 数据之间的延时，单位 ms	TTM:OK\r\n\0 TTM:ERR\r\n\0	实际延时不大于 X 设置的数 据
TTM:CDL-?	-	获取广播周期数据	TTM:CDL-x\r\n\0 TTM:ERR\r\n\0	获取成功 获取失败
TTM:PWR-POWERD OWN	-	进入低功耗模式的指令	TTM:OK\r\n\0 TTM:ERR\r\n\0	解析成功 解析失败
TTM:VER-VERSION	-	获取版本号的指令	TTM:Vx.xx\r\n\0 x.xx"为版本号 TTM:ERR\r\n\0	返回版本号 解析失败
TTM:CON-CONSTAT E	-	主动断开连接指令	TTM:DISCONNECT\r\n\0 TTM:CONNECT\r\n\0 TTM:ERR\r\n\0	断开连接状态 连接状态 解析失败
TTM:DSC-DISCON	-	主动断开连接指令	TTM:OK\r\n\0 TTM:ERR\r\n\0	断开连接成功 解析失败
TTM:PWD-xxxxxx	是	修改模块配对码的指令 xxxxxx: 新配对码	TTM:OK\r\n\0 TTM:ERR\r\n\0	密码更改成功 密码更新失败
TTM:PWD-?	-	获取模块的配对码	TTM:PWD-xxxxxx\r\n\0 TTM:ERR\r\n\0	获取成功 获取失败
TTM:ENC-TRUE TTM:ENC-FALSE	是	开启传输加密功能 关闭传输加密功能	TTM:OK\r\n\0 TTM:ERR\r\n\0	设置成功 设置失败

续上表

AT 指令格式	掉电保存	参数说明	可能的回应	含义
TTM:ENC-?	-	获取模块的加密状态	TTM:ENC-TRUE TTM:ENC-FALSE	加密模式 非加密模式

5. 系统复位与恢复

5.1 模块复位

让模块复位有两种方法：

- 1、复位引脚 RESET 接一段时间的低电平；
- 2、使用 AT 指令复位模块(详见 4.1.8 章节)。

5.2 恢复出厂设置

使用硬件 RESTORE 脚位（见表 2.1），上电 30 秒内，将此脚位拉低 10 秒后恢复到出厂设置（深度恢复），并立即复位。此脚位带内部上拉，在悬空状态下不会进入此模式。

出厂设置的具体参数如下：

- 串口波特率，恢复到 9600bps；
- 无线发射功率恢复到 0dBm；
- 设备连接周期恢复到 20ms；
- 设备名称，恢复到“ZLG9021P0-1(xxxxxx)”（名字+MAC 地址）；
- 串口数据延时，恢复到 0；
- 广播周期，恢复到 2(200ms)；
- 配对密码恢复到无密码状态。

注：RESTORE 脚位的特殊性，在电路设计中，需避免上电前 30 秒持续下地，否则会进入恢复模式。

6. 主机参考代码

模块间是用 BCTS, BRTS 两个 IO 口进行发送接收的通知和控制。

这两个 IO 常态高阻, 置低触发, 如果模块有数据要发, 置低 BCTS 通知单片机接收, 如果单片机有数据要发, 置低 BRTS 通知模块接收。示意性代码如下:

```
void main(void)
{
    EN = 0; //使能 EN, 开始广播
    while(!BLEMoudleAck("TTM:OK\r\n0")); //等待手机端扫描, 连接
    //等待连接成功, 也可加入限时等待
    //也可判断连接提示信号线的电平
    BRTS = 0; //BRTS 置低通知 BLE 模块准备接收
    halMcuWaitMs(2); //延迟 2ms
    UARTWrite( HAL_UART_PORT_0, "TTM:CIT-100", 14);
    //修改连接间隔, 从串口得到确认:
    halMcuWaitMs(5); //延迟 5ms,确保数据已经发出
    BRTS = 1; //RTS 置高, 发送完毕
    while(!BLEMoudleAck("TTM:OK\r\n0")); //等待设置成功, 也可加入限时等待
    while(1){ //循环收发测试
        while(1){
            if(BCTS == 0){ //检测, 若 BCTS 置低则准备接收
                while(BCTS==0); //等待发送完毕, 也可限时等待
                if(UARTRead(uartBuffer) == SUCCESS) //串口读取数据
                { ... ... } //使用数据
            }
            BRTS = 0; //RTS 置低通知 BLE 模块准备接收
            halMcuWaitMs(2); //延迟 2ms
            send_TX("1234567890"); //发送任意数据 (200byte 以内)
            halMcuWaitMs(5); //延迟 5ms,确保数据已经发出
            BRTS = 1; //RTS 置高, 发送完毕
            halMcuWaitMs(20); //延迟再发下一个包, 延时视包大小而定
        }
    }
}
```

7. 免责声明

本着为用户提供更好服务的原则，广州立功科技股份有限公司（下称“立功科技”）在本手册中将尽可能地向用户呈现详实、准确的产品信息。但鉴于本手册的内容具有一定的时效性，立功科技不能完全保证该文档在任何时段的时效性与适用性。立功科技有权在没有通知的情况下对本手册上的内容进行更新，恕不另行通知。为了得到最新版本的信息，请尊敬的用户定时访问立功科技官方网站或者与立功科技工作人员联系。感谢您的包容与支持！

专业 · 专注成就梦想

Dreams come true with professionalism and dedication.

广州立功科技股份有限公司

更多详情请访问
www.zlgmcu.com

欢迎拨打全国服务热线
400-888-2705

