



密级：公开资料

低功耗蓝牙（BLE）4.1 透传规格书

文档版本：V2.0（旧脚位）

深圳市昇润科技有限公司

2017年07月01日

版权所有

目 录

1. 概述.....	1
1.1. 功能简介.....	1
1.2. 主要功能特点.....	1
2. 硬件规格.....	2
2.1. 模组对比.....	2
2.2. 模组脚位及尺寸.....	3
2.3. 模组脚位定义说明.....	5
2.3.1. 小模组引脚功能.....	5
2.3.2. 大模组引脚功能.....	6
2.4. 模组电气特性.....	8
2.5. 模组蓝牙功耗对照表.....	9
2.6. 工作模式.....	10
2.7. 电路连接.....	10
2.7.1. 模块通信接口选择.....	10
2.7.2. 两种通信方式连接示意图.....	11
3. 软件规格.....	13
3.1. 命令速查表.....	13
3.2. 指令格式说明.....	17
3.2.1. 模块通信接口选择.....	17
3.2.2. UART 命令格式说明.....	17
3.2.3. SPI 命令格式说明.....	18
3.3. 指令集说明.....	20
4. 蓝牙协议（APP 接口）.....	37
4.1. 模块 UUID 说明.....	37
4.2. 模组蓝牙通道操作说明.....	37
5. 工具使用说明.....	38
5.1. 手机 APP（TTC-BLE）.....	38
5.1.1. 手机 APP 下载.....	38
5.1.2. 数据加密选项设置.....	39
5.1.3. 读取/设置模组参数.....	39
5.1.4. 接收/发送透传数据.....	40
5.2. PC 端软件（BleConfig）.....	40
6. 文件修订说明.....	41
7. 联系我们.....	42

1. 概述

1.1. 功能简介

模块支持从机模式。支持桥接模式（透传模式）和直驱模式。模块通过初始设置后会自动进行广播，已打开特定 APP 的手机会对其进行扫描和对接，成功之后便可以通过 BLE 协议对其进行控制。

桥接模式：用户 CPU 可以通过模块的通用 UART 或 SPI 和移动设备进行双向通讯，用户也可以通过特定的指令，对模块的蓝牙参数进行管理控制。用户数据的具体含义由上层应用程序自行定义。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 UART 或 SPI 发送给用户的 CPU。模块收到来自用户 CPU 数据包后，将自动转发给移动设备。此模式下的开发，用户必须负责主 CPU 的代码设计，以及智能移动设备端 APP 代码设计。

直驱模式：用户对模块进行简单外围扩展，APP 通过 BLE 协议直接对模块进行驱动，完成智能移动设备对模块的监管和控制。此模式下的软件开发，用户只须负责智能移动设备端 APP 代码设计。

1.2. 主要功能特点

- 使用简单，无需任何蓝牙协议栈应用经验。
- 支持多连接蓝牙数据传输功能，模组作为蓝牙从机，最多可连接 3 个主机。
- 用户接口使用通用 UART 设计，全双工双向通讯，最低波特率支持 75bps，最高支持波特率 375000bps。
- 同时支持桥接模式和直接驱动模式(无需额外 CPU)；直驱模式支持 SPI/UART 接口。
- 默认 40ms 连接间隔。随着连入设备的增加，后连入的设备连接间隔会逐步递增 25ms。
- 支持 AT/SPI 指令软件复位模块，获取 MAC 地址。
- 支持 AT/SPI 指令调整蓝牙连接间隔，控制不同的转发速率。(动态功耗调整)
- 支持 APP/SPI/UART 指令调整发射功率，修改广播间隔/连接超时时间/产品连接延迟，修改 UART 波特率，修改模块名，均会掉电保存。
- UART 数据包长度，可以是 80byte 以下(含 80byte)的任意长度。(自动分包)
- 模组与一个主机连接时，双向通信 1Kbyte/s。
- 模组同时与 3 个主机连接时，双向通信 80byte/s。
- 支持 APP/SPI/AT 修改模块名称，掉电保存，修改 UART 波特率，产品识别码，自定义广播内容，广播周期，均掉电保存。
- 支持移动设备 APP/SPI/AT 对模块进行复位，设置发射功率。
- 支持移动设备 APP/SPI/AT 调节蓝牙连接间隔，掉电保存。(动态功耗调整)
- APP/SPI/AT 均可操作所有 IO 外扩。
- 支持连接状态，广播状态提示脚/普通 IO 灵活配置。
- 16 个双向可编程 IO，全低功耗运行。(照明控制，遥控玩具，等各种输入输出开关量应用)
- 2 路 ADC 输入(12 bit)，使能/禁止，采样周期自由配置，可以设定均值滤波。(测温湿度，光度等应用)
- 四路可编程 PWM 输出(最高 2MHz)。(调光，调速等应用)
- 模块端 RSSI 连续采集。(寻物防丢报警应用)

- 支持模块电量提示，电量读取。（设备电量提醒）
- 支持内部 RTC 实时时钟，APP 端可随时同步校准。
- 支持 IO 配置和输出状态保存功能，可自定义默认的初始化状态。
- 支持浅恢复和深度恢复模式，灵活恢复用户数据，而保留产品必须配置。
- 支持从 UART/SPI 获取蓝牙连接状态（连接，正常断线和超时断线）提示。
- 支持低电平使能模式和脉宽使能模式，支持远程关机。
- 极低功耗的待机模式。

2. 硬件规格

2.1. 模组对比

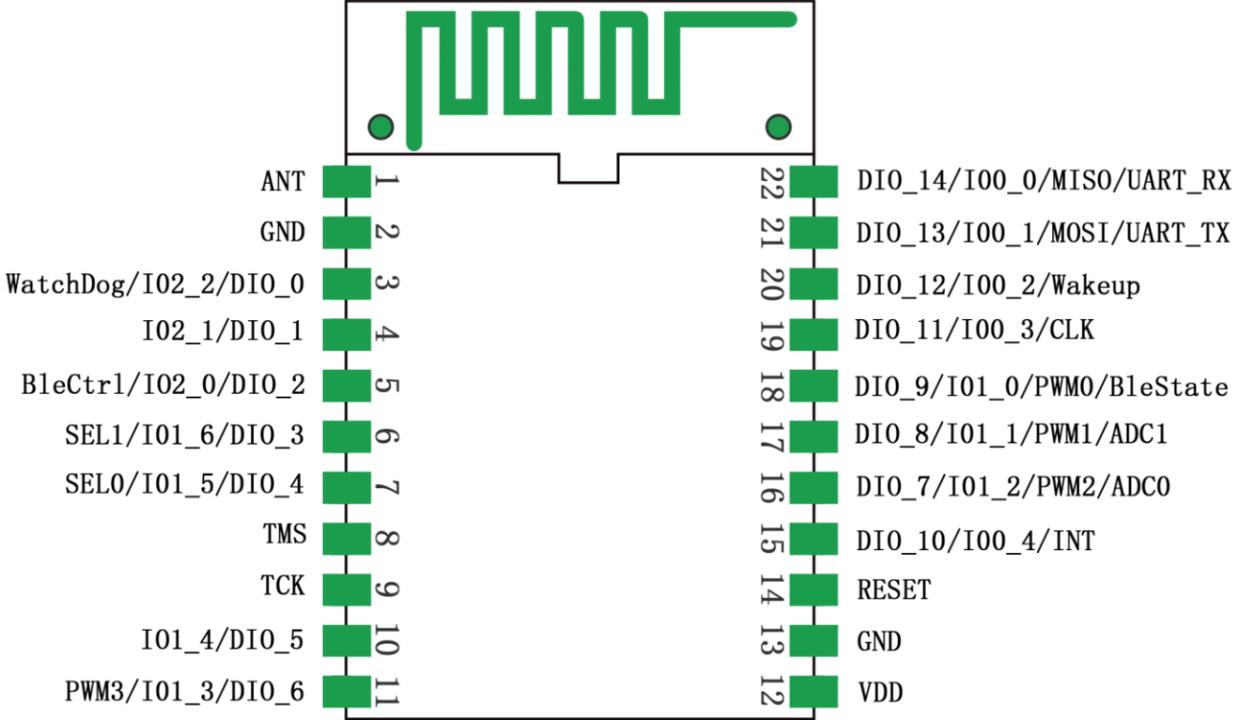
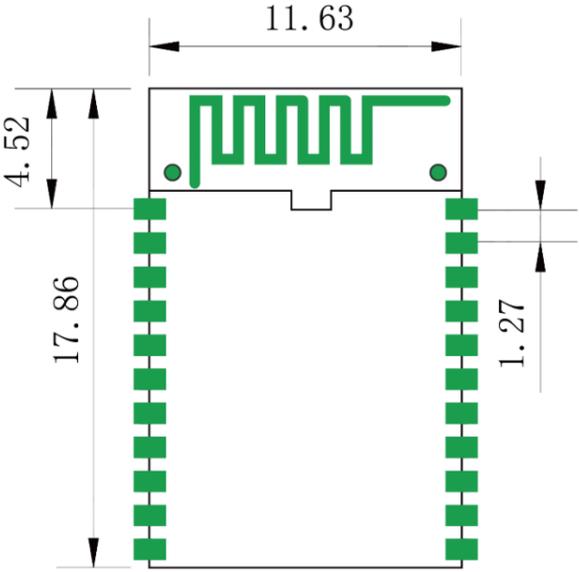
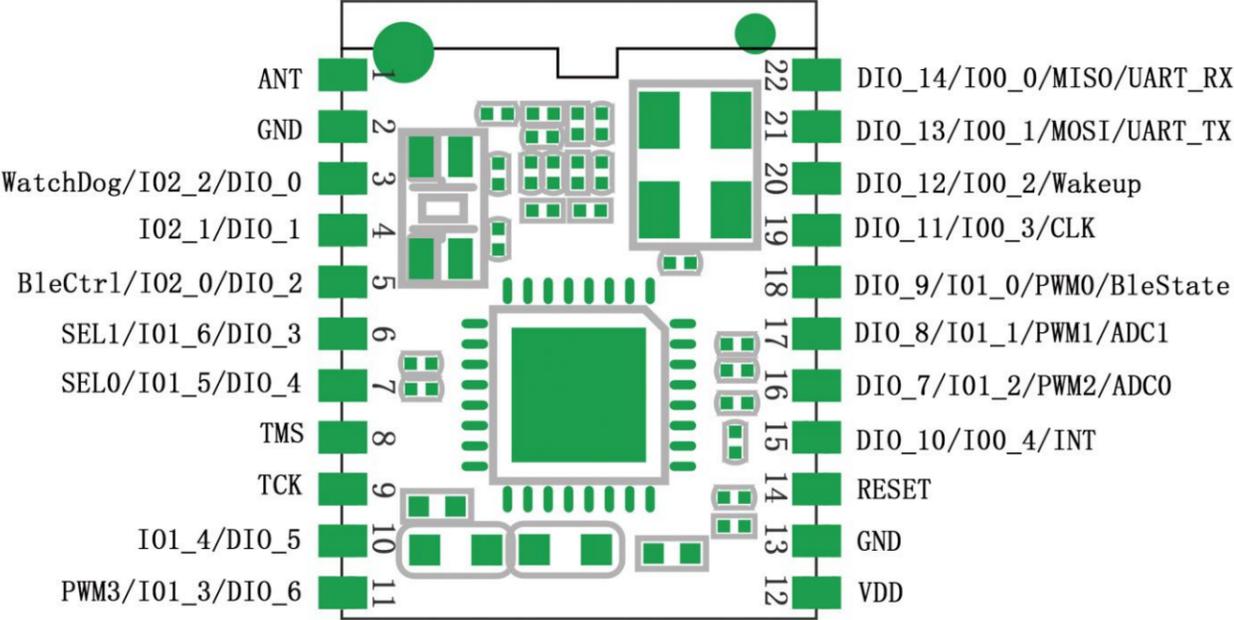
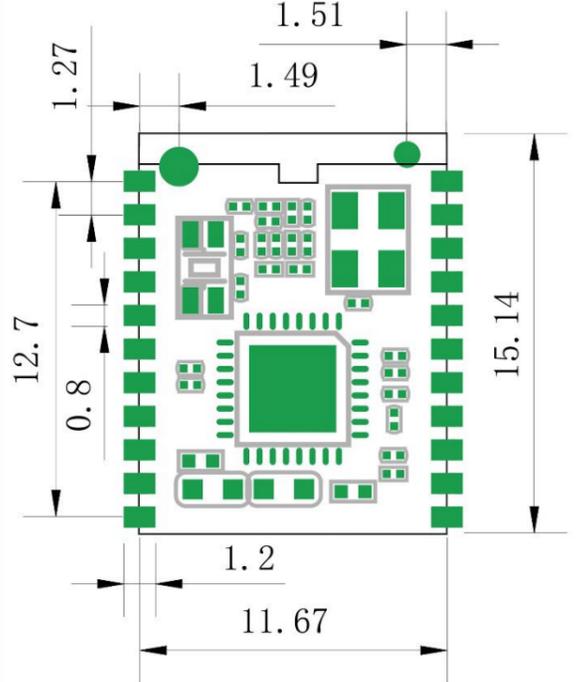
点击对应的产品名称，可查看模组的实物图、脚位图及尺寸，如 [2.2 节](#)。

类型	产品名称	通讯方式	尺寸(mm)	脚位数	天线形式
小模块	HY-264018	UART/SPI	11.63*17.86*2.0	15	PCB 天线
	HY-264024	UART/SPI	11.67*15.14*2.0	15	软天线
	HY-264025	UART/SPI	11.67*17.22*2.3	15	IPEX 天线座
大模块	HY-264027_V1	UART/SPI	15.23*25.25*2.0	31	PCB 天线

备注：

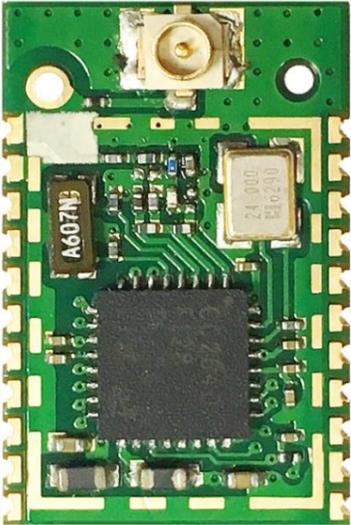
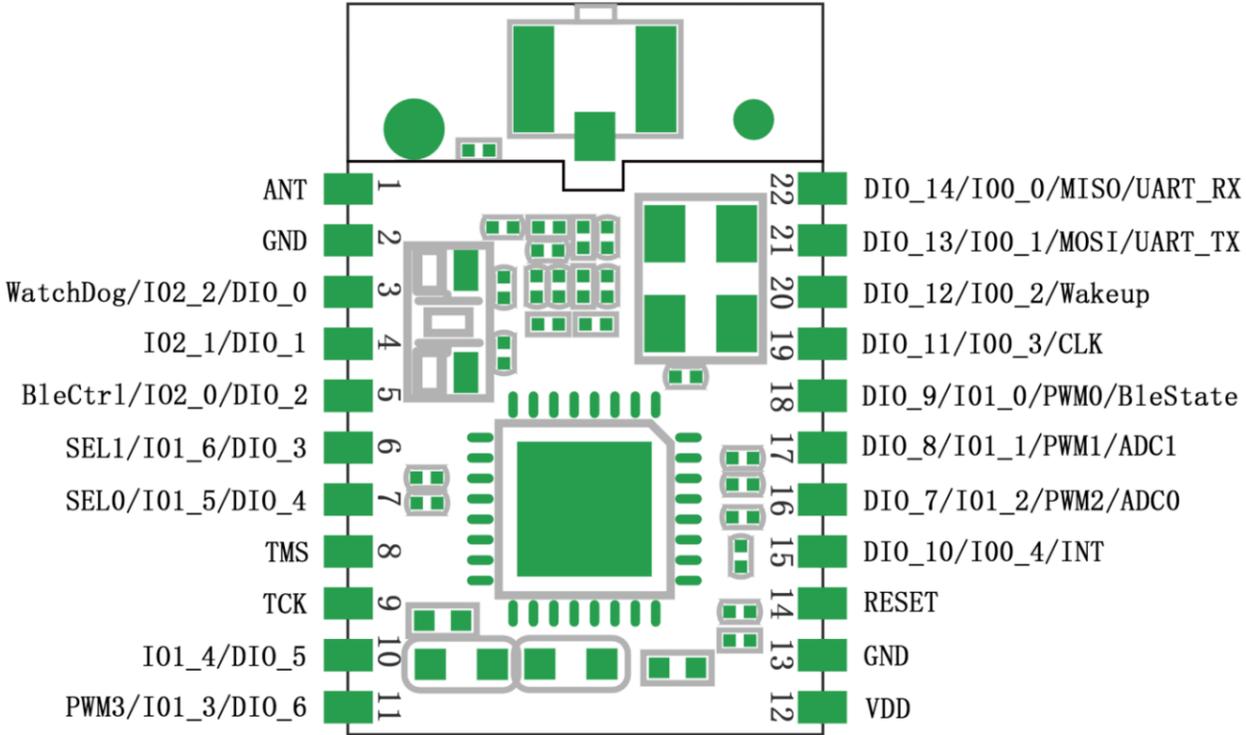
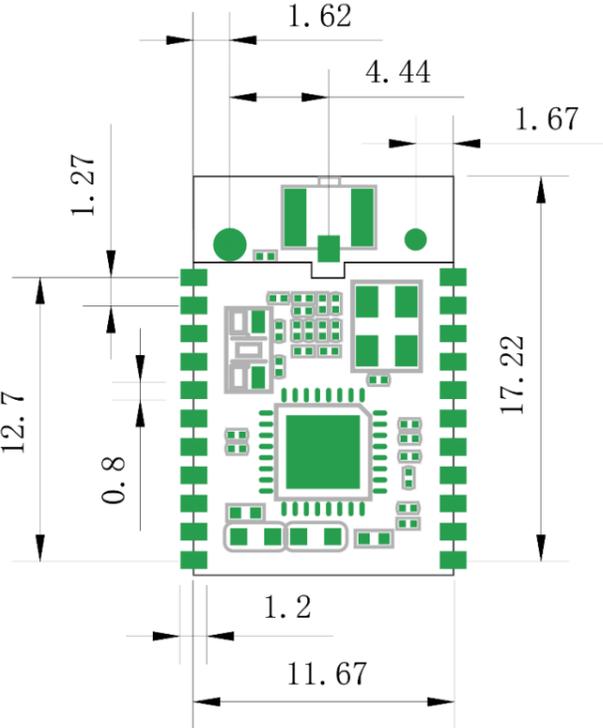
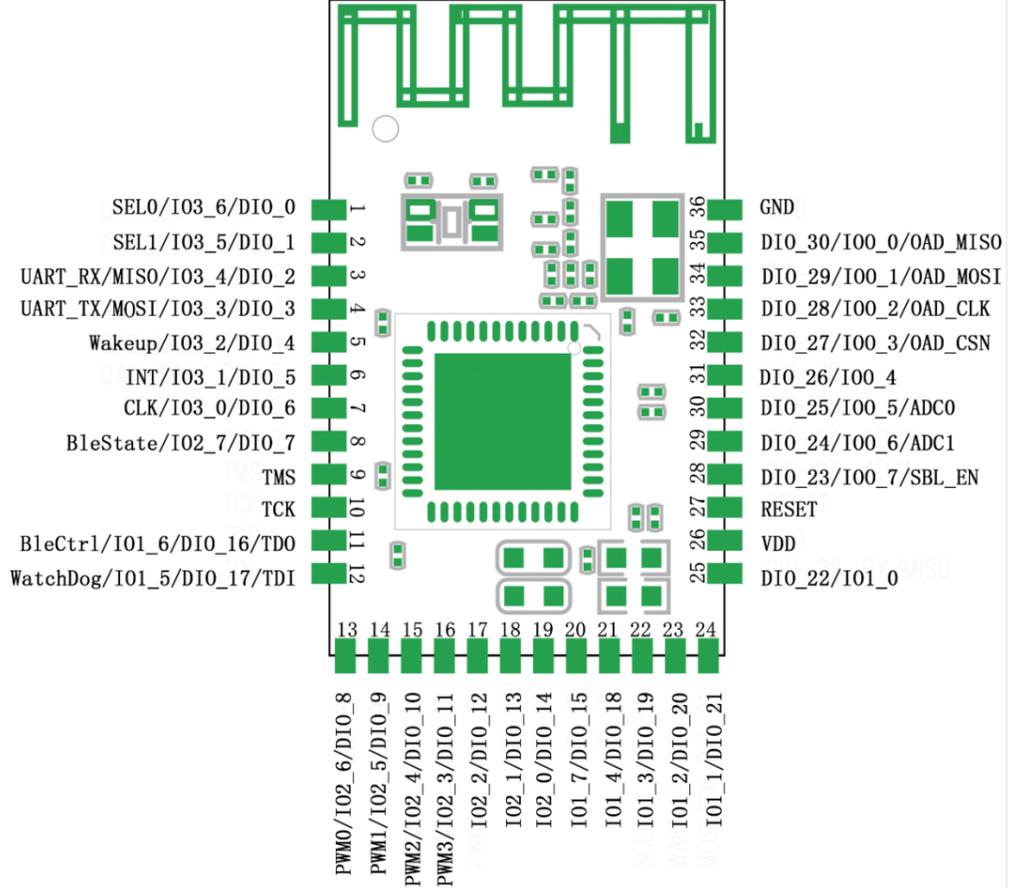
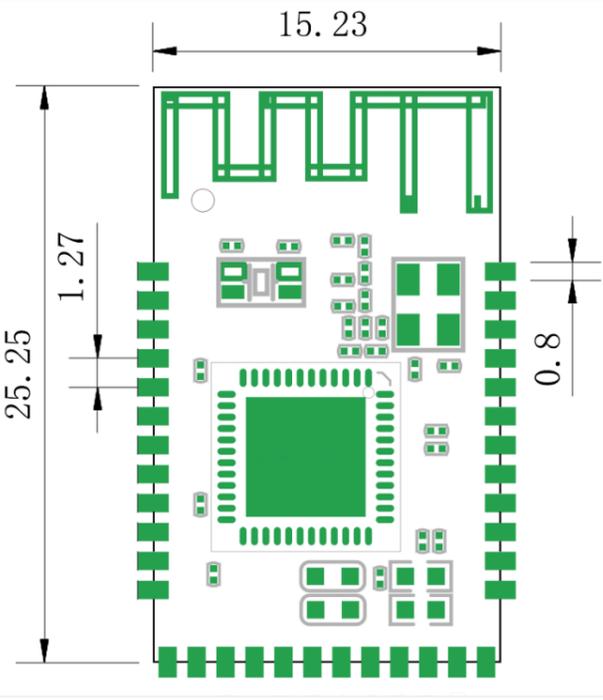
- (1) 每种模组均支持 UART/SPI 两种通信方式（可任选一种通信方式）；
- (2) 通讯方式的选择参见 [2.7.1 节](#)，两种通讯方式的软件特性见 [3.2 节](#)。

2.2. 模组脚位及尺寸

规格参数	实物图	引脚图	尺寸图																																												
<p>名称: HY-264018 芯片: CC2640 5*5 通信方式: UART/SPI 尺寸(mm): 11.63*17.86*2.0 脚位数: 15 天线形式: PCB 天线</p>		 <table border="1" data-bbox="926 451 2160 1003"> <tr><td>ANT</td><td>1</td><td>22</td><td>DIO_14/I00_0/MISO/UART_RX</td></tr> <tr><td>GND</td><td>2</td><td>21</td><td>DIO_13/I00_1/MOSI/UART_TX</td></tr> <tr><td>WatchDog/I02_2/DIO_0</td><td>3</td><td>20</td><td>DIO_12/I00_2/Wakeup</td></tr> <tr><td>I02_1/DIO_1</td><td>4</td><td>19</td><td>DIO_11/I00_3/CLK</td></tr> <tr><td>BleCtrl/I02_0/DIO_2</td><td>5</td><td>18</td><td>DIO_9/I01_0/PWM0/BleState</td></tr> <tr><td>SEL1/I01_6/DIO_3</td><td>6</td><td>17</td><td>DIO_8/I01_1/PWM1/ADC1</td></tr> <tr><td>SEL0/I01_5/DIO_4</td><td>7</td><td>16</td><td>DIO_7/I01_2/PWM2/ADC0</td></tr> <tr><td>TMS</td><td>8</td><td>15</td><td>DIO_10/I00_4/INT</td></tr> <tr><td>TCK</td><td>9</td><td>14</td><td>RESET</td></tr> <tr><td>I01_4/DIO_5</td><td>10</td><td>13</td><td>GND</td></tr> <tr><td>PWM3/I01_3/DIO_6</td><td>11</td><td>12</td><td>VDD</td></tr> </table>	ANT	1	22	DIO_14/I00_0/MISO/UART_RX	GND	2	21	DIO_13/I00_1/MOSI/UART_TX	WatchDog/I02_2/DIO_0	3	20	DIO_12/I00_2/Wakeup	I02_1/DIO_1	4	19	DIO_11/I00_3/CLK	BleCtrl/I02_0/DIO_2	5	18	DIO_9/I01_0/PWM0/BleState	SEL1/I01_6/DIO_3	6	17	DIO_8/I01_1/PWM1/ADC1	SEL0/I01_5/DIO_4	7	16	DIO_7/I01_2/PWM2/ADC0	TMS	8	15	DIO_10/I00_4/INT	TCK	9	14	RESET	I01_4/DIO_5	10	13	GND	PWM3/I01_3/DIO_6	11	12	VDD	
ANT	1	22	DIO_14/I00_0/MISO/UART_RX																																												
GND	2	21	DIO_13/I00_1/MOSI/UART_TX																																												
WatchDog/I02_2/DIO_0	3	20	DIO_12/I00_2/Wakeup																																												
I02_1/DIO_1	4	19	DIO_11/I00_3/CLK																																												
BleCtrl/I02_0/DIO_2	5	18	DIO_9/I01_0/PWM0/BleState																																												
SEL1/I01_6/DIO_3	6	17	DIO_8/I01_1/PWM1/ADC1																																												
SEL0/I01_5/DIO_4	7	16	DIO_7/I01_2/PWM2/ADC0																																												
TMS	8	15	DIO_10/I00_4/INT																																												
TCK	9	14	RESET																																												
I01_4/DIO_5	10	13	GND																																												
PWM3/I01_3/DIO_6	11	12	VDD																																												
<p>名称: HY-264024 芯片: CC2640 5*5 通信方式: UART/SPI 尺寸(mm): 11.67*15.14*2.0 脚位数: 15 天线形式: 软天线</p>		 <table border="1" data-bbox="926 1228 2160 1785"> <tr><td>ANT</td><td>1</td><td>22</td><td>DIO_14/I00_0/MISO/UART_RX</td></tr> <tr><td>GND</td><td>2</td><td>21</td><td>DIO_13/I00_1/MOSI/UART_TX</td></tr> <tr><td>WatchDog/I02_2/DIO_0</td><td>3</td><td>20</td><td>DIO_12/I00_2/Wakeup</td></tr> <tr><td>I02_1/DIO_1</td><td>4</td><td>19</td><td>DIO_11/I00_3/CLK</td></tr> <tr><td>BleCtrl/I02_0/DIO_2</td><td>5</td><td>18</td><td>DIO_9/I01_0/PWM0/BleState</td></tr> <tr><td>SEL1/I01_6/DIO_3</td><td>6</td><td>17</td><td>DIO_8/I01_1/PWM1/ADC1</td></tr> <tr><td>SEL0/I01_5/DIO_4</td><td>7</td><td>16</td><td>DIO_7/I01_2/PWM2/ADC0</td></tr> <tr><td>TMS</td><td>8</td><td>15</td><td>DIO_10/I00_4/INT</td></tr> <tr><td>TCK</td><td>9</td><td>14</td><td>RESET</td></tr> <tr><td>I01_4/DIO_5</td><td>10</td><td>13</td><td>GND</td></tr> <tr><td>PWM3/I01_3/DIO_6</td><td>11</td><td>12</td><td>VDD</td></tr> </table>	ANT	1	22	DIO_14/I00_0/MISO/UART_RX	GND	2	21	DIO_13/I00_1/MOSI/UART_TX	WatchDog/I02_2/DIO_0	3	20	DIO_12/I00_2/Wakeup	I02_1/DIO_1	4	19	DIO_11/I00_3/CLK	BleCtrl/I02_0/DIO_2	5	18	DIO_9/I01_0/PWM0/BleState	SEL1/I01_6/DIO_3	6	17	DIO_8/I01_1/PWM1/ADC1	SEL0/I01_5/DIO_4	7	16	DIO_7/I01_2/PWM2/ADC0	TMS	8	15	DIO_10/I00_4/INT	TCK	9	14	RESET	I01_4/DIO_5	10	13	GND	PWM3/I01_3/DIO_6	11	12	VDD	
ANT	1	22	DIO_14/I00_0/MISO/UART_RX																																												
GND	2	21	DIO_13/I00_1/MOSI/UART_TX																																												
WatchDog/I02_2/DIO_0	3	20	DIO_12/I00_2/Wakeup																																												
I02_1/DIO_1	4	19	DIO_11/I00_3/CLK																																												
BleCtrl/I02_0/DIO_2	5	18	DIO_9/I01_0/PWM0/BleState																																												
SEL1/I01_6/DIO_3	6	17	DIO_8/I01_1/PWM1/ADC1																																												
SEL0/I01_5/DIO_4	7	16	DIO_7/I01_2/PWM2/ADC0																																												
TMS	8	15	DIO_10/I00_4/INT																																												
TCK	9	14	RESET																																												
I01_4/DIO_5	10	13	GND																																												
PWM3/I01_3/DIO_6	11	12	VDD																																												

备注: 通信距离测试环境: 默认发射功率 0 dBm, 以模块与 iPhone 6S 手机面对面自由空间测试

续上表

规格参数	实物图	引脚图	尺寸图
<p>名称: HY-264025 芯片: CC2640 5*5 通信方式: UART/SPI 尺寸(mm): 11.67*17.22*2.3 脚位数: 15 天线形式: IPEX 天线座</p>		<p>引脚图</p> 	
<p>名称: HY-264027 V1 芯片: CC2640 7*7 通信方式: UART/SPI 尺寸(mm): 15.23*25.25*2.0 脚位数: 31 天线形式: PCB 天线</p>		<p>引脚图</p> 	

备注: 通信距离测试环境: 默认发射功率 0 dBm, 以模块与 iPhone 6S 手机面对面自由空间测试

2.3. 模组脚位定义说明

CC2640 透传程序分为新旧脚位两种，请与我司业务确认软件与规格书版本对应。

2.3.1. 小模组引脚功能

小模组使用的是 CC2640 5*5 的芯片，小模组型号为 HY-264018/HY-264024/HY-264025，三个模组引脚定义相同，说明如下：

引脚序号	引脚名	功能描述	功能对应
1	ANT	天线引脚	可供用户外接天线使用
2	GND	地	BLE 模块接地引脚
3	IO2_2	通用 IO 口引脚	该引脚属于 IO2 引脚
		看门狗输出引脚	当设备正常运行时，设备以 500ms 为周期进行 IO 口电平变化。
4	IO2_1	通用 IO 口引脚	该引脚属于 IO2 引脚
		OAD_CS 片选引脚	OAD 升级时，Flash 片选引脚
5	IO2_0	通用 IO 口引脚	该引脚属于 IO2 引脚
		OAD_MISO 引脚	OAD 升级，BLE 模块输入端，Flash 输出端
		蓝牙开关引脚	若该功能使能，拉低引脚则打开蓝牙广播，拉高引脚则关闭蓝牙广播
6	IO1_6	通用 IO 口引脚	该引脚属于 IO1 引脚
		SEL1 通道选择引脚	MCU 与 BLE 通讯方式选择引脚。参见表 1-1
7	IO1_5	通用 IO 口引脚	该引脚属于 IO1 引脚
		SEL0 通道选择引脚	MCU 与 BLE 通讯方式选择引脚。参见表 1-1
8	TMS	烧写程序	\
9	TCK	烧写程序	\
10	IO1_4	通用 IO 口引脚	该引脚属于 IO1 引脚
		OAD_MOSI 引脚	OAD 升级，BLE 模块输出端，Flash 输入端
11	IO1_3	通用 IO 口引脚	该引脚属于 IO1 引脚
		PWM3 输出引脚	16 位 PWM 输出引脚，占空比可调范围 1%-99%，输出频率范围 1KHz-2MHz
		OAD_CLK 引脚	OAD 升级时，Flash 时钟信号端
12	VDD_EB	电源	BLE 模块供电引脚，电压范围 1.8-3.8V
13	GND	地	BLE 模块接地引脚
14	RESET	复位引脚	BLE 硬件复位引脚，低电平复位
15	IO0_4	通用 IO 口引脚	该引脚属于 IO0 引脚
		INT 引脚	中断输出引脚

16	I01_2	通用 IO 口引脚	该引脚属于 I01 引脚
		PWM2 输出引脚	16 位 PWM 输出引脚，占空比可调范围 1%-99%，输出频率范围 1KHz-2MHz
		ADC0 引脚	12 位 ADC，通道 1
17	I01_1	通用 IO 口引脚	该引脚属于 I01 引脚
		PWM1 输出引脚	16 位 PWM 输出引脚，占空比可调范围 1%-99%，输出频率范围 1KHz-2MHz
		ADC1 引脚	12 位 ADC，通道 1
18	I01_0	通用 IO 口引脚	该引脚属于 I01 引脚
		PWM0 输出引脚	16 位 PWM 输出引脚，占空比可调范围 1%-99%，输出频率范围 1KHz-2MHz
		蓝牙指示引脚	若该功能使能，蓝牙连接设备后该引脚拉低，蓝牙断开设备后该引脚拉高
19	I00_3	通用 IO 口引脚	该引脚属于 I00 引脚。 注：当通讯方式选为 SPI 时只可读方向读电平，不可写。
		CLK 引脚	SPI 总线时钟信号端
20	I00_2	通用 IO 口引脚	该引脚属于 I00 引脚
		WAKEUP 引脚	BLE 唤醒引脚，低电平唤醒，高电平 BLE 模块自动睡眠
21	I00_1	通用 IO 口引脚	该引脚属于 I00 引脚
		UART_TX 引脚	UART 总线数据输出端
		MOSI 引脚	SPI 通讯方式，MCU 输出，BLE 模块输入端
22	I00_0	通用 IO 口引脚	该引脚属于 I00 引脚
		UART_RX 引脚	UART 总线数据输入端
		MISO 引脚	SPI 通讯方式，MCU 输入，BLE 模块输出端

2.3.2. 大模组引脚功能

大模组使用的是 CC2640 7*7 的芯片，大模组型号为 HY-264027, 引脚定义说明如下：

引脚序号	引脚名	功能描述	功能对应
1	I03_6	通用 IO 口引脚	该引脚属于 I03 引脚
		SEL0 通道选择引脚	MCU 与 BLE 通讯方式选择引脚。
2	I03_5	通用 IO 口引脚	该引脚属于 I03 引脚
		SEL1 通道选择引脚	MCU 与 BLE 通讯方式选择引脚。
3	I03_4	通用 IO 口引脚	该引脚属于 I03 引脚
		UART_RX 引脚	UART 总线数据输入端

		MISO 引脚	SPI 通讯方式, MCU 输入, BLE 模块输出端
4	I03_3	通用 IO 口引脚	该引脚属于 I03 引脚
		UART_TX 引脚	UART 总线数据输出端
		MOSI 引脚	SPI 通讯方式, MCU 输出, BLE 模块输入端
5	I03_2	通用 IO 口引脚	该引脚属于 I03 引脚
		WAKEUP 引脚	BLE 唤醒引脚, 低电平唤醒, 高电平 BLE 模块自动睡眠
6	I03_1	通用 IO 口引脚	该引脚属于 I03 引脚
		INT 引脚	中断输出引脚
7	I03_0	通用 IO 口引脚	该引脚属于 I03 引脚
		CLK 引脚	SPI 总线时钟信号端
8	I02_7	通用 IO 口引脚	该引脚属于 I02 引脚
		BleState 引脚	若该功能使能, 蓝牙连接设备后该引脚拉低, 蓝牙断开设备后该引脚拉高
9	TMS	烧写程序	\
10	TCK	烧写程序	\
11	I01_6	通用 IO 口引脚	该引脚属于 I01 引脚
		蓝牙开关引脚	若该功能使能, 拉低引脚则打开蓝牙广播, 拉高引脚则关闭蓝牙广播
12	I01_5	通用 IO 口引脚	该引脚属于 I01 引脚
		看门狗状态引脚	当设备正常运行时, 设备以 500ms 为周期进行 IO 口电平变化。
13	I02_6	通用 IO 口引脚	该引脚属于 I02 引脚
		PWM0 输出引脚	16 位 PWM 输出引脚, 占空比可调范围 1%-99%, 输出频率范围 1KHz-2MHz
14	I02_5	通用 IO 口引脚	该引脚属于 I02 引脚
		PWM1 输出引脚	16 位 PWM 输出引脚, 占空比可调范围 1%-99%, 输出频率范围 1KHz-2MHz
15	I02_4	通用 IO 口引脚	该引脚属于 I02 引脚
		PWM2 输出引脚	16 位 PWM 输出引脚, 占空比可调范围 1%-99%, 输出频率范围 1KHz-2MHz
16	I02_3	通用 IO 口引脚	该引脚属于 I02 引脚
		PWM3 输出引脚	16 位 PWM 输出引脚, 占空比可调范围 1%-99%, 输出频率范围 1KHz-2MHz
17	I02_2	通用 IO 口引脚	该引脚属于 I02 引脚

18	I02_1	通用 IO 口引脚	该引脚属于 I02 引脚
19	I02_0	通用 IO 口引脚	该引脚属于 I02 引脚
20	I01_7	通用 IO 口引脚	该引脚属于 I01 引脚
21	I01_4	通用 IO 口引脚	该引脚属于 I01 引脚
22	I01_3	通用 IO 口引脚	该引脚属于 I01 引脚
23	I01_2	通用 IO 口引脚	该引脚属于 I01 引脚
24	I01_1	通用 IO 口引脚	该引脚属于 I01 引脚
25	I01_0	通用 IO 口引脚	该引脚属于 I01 引脚
26	VDD	电源	BLE 模块供电引脚，电压范围 2.0-3.6V
27	RESET	复位引脚	BLE 硬件复位引脚，低电平复位
28	I00_7	通用 IO 口引脚	该引脚属于 I01 引脚
		SBL_EN 使能引脚	串口升级使能引脚
29	I00_6	通用 IO 口引脚	该引脚属于 I01 引脚
		ADC1 引脚	12 位 ADC，通道 1
30	I00_5	通用 IO 口引脚	该引脚属于 I01 引脚
		ADC0 引脚	12 位 ADC，通道 0
31	I00_4	通用 IO 口引脚	该引脚属于 I01 引脚
32	I00_3	通用 IO 口引脚	该引脚属于 I01 引脚
		OAD_CSN 片选引脚	OAD 升级时，Flash 片选引脚
33	I00_2	通用 IO 口引脚	该引脚属于 I01 引脚
		OAD_CLK 引脚	OAD 升级时，Flash 时钟信号端
34	I00_1	通用 IO 口引脚	该引脚属于 I01 引脚
		OAD_MOSI 引脚	OAD 升级，BLE 模块输出端，Flash 输入端
35	I00_0	通用 IO 口引脚	该引脚属于 I01 引脚
		OAD_MISO 引脚	OAD 升级，BLE 模块输入端，Flash 输出端
36	GND	地	BLE 模块接地引脚

2.4. 模组电气特性

- 调制模式：GFSK
- 接收灵敏度：-96dBm（典型值）
- 频率范围：2402~2480MHZ（2.4G ISM band）
- 输出功率设置：-20~+5 dBm（可通过软件编程设定）
- 工作温度：-30℃~+80℃
- 储存温度：-40℃~+100℃

- 工作湿度：< 85%RH (at 40°C)
- 电源电压：2.0V-3.6V DC(最大消耗电流时需能保持住)。
- 接收模式瞬间最大电流：5.9 mA max
- 发射模式瞬间最大电流（设定 0dBm）：6.1 mA max
- 发射模式瞬间最大电流（设定+5dBm）：9.1 mA max
- 待机功耗：1 μ A
- 深度睡眠功耗：0.13 μ A
- 有效接收距离：以模块与 iPhone 6S 手机面对面自由空间测试
60 米（发射功率设定 0 dBm）
80 米（发射功率设定+5 dBm）

2.5. 模组蓝牙功耗对照表

蓝牙状态	设置参数	实际间隔时间 (ms)	平均电流值
广播间隔	32	20	768uA
	160	100	201uA
	800	500	41.3uA
	1600	1000	21.3uA
连接间隔	16	20	461uA
	80	100	95.7uA
	160	200	51.3uA
	400	500	25.6uA
	800	1000	16.9uA

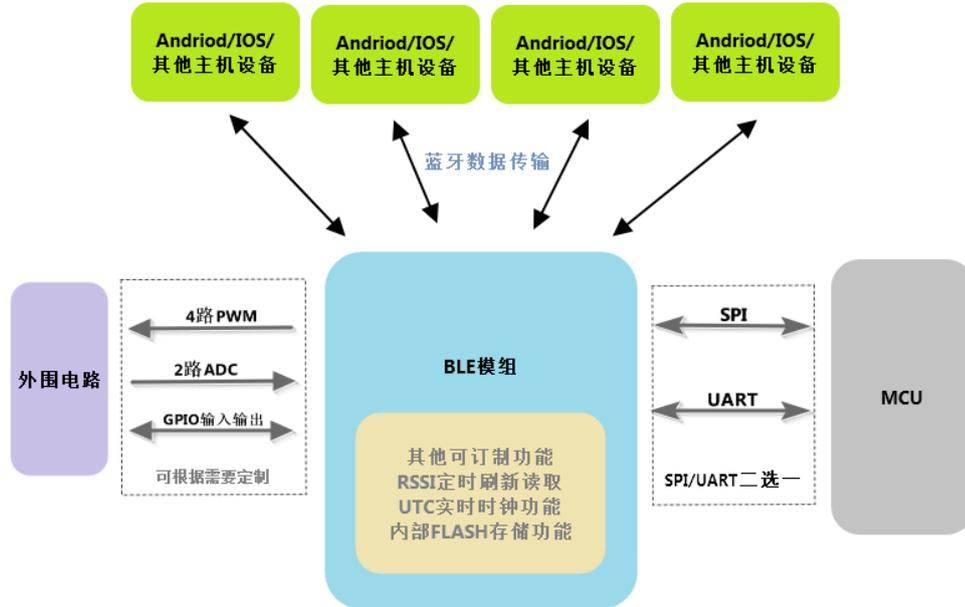
注：以上测试内容均在 WAKEUP 引脚拉高的情况下测试。

2.6. 工作模式

分为直驱模式以及桥接模式，总体示意图如下：

蓝牙工作状态：从机模式

支持同时连接3个主机，并进行数据收发



2.7. 电路连接

2.7.1. 模块通信接口选择

工作在桥接模式时，模组与外部 MCU 进行通信，所支持的通信接均为两种：UART/SPI (可任选一种通信方式)。

选择通信接口有三种方法：硬件设置、AT 指令设置、出厂预先配置。

2.7.1.1. 硬件设置

透传程序，模组默认需要通过 SEL 引脚确定通信方式。硬件设计时，需通过设定 SEL 引脚的高低电平，选择通讯方式，见下表。

SEL 引脚选择通讯模式（仅限大模组）

序号	通道选择 PIN 状态		通讯接口状态		备注
	SEL1	SEL0	UART	SPI	
1	0	0	OK	X	硬件连接示意图， 参见 2.7.2 节
2	0	1	OK	X	
3	1	0	X	OK	
4	1	1	OK	X	
5	X	X	OK	X	

备注：

0 表示低电平，1 表示高电平，X 表示悬空

2.7.1.2. AT 指令设置

通过 AT 指令修改通信接口，参见 AT 指令说明。假设已经使用硬件设置的方法选择了某种通信接口，也可再次通过 AT 指令修改通信方式。

2.7.1.3. 出厂预先配置

倘若硬件设计时，为了更优化硬件资源的使用，欲将 SEL 引脚作为 GPIO，可与我司业务取得联系，在模组出厂时预先配置好通信接口。

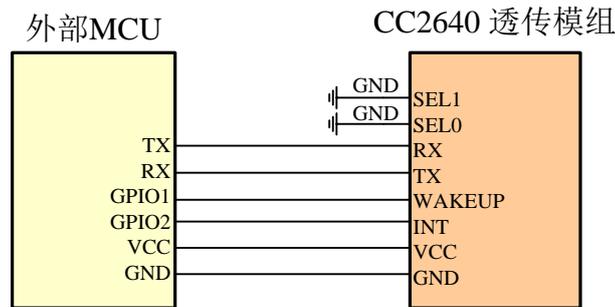
2.7.2. 两种通信方式连接示意图

透传模组与外部 MCU 通信时，需严格按照通信时序操作。当然，在测试时，可将透传模组 WAKEUP 引脚拉低，使模组一直处于唤醒状态，便于测试。

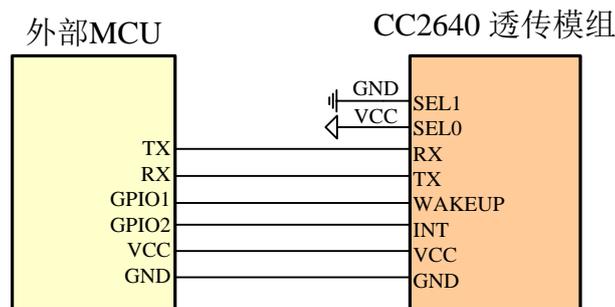
2.7.2.1. UART 连接示意图

UART 方式，SEL0/SEL1 连接方式有以下四种：

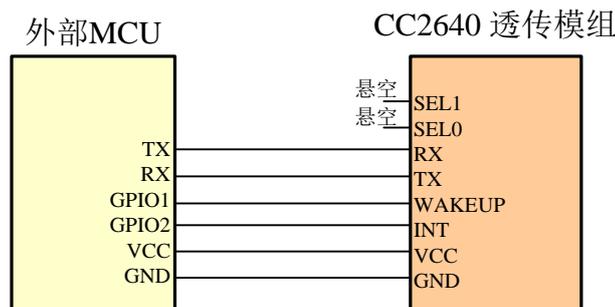
- (1) SEL1=0, SEL0=0 (**推荐**)



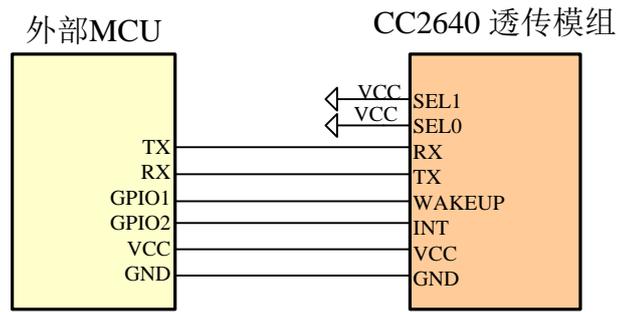
- (2) SEL1=0, SEL0=1



- (3) SEL0 及 SEL1 均悬空

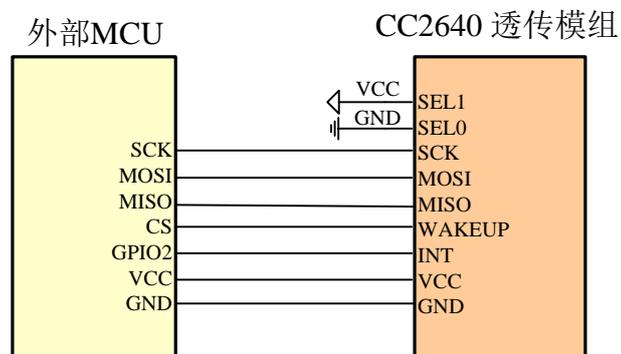


(4) SEL0 及 SEL1 均为高



2.7.2.2.SPI 连接示意图

SPI 方式，SEL0/SEL1 连接方式，如 SEL1=1, SEL0=0:



3. 软件规格

3.1. 命令速查表

序号	寄存器地址	AT 指令 <small>注：所有 AT 指令结尾都需要带上“\r\n”表格中不再重复写</small>	掉电保存	SPI/UART 读写属性	APP 读写属性	备注
1.	0x02	AT+MOD=n	NO	R/W	NO	数据模式选择
2.	0x06	/	NO	R	NO	查看当前命令状态
3.	0x07	AT+SYS_STATE=?	NO	R	NO	系统状态
4.	0x0B	/	NO	R	NO	模块接收到的蓝牙数据长度 (SPI)
5.	0x0D	AT+BAUD=n	YES	R/W	R/W	UART 波特率设置
6.	0x0E	AT+NAME=xxx	YES	R/W	R/W	BLE 模块名称设置
7.	0x0F	AT+PSWD=xxx	YES	R/W	R/W	蓝牙连接密码设置 (未开放)
8.	0x11	AT+DEVID=?	YES	R	R	本机 MAC 地址读取
9.	0x50	AT+DEV0=?	NO	R	R	读取已连入的设备 0 MAC 地址
10.	0x51	AT+DEV1=?	NO	R	R	读取已连入的设备 1 MAC 地址
11.	0x52	AT+DEV2=?	NO	R	R	读取已连入的设备 2 MAC 地址
12.	0x53	AT+DEV3=?	NO	R	R	读取已连入的设备 3 MAC 地址
13.	0x54	AT+DEV0_ROLE=?	NO	R	R	读取模块与设备 0 连接的角色
14.	0x55	AT+DEV1_ROLE=?	NO	R	R	读取模块与设备 1 连接的角色
15.	0x56	AT+DEV2_ROLE=?	NO	R	R	读取模块与设备 2 连接的角色

16.	0x57	AT+DEV3_ROLE=?	NO	R	R	读取模块与设备 3 连接的角色
17.	0x1D	AT+FUNSTATE0=?	NO	R	R	状态 0 寄存器
18.	0x1E	AT+FUNSTATE1=?	NO	R/W	R/W	状态 1 寄存器
19.	0x24	AT+TX=n	YES	R/W	R/W	发射功率寄存器
20.	0x26	AT+ADV_INTERVAL=n	YES	R/W	R/W	广播间隔
21.	0x27	AT+CONN_INTERVAL=n	YES	R/W	R/W	连接间隔
22.	0x29	AT+ADV=n	NO	R/W	R/W	广播开关
23.	0x1C	AT+DISCON=n	NO	W	NO	断开连接
24.	0x2A	AT+SOFT_RST=1	NO	W	NO	软件复位
25.	0x3D	AT+VERION=?	NO	R	R	软件版本号
26.	0x3E	/	NO	W	NO	清空蓝牙数据缓存 (SPI)
27.	0x42	AT+SLAVE_LATENCY=n	YES	R/W	R/W	蓝牙从机延迟
28.	0x43	AT+CONN_TIMEOUT=n	YES	R/W	R/W	连接超时
29.	0x44	AT+PSD_EN=n	YES	R/W	R/W	密码使能 (未开放)
30.	0x47	AT+ADV_MFR_SPC=xxx	YES	R/W	R/W	广播数据
31.	0x58	AT+RSSI_READ0=?	NO	R	NO	读取设备 0 RSSI
32.	0x59	AT+RSSI_READ1=?	NO	R	NO	读取设备 1 RSSI
33.	0x5A	AT+RSSI_READ2=?	NO	R	NO	读取设备 2 RSSI
34.	0x5B	AT+RSSI_READ3=?	NO	R	NO	读取设备 3 RSSI
35.	0x12	AT+RSSI_PERIOD=n	YES	R/W	R/W	RSSI 获取周期设置
36.	0x2C	AT+IO0=n	NO	R/W	R/W	IO0 读写

37.	0x2D	AT+IO1=n	NO	R/W	R/W	IO1 读写
38.	0x2E	AT+IO2=n	NO	R/W	R/W	IO2 读写
39.	0x2F	AT+IO3=n	NO	R/W	R/W	IO3 读写
40.	0x30	AT+DIR0=n	NO	R/W	R/W	IO0 方向
41.	0x31	AT+DIR1=n	NO	R/W	R/W	IO1 方向
42.	0x32	AT+DIR2=n	NO	R/W	R/W	IO2 方向
43.	0x33	AT+DIR3=n	NO	R/W	R/W	IO3 方向
44.	0x34	AT+PWM0_PER=n	YES	R/W	R/W	PWM0 频率设置
45.	0x35	AT+PWM1_PER=n	YES	R/W	R/W	PWM1 频率设置
46.	0x36	AT+PWM2_PER=n	YES	R/W	R/W	PWM2 频率设置
47.	0x37	AT+PWM3_PER=n	YES	R/W	R/W	PWM3 频率设置
48.	0x4A	AT+PWM0_DUTY =n	YES	R/W	R/W	PWM0 占空比
49.	0x4B	AT+PWM1_DUTY =n	YES	R/W	R/W	PWM1 占空比
50.	0x4C	AT+PWM2_DUTY =n	YES	R/W	R/W	PWM2 占空比
51.	0x4D	AT+PWM3_DUTY =n	YES	R/W	R/W	PWM3 占空比
52.	0x2B	AT+UTC_TIME=xx	NO	R/W	R/W	实时时钟
53.	0x1F	AT+ADCCFG=n	NO	R/W	R/W	ADC 配置寄存器
54.	0x20	AT+ADCVAL=?	NO	R	R	获取 ADC 值
55.	0x41	AT+BAT_LEVEL=?	NO	R	R	电池电量百分比
56.	0x21	AT+NV_ADDR=n	NO	R/W	R/W	非易失存储器地址
57.	0x22	AT+NV_DAT_LEN= n	NO	R/W	R/W	非易失存储器数据长度
58.	0x23	AT+NV_DATA=xxx	YES	R/W	R/W	非易失存储器数据
59.	0x5C	AT+D0=xxx	NO	R/W	NO	读写设备 0 蓝牙数据
60.	0x5D	AT+D1=xxx	NO	R/W	NO	读写设备 1 蓝牙数据

61.	0x5E	AT+D2=xxx	NO	R/W	NO	读写设备 2 蓝牙数据
62.	0x5F	AT+D3=xxx	NO	R/W	NO	读写设备 3 蓝牙数据
63.	\	AT+DATA=xxx	NO	W	NO	UART 向所有设备发送透传数据
64.	\	\	\	\	\	UART 接收透传数据 说明

表 2-1

3.2. 指令格式说明

3.2.1. 模块通信接口选择

工作在桥接模式时，模组与外部 MCU 进行通信，所支持的通信接均为两种：UART/SPI (可任选一种通信方式)。

选择通信接口有三种方法：硬件设置、AT 指令设置、出厂预先配置。

3.2.1.1. 硬件设置

透传程序，模组默认需要通过 SEL 引脚确定通信方式。硬件设计时，需通过设定 SEL 引脚的高低电平，选择通讯方式，见下表。

SEL 引脚选择通讯模式（仅限大模组）

序号	通道选择 PIN 状态		通讯接口状态		备注
	SEL1	SEL0	UART	SPI	
1	0	0	OK	X	硬件连接示意图， 参见 2.7.2 节
2	0	1	OK	X	
3	1	0	X	OK	
4	1	1	OK	X	
5	X	X	OK	X	

备注：

0 表示低电平，1 表示高电平，X 表示悬空

3.2.1.2. AT 指令设置

通过 AT 指令修改通信接口，参见 AT 指令说明。假设已经使用硬件设置的方法选择了某种通信接口，也可再次通过 AT 指令修改通信方式。

3.2.1.3. 出厂预先配置

倘若硬件设计时，为了更优化硬件资源的使用，欲将 SEL 引脚作为 GPIO，可与我司业务取得联系，在模组出厂时预先配置好通信接口。

3.2.2. UART 命令格式说明

1. UART 发送 AT 指令(具体指令参照表 2-1)，若指令正确则会返回 AT+OK\r\n，若错误则会返回错误状态。
2. UART 的读取及写入由寄存器的读写属性决定。
3. 查询具有读属性的寄存器在对应的 AT 指令后加“?”即可读取到相应的内容。

注意：所有 AT 指令都以“\r\n”结尾。

3.2.2.1. UART 读写命令

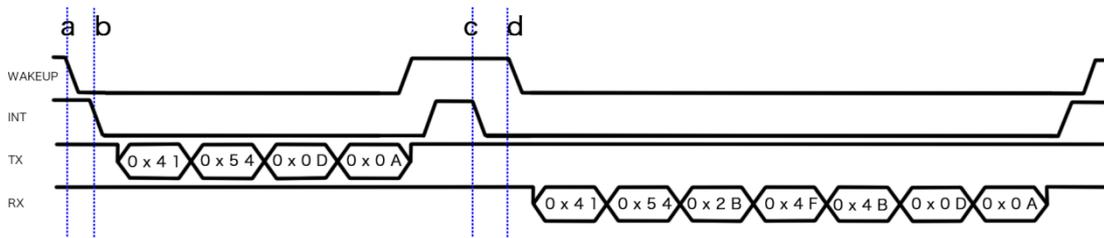
示例：

UART 写：AT+NV_ADDR=1\r\n

模块回传数据：AT+OK\r\n

UART 读：AT+NV_ADDR=?\r\n

模块回传数据: AT+OK\r\n\r\n

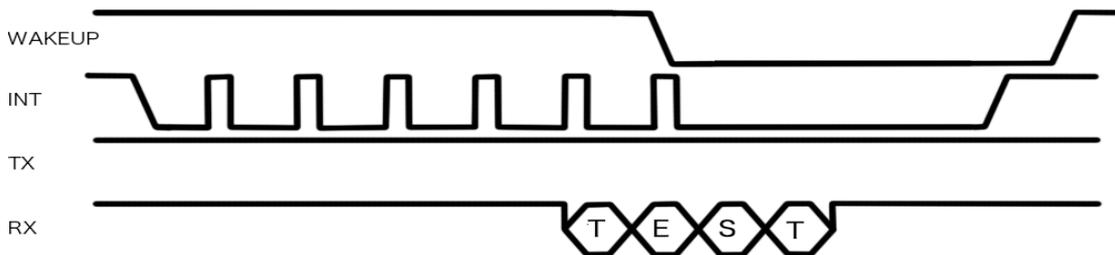


$\Delta a-b \geq 600 \text{ us}$

$\Delta c-d \leq 20 \text{ us}$

3.2.2.2. UART 接收 BLE 数据

1. 当 WAKEUP 引脚处于低电平状态时，模块收到数据后会直接发送 UART 数据到 MCU。
2. 当 WAKEUP 引脚处于高电平状态时，模块收到数据后会反复拉高拉低 INT 引脚，直到 MCU 拉低 WAKEUP 引脚为止，当 MCU 拉低 WAKEUP 引脚后会发送 UART 数据到 MCU。



3.2.3. SPI 命令格式说明

- SPI 协议采用消息机制来反馈数据，数据读在中断信号线“INT”为低电平时有效。
- SPI 协议采用一问一答方式交互数据，数据的读取及写入由寄存器的读写属性决定。
- SPI 数据由数据头、数据长度、寄存器地址、数据、校验五个部分组成。

数据头： 固定为 0xFE。

寄存器地址： 参考表 2-1

数据长度： 寄存器地址（1 字节）+ 数据内容长度（n 字节）

校验： 数据长度 ^ 寄存器地址 ^ 数据 1 ^ 数据 2 ^ ... 数据 n

SPI 数据格式：

字节 1: 0xFE（数据头）

字节 2: 数据长度

字节 3: 寄存器地址

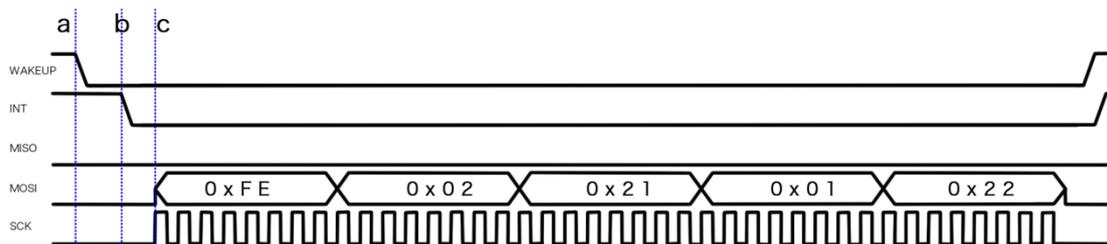
字节 4: 数据内容

...

字节 n: 数据内容

字节 n + 1: 校验

3.2.3.1. SPI 写命令



$$\Delta a-b \geq 600 \text{ us}$$

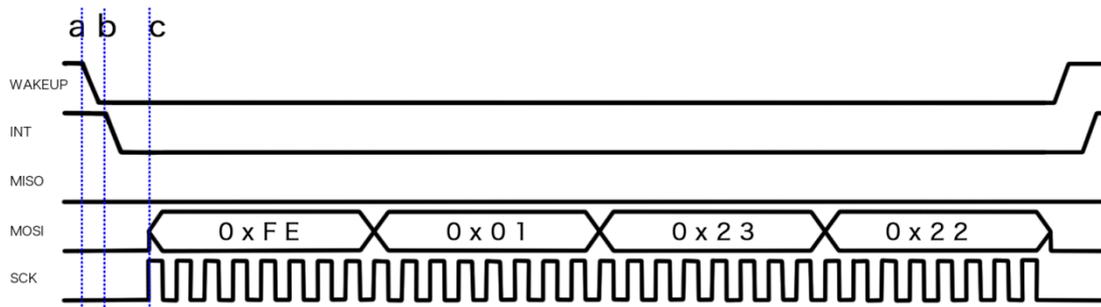
$$\Delta a-c > \Delta b-c$$

示例:

写入	数据头	数据长度	命令	1'字节	校验
	0xFE	0x02	0x21	0x01	0x22

3.2.3.2. SPI 读命令

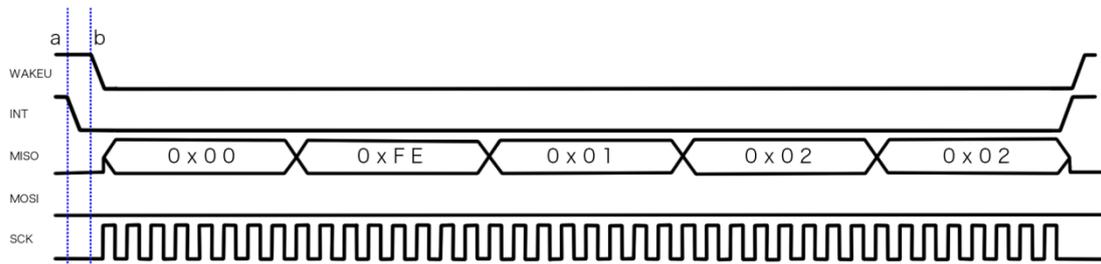
SPI 读请求



$$a-b \geq 600 \text{ us}$$

$$\Delta a-c > \Delta b-c$$

SPI 读数据



$$\Delta a-b \geq 600 \text{ us}$$

示例:

写入	数据头	数据长度	命令	校验
	0xFE	0x01	0x23	0x22

读取	空读	数据头	数据长度	命令	1'字节	...	N'字节	校验
	0X00	0XFE	N+1	0x23	n	...	n	XX

3.3. 指令集说明

注意：UART 模式：数据均为 ASCII 格式

SPI 模式：以下数据为发送 SPI 指令的数据内容，数据均为 HEX 格式。

(SPI 数据发送格式请参照 2.2 SPI 命令格式说明)

3.3.1. 数据模式选择 (0x02)

字符模式：模组会将收到的 UART 数据直接发送到蓝牙接收端。

十六进制模式：模组会将收到的 UART 数据转以 HEX 格式发送到蓝牙接收端。

默认为字符模式。

AT 指令：

字符模式：AT+MOD=0\r\n

十六进制模式：AT+MOD=1\r\n

读取数据模式：AT+MOD=?\r\n

SPI 数据：

字符模式： 0x00

十六进制模式：0x01

读取数据模式：不填入数据

3.3.2. 查看当前命令状态 (0x06)

AT 指令：

无

SPI 数据：

读取当前命令状态：不填入数据

SPI 数据读取返回内容，数据内容长度为 1，数据内容对应如表 2-2。

状态字节 (HEX)	描述
0X00	无回应
0X01	回应 OK
0X02	无此命令
0X03	设备未就绪，或者缓存溢出，请稍后操作
0X04	空间不够，或者正在执行当前任务，或者未就绪
0X05	扫描错误
0X06	设备处于连接状态或者未初始化
0X07	设备处于为连接状态或者未初始化，或者设备索引超出
0X08	设备处于为连接状态或者未初始化，或者当前处于未连接状态
0X09	索引超出
0X0A	参数溢出

0X0B	保存失败
0X0C	参数错误

表 2-2

3.3.3. 查看系统状态 (0x07)

AT 指令:

读取系统状态: AT+SYS_STATE=?\r\n

SPI 数据:

读取系统状态: 不填入数据

数据内容长度为 1, 数据内容的低四位对应连接当前设备的连接数量。

数据内容的高 4 位对应 表 2-3

状态字节 (HEX)	描述	
	主机模式 (暂不支持)	从机模式
0X00	主机空闲	初始化中
0X10	主机正在连接	已启动从模式, 但没有广播
0X20	主机已连接	正在广播
0X30	主机未连接	模组已到达连接最大数, 不再广播
0X40	NC	无效状态

表 2-3

3.3.4. 模块接收到的蓝牙数据长度 (SPI) (0x0B)

AT 指令:

无

SPI 数据:

读取蓝牙数据长度: 不填入数据

数据内容长度为 4。

数据内容字节 1~字节 4 分别表示接收到设备 0~设备 3 的蓝牙数据长度。

3.3.5. UART 波特率设置 (0x0D)

AT 指令:

设置波特率: AT+BAUD=n\r\n (参数参照表 2-4)

读取波特率: AT+BAUD=?\r\n

SPI 数据:

设置波特率: n (参数参照表 2-4)

读取波特率: 不填入数据

数据参数	波特率	数据参数	波特率
0	9600 bps	10	110 bps
1	19200 bps	11	150 bps
2	38400 bps	12	300 bps
3	57600 bps	13	600 bps
4	115200 bps	14	1200 bps

5	14400 bps	15	1800 bps
6	128000 bps	16	2400 bps
7	256000 bps	17	4800 bps
8	375000 bps	18	7200 bps
9	75 bps	/	/

表 2-4

3.3.6. BLE 模块名称设置 (0x0E)

AT 指令:

设置模块名称: AT+NAME=xxxxxxx\r\n

读取模块名称: AT+NAME=?\r\n

SPI 数据:

设置模块名称: 以 HEX 形式写入模块名称 (ASCII 形式数据)

读取模块名称: 不填入数据

注意: 长度不能超过 18 个字节

3.3.7. 本机 MAC 地址读取 (0x11)

AT 指令:

读取本机 MAC 地址: AT+DEVID=?\r\n

SPI 数据:

读取本机 MAC 地址: 不填入数据

3.3.8. 读取已连入的设备 0 MAC 地址 (0x50)

AT 指令:

读取已连接的设备 0 MAC 地址: AT+DEV0=?\r\n

SPI 数据:

读取已连接的设备 0 MAC 地址: 不填入数据

3.3.9. 读取已连入的设备 1 MAC 地址 (0x51)

AT 指令:

读取已连接的设备 1 MAC 地址: AT+DEV1=?\r\n

SPI 数据:

读取已连接的设备 1 MAC 地址: 不填入数据

3.3.10. 读取已连入的设备 2 MAC 地址 (0x52)

AT 指令:

读取已连接的设备 2 MAC 地址: AT+DEV2=?\r\n

SPI 数据:

读取已连接的设备 2 MAC 地址: 不填入数据

3.3.11. 读取已连入的设备 3 MAC 地址 (0x53)

AT 指令:

读取已连接的设备 3 MAC 地址: AT+DEV3=?\r\n

SPI 数据:

读取已连接的设备 3 MAC 地址: 不填入数据

3.3.12. 读取模块与设备 0 连接的角色 (0x54)

AT 指令:

读取与已连接的设备 0 连接的角色: AT+DEV0_ROLE=?\r\n

SPI 数据:

读取与已连接的设备 0 连接的角色: 不填入数据

读取的数据参照表 2-5

3.3.13. 读取模块与设备 1 连接的角色 (0x55)

AT 指令:

读取与已连接的设备 1 连接的角色: AT+DEV1_ROLE=?\r\n

SPI 数据:

读取与已连接的设备 1 连接的角色: 不填入数据

读取的数据参照表 2-5

3.3.14. 读取模块与设备 2 连接的角色 (0x56)

AT 指令:

读取与已连接的设备 2 连接的角色: AT+DEV2_ROLE=?\r\n

SPI 数据:

读取与已连接的设备 2 连接的角色: 不填入数据

读取的数据参照表 2-5

3.3.15. 读取模块与设备 3 连接的角色 (0x57)

AT 指令:

读取与已连接的设备 3 连接的角色: AT+DEV3_ROLE=?\r\n

SPI 数据:

读取与已连接的设备 3 连接的角色: 不填入数据

读取的数据参照表 2-5

数据参数	波特率
1	广播角色
2	观察角色
4	从机角色
8	主机角色

表 2-5

3.3.16. 状态 0 寄存器 (0x1D)

AT 指令:

读取状态 0 寄存器: AT+FUNSTATE0=?\r\n

SPI 数据:

读取状态 0 寄存器: 不填入数据

返回数据长度为 1，数据对应内容请参照表 2-6

位	名称	初始值	R/W	描述
3~7	NC	0	R/W	保留
2	scanRdy	0	R	扫描忙标志 1：正在扫描，0：扫描结束（暂不支持）
1	NC	0	R	保留
0	adcRdy	0	R	ADC 转换完成标志 1：就绪 0：正在转换

表 2-6

3.3.17. 状态 1 寄存器 (0x1E)

AT 指令：

设置状态 1 寄存器：AT+FUNSTATE1=n\r\n （参数参照表 2-6）

读取状态 1 寄存器：AT+FUNSTATE1=?\r\n

SPI 数据：

设置状态 1 寄存器：n （参数参照表 2-7）

读取状态 1 寄存器：不填入数据

位	名称	初始值	R/W	描述
7	NC	0	R/W	保留
6	speakerEn	0	R/W	蜂鸣器使能，1：开蜂鸣器 0：关蜂鸣器
5	pwm5En	0	R/W	PWM5 使能，1：开 PWM5 0：关 PWM5
4	pwm4En	0	R/W	PWM4 使能，1：开 PWM4 0：关 PWM4
3	pwm3En	0	R/W	PWM3 使能，1：开 PWM3 0：关 PWM3
2	pwm2En	0	R/W	PWM2 使能，1：开 PWM2 0：关 PWM2
1	pwm1En	0	R/W	PWM1 使能，1：开 PWM1 0：关 PWM1
0	pwm0En	0	R/W	PWM0 使能，1：开 PWM0 0：关 PWM0

表 2-7

3.3.18. 发射功率寄存器 (0x24)

AT 指令：

设置发射功率：AT+TX=n\r\n （参数参照表 2-8）

读取发射功率：AT+TX=?\r\n

SPI 数据：

设置发射功率：n （参数参照表 2-8）

读取发射功率：不填入数据

数据参数	TX 功率
0	-21dbm

1	-18dbm
2	-15dbm
3	-12dbm
4	-9dbm
5	-6dbm
6	-3dbm
7	0dbm
8	1dbm
9	2dbm
10	3dbm
11	4dbm
12	5dbm

表 2-8

3.3.19. 广播间隔 (0x26)

AT 指令:

设置广播间隔: AT+ADV_INTERVAL=n\r\n

读取广播间隔: AT+ADV_INTERVAL=?\r\n

SPI 数据:

设置广播间隔: 字节 1: 填入数据低位, 字节 2: 填入数据高位。

读取广播间隔: 不填入数据

注: n 取值范围为: 32-24000, 每单位/625us, 实际对应时间为 20ms-15000ms (15s), 以大端模式发送。

3.3.20. 连接间隔 (0x27)

AT 指令:

设置连接间隔: AT+CON_INTERVAL=n\r\n

读取连接间隔: AT+CON_INTERVAL=?\r\n

SPI 数据:

设置连接间隔: 字节 1: 填入数据低位, 字节 2: 填入数据高位。

读取连接间隔: 不填入数据

注: n 取值范围为: 16-3200, 每单位/1.25ms, 实际对应时间为 20ms-4000ms (4s), 以大端模式发送。

IOS 设置时, 此项参数不得超过 1578, 最小不小于 17 (对应 20-2s) 否则 IOS 系统采用默认参数,

(需要开启 BLE_TRANS_SPEED), 此参数需在设置角色之前设置, 下次开机依然有效。

3.3.21. 广播开关 (0x29)

AT 指令:

设置广播开关: AT+ADV=n\r\n (参数参照表 2-9)

读取广播开关: AT+ADV=?\r\n

SPI 数据:

设置广播开关: n (参数参照表 2-9)

读取广播开关：不填入数据

数据参数	传输速度
0	关闭广播
1	开启广播

表 2-9

3.3.22. 断开连接 (0x1C)

AT 指令：

设置断开当前连接：AT+DISCON=n\r\n

SPI 数据：

设置断开当前连接：n

注：n 取值范围为：0-3，分别代表了已连入设备的索引号。

3.3.23. 软件复位 (0x2A)

AT 指令：

设置软件复位：AT+SOFT_RST=1\r\n

SPI 数据：

设置软件复位：0x01

3.3.24. 软件版本号 (0x3D)

AT 指令：

读取软件版本号：AT+VERION=?\r\n

SPI 数据：

读取软件版本号：不填入数据

3.3.25. 清空蓝牙数据缓存 (SPI) (0x3E)

AT 指令：

无

SPI 数据：

清空蓝牙数据缓存：n (填入任意值)

3.3.26. 蓝牙从机延迟 (0x42)

AT 指令：

设置从机延迟：AT+SLAVE_LATENCY=n\r\n

读取从机延迟：AT+SLAVE_LATENCY=?\r\n

SPI 数据：

设置从机延迟：字节 1：填入数据低位， 字节 2：填入数据高位

读取从机延迟：不填入数据

注：从机延迟取值范围是 0-499 个单位

3.3.27. 连接超时 (0x43)

AT 指令：

设置连接超时时间：AT+CONN_TIMEOUT=n\r\n

读取连接超时时间：AT+CONN_TIMEOUT=?\r\n

SPI 数据：

设置连接超时时间：字节 1：填入数据低位， 字节 2：填入数据高位

读取连接超时时间：不填入数据

注：连接超时取值范围是 10-1000 个单位，每个单位为 10ms

3.3.28. 广播数据 (0x47)

AT 指令：

设置广播数据：AT+ADV_MFR_SPC=xxxx\r\n

读取广播数据：AT+ADV_MFR_SPC=?\r\n

SPI 数据：

设置广播数据：xxx

读取广播数据：不填入数据

注：广播数据长度最长不超过 22 个字节。

3.3.29. 读取设备 0 RSSI (0x58)

AT 指令：

读取 RSSI 值：AT+RSSI_READ0=?\r\n

SPI 数据：

读取 RSSI 值：不填入数据

注：该功能为订制功能，用户可根据需要订制。

3.3.30. 读取设备 1 RSSI (0x59)

AT 指令：

读取 RSSI 值：AT+RSSI_READ1=?\r\n

SPI 数据：

读取 RSSI 值：不填入数据

注：该功能为订制功能，用户可根据需要订制。

3.3.31. 读取设备 2 RSSI (0x5A)

AT 指令：

读取 RSSI 值：AT+RSSI_READ2=?\r\n

SPI 数据：

读取 RSSI 值：不填入数据

注：该功能为订制功能，用户可根据需要订制。

3.3.32. 读取设备 3 RSSI (0x5B)

AT 指令：

读取 RSSI 值：AT+RSSI_READ3=?\r\n

SPI 数据：

读取 RSSI 值：不填入数据

注：该功能为订制功能，用户可根据需要订制。

3.3.33. RSSI 获取周期设置 (0x12)

AT 指令:

设置 RSSI 获取周期: AT+RSSI_PERIOD=n\r\n

读取 RSSI 获取周期: AT+RSSI_PERIOD=?\r\n

SPI 数据:

设置 RSSI 获取周期: 字节 1: 填入数据低位, 字节 2: 填入数据高位。

读取 RSSI 获取周期: 不填入数据

注: 1. 周期值为十进制数据, 取值范围为 20ms-5000ms, 以大端模式发送数据。

该功能为订制功能, 用户可根据需要订制。

3.3.34. IO0 读写 (0x2C)

AT 指令:

设置 IO0 输出电平: AT+IO0=n\r\n (以 HEX 格式发送)

读取 IO0 电平: AT+IO0=?\r\n

SPI 数据:

设置 IO0 输出电平: n

读取 IO0 电平: 不填入数据

注: IO0 读写寄存器可对 IO0_0-IO0_7 进行读操作和写操作。

数据以 HEX 格式发送, 并且数据的每一位对应一个 IO 口。数据的最低位对应 IO0_0, 依次往上推。

往寄存器对应位写 1 表示输出高电平, 0 表示输出低电平。

读寄存器对应位为 1 表示当前为高电平, 0 表示当前为低电平。

该功能为订制功能, 用户可根据需要订制。

3.3.35. IO1 读写 (0x2D)

AT 指令:

设置 IO1 输出电平: AT+IO1=n\r\n (以 HEX 格式发送)

读取 IO1 电平: AT+IO1=?\r\n

SPI 数据:

设置 IO1 输出电平: n

读取 IO1 电平: 不填入数据

注: IO1 读写寄存器可对 IO1_0-IO1_7 进行读操作和写操作。

数据以 HEX 格式发送, 并且数据的每一位对应一个 IO 口。数据的最低位对应 IO1_0, 依次往上推。

往寄存器对应位写 1 表示输出高电平, 0 表示输出低电平。

读寄存器对应位为 1 表示当前为高电平, 0 表示当前为低电平。

该功能为订制功能, 用户可根据需要订制。

3.3.36. IO2 读写 (0x2E)

AT 指令:

设置 IO2 输出电平: AT+IO2=n\r\n (以 HEX 格式发送)

读取 IO2 电平: AT+IO2=?\r\n

SPI 数据:

设置 I02 输出电平：n
读取 I02 电平：不填入数据

注：I02 读写寄存器可对 I02_0-I02_2 进行读操作和写操作。

数据以 HEX 格式发送，并且数据的每一位对应一个 IO 口。数据的最低位对应 I02_0，依次往上推。

往寄存器对应位写 1 表示输出高电平，0 表示输出低电平。

读寄存器对应位为 1 表示当前为高电平，0 表示当前为低电平。

该功能为订制功能，用户可根据需要订制。

3.3.37. I03 读写 (0x2F)

AT 指令：

设置 I03 输出电平：AT+I03=n\r\n (以 HEX 格式发送)

读取 I03 电平：AT+I03=?\r\n

SPI 数据：

设置 I03 输出电平：n

读取 I03 电平：不填入数据

注：I03 读写寄存器可对 I03_0-I03_1 进行读操作和写操作。

数据以 HEX 格式发送，并且数据的每一位对应一个 IO 口。数据的最低位对应 I03_0，依次往上推。

往寄存器对应位写 1 表示输出高电平，0 表示输出低电平。

读寄存器对应位为 1 表示当前为高电平，0 表示当前为低电平。

该功能为订制功能，用户可根据需要订制。

3.3.38. I00 方向 (0x30)

AT 指令：

设置 I00 输入输出方向：AT+DIR0=n\r\n (以 HEX 格式发送)

读取 I00 方向：AT+DIR0=?\r\n

SPI 数据：

设置 I00 输入输出方向：n

读取 I00 方向：不填入数据

注：I00 方向寄存器可对 I00_0-I00_7 输入输出方向进行设置和读取。

数据以 HEX 格式发送，并且数据的每一位对应一个 IO 口。数据的最低位对应 I00_0，依次往上推。

往寄存器对应位写 1 表示输出，0 表示输入。

读寄存器对应位为 1 表示输出，0 表示输入。

该功能为订制功能，用户可根据需要订制。

3.3.39. I01 方向 (0x31)

AT 指令：

设置 I01 输入输出方向：AT+DIR1=n\r\n (以 HEX 格式发送)

读取 I01 方向：AT+DIR1=?\r\n

SPI 数据：

设置 I01 输入输出方向：n

读取 I01 方向：不填入数据

注：I01 方向寄存器可对 I01_0-I01_7 输入输出方向进行设置和读取。

数据以 HEX 格式发送，并且数据的每一位对应一个 IO 口。数据的最低位对应 I01_0，依次往上推。

往寄存器对应位写 1 表示输出，0 表示输入。

读寄存器对应位为 1 表示输出，0 表示输入。

该功能为订制功能，用户可根据需要订制。

3.3.40. IO2 方向 (0x32)

AT 指令:

设置 IO2 输入输出方向: AT+DIR2=n\r\n (以 HEX 格式发送)

读取 IO2 方向: AT+DIR2=?\r\n

SPI 数据:

设置 IO2 输入输出方向: n

读取 IO2 方向: 不填入数据

注: IO2 方向寄存器可对 IO2_0-IO2_2 输入输出方向进行设置和读取。

数据以 HEX 格式发送, 并且数据的每一位对应一个 IO 口。数据的最低位对应 IO2_0, 依次往上推。

往寄存器对应位写 1 表示输出, 0 表示输入。

读寄存器对应位为 1 表示输出, 0 表示输入。

该功能为订制功能, 用户可根据需要订制。

3.3.41. IO3 方向 (0x33)

AT 指令:

设置 IO3 输入输出方向: AT+DIR3=n\r\n (以 HEX 格式发送)

读取 IO3 方向: AT+DIR3=?\r\n

SPI 数据:

设置 IO3 输入输出方向: n

读取 IO3 方向: 不填入数据

注: IO3 方向寄存器可对 IO3_0-IO3_1 输入输出方向进行设置和读取。

数据以 HEX 格式发送, 并且数据的每一位对应一个 IO 口。数据的最低位对应 IO3_0, 依次往上推。

往寄存器对应位写 1 表示输出, 0 表示输入。

读寄存器对应位为 1 表示输出, 0 表示输入。

该功能为订制功能, 用户可根据需要订制。

3.3.42. PWM0 频率设置 (0x34)

AT 指令:

设置 PWM0 频率: AT+PWM0_PER=n\r\n (以十进制格式发送)

读取 PWM0 频率: AT+PWM0_PER=?\r\n

SPI 数据:

设置 PWM0 频率: n

读取 PWM0 频率: 不填入数据

注: PWM 频率设置范围为 1KHz-2MHz

该功能为订制功能, 用户可根据需要订制。

3.3.43. PWM1 频率设置 (0x35)

AT 指令:

设置 PWM1 频率: AT+PWM1_PER=n\r\n (以十进制格式发送)

读取 PWM1 频率: AT+PWM1_PER=?\r\n

SPI 数据:

设置 PWM1 频率: n

读取 PWM1 频率: 不填入数据

注: PWM 频率设置范围为 1KHz-2MHz

该功能为订制功能, 用户可根据需要订制。

3.3.44. PWM2 频率设置 (0x36)

AT 指令:

设置 PWM2 频率: AT+PWM2_PER=n\r\n (以十进制格式发送)

读取 PWM2 频率: AT+PWM2_PER=?\r\n

SPI 数据:

设置 PWM2 频率: n

读取 PWM2 频率: 不填入数据

注: PWM 频率设置范围为 1KHz-2MHz

该功能为订制功能, 用户可根据需要订制。

3.3.45. PWM3 频率设置 (0x37)

AT 指令:

设置 PWM3 频率: AT+PWM3_PER=n\r\n (以十进制格式发送)

读取 PWM3 频率: AT+PWM3_PER=?\r\n

SPI 数据:

设置 PWM3 频率: n

读取 PWM3 频率: 不填入数据

注: PWM 频率设置范围为 1KHz-2MHz

该功能为订制功能, 用户可根据需要订制。

3.3.46. PWM0 占空比 (0x4A)

AT 指令:

设置 PWM0 占空比: AT+PWM0_DUTY=n\r\n (以十进制格式发送)

读取 PWM0 占空比: AT+PWM0_DUTY=?\r\n

SPI 数据:

设置 PWM0 占空比: n

读取 PWM0 占空比: 不填入数据

注: PWM 占空比设置范围 1-99, 单位为%

3.3.47. PWM1 占空比 (0x4B)

AT 指令:

设置 PWM1 占空比: AT+PWM1_DUTY=n\r\n (以十进制格式发送)

读取 PWM1 占空比: AT+PWM1_DUTY=?\r\n

SPI 数据:

设置 PWM1 占空比: n

读取 PWM1 占空比: 不填入数据

注：PWM 占空比设置范围 1-99，单位为%

该功能为订制功能，用户可根据需要订制。

3.3.48. PWM2 占空比 (0x4C)

AT 指令：

设置 PWM2 占空比：AT+PWM2_DUTY=n\r\n (以十进制格式发送)

读取 PWM2 占空比：AT+PWM2_DUTY=?\r\n

SPI 数据：

设置 PWM2 占空比：n

读取 PWM2 占空比：不填入数据

注：PWM 占空比设置范围 1-99，单位为%

该功能为订制功能，用户可根据需要订制。

3.3.49. PWM3 占空比 (0x4D)

AT 指令：

设置 PWM3 占空比：AT+PWM3_DUTY=n\r\n (以十进制格式发送)

读取 PWM3 占空比：AT+PWM3_DUTY=?\r\n

SPI 数据：

设置 PWM3 占空比：n

读取 PWM3 占空比：不填入数据

注：PWM 占空比设置范围 1-99，单位为%

该功能为订制功能，用户可根据需要订制。

3.3.50. 实时时钟 (0x2B)

AT 指令：

设置实时时钟：AT+UTC_TIME=ssmmhhDDMMY1Y1Y2Y2\r\n
(以 HEX 格式发送)

UART 设置 UTC 时钟格式：

ss: 秒

mm: 分

hh: 时

DD: 日

MM: 月

Y1Y1:年 (低位)

Y2Y2:年 (高位)

例如：设置时间 2016-01-01 12: 30: 00

AT+UTC_TIME=001E0C0101E007\r\n

读取实时时钟：AT+UTC_TIME=?\r\n

例如：读取时间 2016-01-01 12: 30: 00

模组返回为：AT+OK\r\n2016-01-01\r\n12:30:00\r\n

SPI 数据：

设置实时时钟：xxxx

读取广播开关：不填入数据

SPI 实时时钟设置及读取格式 (HEX)：

- 字节 1：秒
- 字节 2：分
- 字节 3：时
- 字节 4：日
- 字节 5：月
- 字节 6：年 (低位)
- 字节 7：年 (高位)

注：该功能为订制功能，用户可根据需要订制。

3.3.51. ADC 配置寄存器 (0x1F)

AT 指令：

设置 ADC 配置寄存器：AT+ADCCFG=n\r\n (参数参照表 2-7)

读取 ADC 配置寄存器：AT+ADCCFG=?\r\n

SPI 数据：

设置 ADC 配置寄存器：n (参数参照表 2-10)

读取 ADC 配置寄存器：不填入数据

位	名称	初始值	R/W	描述
15-8	滤波次数	0	R/W	填入滤波的次数
7~6	参考电压	0	R/W	参考电压选择 00 : 4.3V 01 : VDD 10 : 外部参考电压 (暂未开放)
5~4	通道选择	0	R/W	通道选择 00 : 通道 0 01 : 通道 1
3~0	采样持续时间	0	R/W	采样持续时间： 0011: 2.7us 0100: 5.3us 0101: 10.6us 0110: 21.3us 0111: 42.6us 1000: 85.3us 1001: 170us 1010: 341us 1011: 682us

表 2-10

注：该功能为订制功能，用户可根据需要订制。

3.3.52. 获取 ADC 值 (0x20)

AT 指令：

读取 ADC 值：AT+ADCVAL=?\r\n

SPI 数据:

读取 ADC 值: 不填入数据

注: 返回的 AD 值未转化为电压值, 需用户自行转换

该功能为订制功能, 用户可根据需要订制。

3.3.53. 电池电量百分比 (0x41)

AT 指令:

读取电池电量百分比: AT+BAT_LEVEL=?\r\n

SPI 数据:

读取电池电量百分比: 不填入数据

注: 该功能为订制功能, 用户可根据需要订制。

3.3.54. 非易失存储器地址 (0x21)

AT 指令:

设置非易失存储器地址: AT+NV_ADDR=n\r\n (n 取值范围 0-8)

读取非易失存储器地址: AT+NV_ADDR=?\r\n

SPI 数据:

设置非易失存储器地址: AT+NV_ADDR=n\r\n (n 取值范围 0-8)

读取非易失存储器地址: 不填入数据

注: 该功能为订制功能, 用户可根据需要订制。

3.3.55. 非易失存储器数据长度 (0x22)

AT 指令:

设置非易失存储器数据长度: AT+NV_DAT_LEN=n\r\n (n 取值范围 1-90)

读取非易失存储器数据长度: AT+NV_DAT_LEN=?\r\n

SPI 数据:

设置非易失存储器数据长度: n (n 取值范围 1-64)

读取非易失存储器数据长度: 不填入数据

注: 该功能为订制功能, 用户可根据需要订制。

3.3.56. 非易失存储器数据 (0x23)

AT 指令:

设置非易失存储器数据: AT+NV_DATA=xxxx\r\n

读取非易失存储器数据: AT+NV_DATA=?\r\n

SPI 数据:

设置非易失存储器数据: 存储数据内容 (HEX 格式)

读取非易失存储器数据: 不填入数据

注: 1. 在写入和读取非易失存储器数据前, 应先设置非易失存储器地址和非易失存储器数据长度。

2. 注意数据长度不应超过 20。

该功能为订制功能, 用户可根据需要订制。

3.3.57. 读写设备 0 蓝牙数据 (0x5C)

AT 指令:

AT+D0=xxx\r\n

SPI 数据:

写入蓝牙数据: 写入要发送的数据内容

读取蓝牙数据: 不填入数据

注: 1. 该指令只会将数据发往已连入的设备 0 中。

2. UART 发送透传数据, 允许 UART 每包最大数据 80 个字节(包括 AT 指令)。

3. SPI 发送透传数据, 允许 SPI 每包最大数据 80 个字节。

4. 模组会自动分包, 以每包 20 个字节的形式进行蓝牙数据发送。

5. 透传数据最大间隔 = (最大数据量(总字节(不包括 AT 指令))/20) * 蓝牙最大发送间隔。

6. 应减少出现每包透传数据的间隔远小于蓝牙的最大连接间隔的情况, 容易导致数据丢失。

7. UART 透传数据发送量计算方法, 计算数据发送量方法如下:

HEX 模式下: 实际发送字节数 = (数据包总字节数 - AT 指令(10 个字节))/2

字符模式下: 实际发送字节数 = 数据包总字节数 - AT 指令(10 个字节)

8. 数据发送间隔不应低于 250ms

3.3.58. 读写设备 1 蓝牙数据 (0x5D)

AT 指令:

AT+D1=xxx\r\n

SPI 数据:

写入蓝牙数据: 写入要发送的数据内容

读取蓝牙数据: 不填入数据

注: 1. 该指令只会将数据发往已连入的设备 1 中。

2. UART 发送透传数据, 允许 UART 每包最大数据 80 个字节(包括 AT 指令)。

3. SPI 发送透传数据, 允许 SPI 每包最大数据 80 个字节。

4. 模组会自动分包, 以每包 20 个字节的形式进行蓝牙数据发送。

5. 透传数据最大间隔 = (最大数据量(总字节(不包括 AT 指令))/20) * 蓝牙最大发送间隔。

6. 应减少出现每包透传数据的间隔远小于蓝牙的最大连接间隔的情况, 容易导致数据丢失。

7. UART 透传数据发送量计算方法, 计算数据发送量方法如下:

HEX 模式下: 实际发送字节数 = (数据包总字节数 - AT 指令(10 个字节))/2

字符模式下: 实际发送字节数 = 数据包总字节数 - AT 指令(10 个字节)

8. 数据发送间隔不应低于 250ms

3.3.59. 读写设备 2 蓝牙数据 (0x5E)

AT 指令:

AT+D2=xxx\r\n

SPI 数据:

写入蓝牙数据: 写入要发送的数据内容

读取蓝牙数据: 不填入数据

注: 1. 该指令只会将数据发往已连入的设备 2 中。

2. UART 发送透传数据, 允许 UART 每包最大数据 80 个字节(包括 AT 指令)。

3. SPI 发送透传数据, 允许 SPI 每包最大数据 80 个字节。

4. 模组会自动分包, 以每包 20 个字节的形式进行蓝牙数据发送。

5. 透传数据最大间隔 = (最大数据量(总字节(不包括 AT 指令))/20) * 蓝牙最大发送间隔。

6. 应减少出现每包透传数据的间隔远小于蓝牙的最大连接间隔的情况, 容易导致数据丢失。

7. UART 透传数据发送量计算方法，计算数据发送量方法如下：

HEX 模式下：实际发送字节数 = (数据包总字节数 - AT 指令 (10 个字节)) / 2

字符模式下：实际发送字节数 = 数据包总字节数 - AT 指令 (10 个字节)

8. 数据发送间隔不应低于 250ms

3.3.60. 读写设备 3 蓝牙数据 (0x5F)

AT 指令：

AT+D3=xxx\r\n

SPI 数据：

写入蓝牙数据：写入要发送的数据内容

读取蓝牙数据：不填入数据

注：1. 该指令只会将数据发往已连入的设备 3 中。

2. UART 发送透传数据，允许 UART 每包最大数据 80 个字节(包括 AT 指令)。

3. SPI 发送透传数据，允许 SPI 每包最大数据 80 个字节。

4. 模组会自动分包，以每包 20 个字节的形式进行蓝牙数据发送。

5. 透传数据最大间隔 = (最大数据量 (总字节 (不包括 AT 指令)) / 20) * 蓝牙最大发送间隔。

6. 应减少出现每包透传数据的间隔远小于蓝牙的最大连接间隔的情况，容易导致数据丢失。

7. UART 透传数据发送量计算方法，计算数据发送量方法如下：

HEX 模式下：实际发送字节数 = (数据包总字节数 - AT 指令 (10 个字节)) / 2

字符模式下：实际发送字节数 = 数据包总字节数 - AT 指令 (10 个字节)

8. 数据发送间隔不应低于 250ms

3.3.61. UART 向所有设备发送透传数据 (/)

AT 指令：

发送 UART 透传数据：AT+DATA=xxx\r\n

SPI 数据：

无

注：1. 该指令会将数据发送所有已连入的设备中。

2. UART 发送透传数据，允许 UART 每包最大数据 80 个字节(包括 AT 指令)。

3. 模组会自动分包，以每包 20 个字节的形式进行蓝牙数据发送。

4. UART 透传数据最大间隔 = (UART 最大数据量 (总字节 (不包括 AT 指令)) / 20) * 蓝牙最大发送间隔。

5. 应减少出现每包串口透传数据的间隔远小于蓝牙的最大连接间隔的情况，这样容易导致数据丢失。

6. 透传数据发送量计算方法，计算数据发送量方法如下：

HEX 模式下：实际发送字节数 = (数据包总字节数 - AT 指令 (10 个字节)) / 2

字符模式下：实际发送字节数 = 数据包总字节数 - AT 指令 (10 个字节)

7. 数据发送间隔不应低于 250ms

3.3.62. UART 接收透传数据说明

UART 模式下，模组收到蓝牙数据后会以 Dn:xxxxxxx 的形式发送给 MCU。其中 Dn 表示连接到模组的设备标号。n 的范围为:0~3。

4. 蓝牙协议（APP 接口）

iOS/Android APP 开发可使用我司提供的 sdk，可参见相关资料《AndroidBLE_API 使用说明》及《iOS BLE_API 使用说明》。以下为蓝牙特性的简要介绍。

4.1. 模块 UUID 说明

UUID 类型	UUID 值	UUID 属性	数据长度(字节)	备注
服务	0X1000	NC	NC	NC
特性	0X1001	READ/WRITE/ NOTIFY	20	BLE 接收
	0X1002	READ/NOTIFY	20	BLE 发送
	0X1003	WRITE	20	REG_WRITE
	0x1004	READ	20	REG_READ
	0x1005	READ/WRITE	20	REG

注：通过 UUID 1003、1004、1005 直接可以修改或设置模组寄存器。

4.2. 模组蓝牙通道操作说明

1. 操作方法举例：查询当前系统状态

第一步：向 REG（UUID1005）写入 0x07（寄存器地址）。

第二步：向 REG_READ（UUID1004）读取，会返回 0x05，表示当前处于从机已经连接状态。

2. 操作方法举例：查询当前命令状态

第一步：向 REG（UUID1005）写入 0x06（寄存器地址）。

第二步：向 REG_READ（UUID1004）读取，会返回 0x01，表示最近一次命令正确。

3. 操作方法举例：设置模组名称

第一步：向 REG（UUID1005）写入 0x06（寄存器地址）。

第二步：向 REG_WRITE（UUID1003）写入

0x62, 0x6c, 0x75, 0x65, 0x74, 0x6f, 0x6f, 0x74, 0x68

第三步：断开连接，重新连接后名字变成“bluetooth”

注：1. 其他寄存器操作及写入内容请看 3.3 命令集说明。

2. 目前 2640 多连接透传有订制功能，所以如果用户去读取未开放的命令则 REG_READ（UUID1004）将全部为 0xFF

5. 工具使用说明

配合手机 APP (TTC-BLE) 以及 PC 端软件 (BleConfig)，可以完成模组功能的调试开发。

5.1. 手机 APP (TTC-BLE)

为方便客户更高效快速完成产品开发，我司可提供与模组配套的 iOS/Android APP 的 SDK, 如有需求，可联系我司业务联系。

5.1.1. 手机 APP 下载

我司免费提供 APP 透传模块软件的技术支持。IOS 用户可以通过苹果的 App Store 搜索“TTC-BLE 数传”，下载安装使用。安卓用户可以通过腾讯应用宝搜索“BLE 数传模块”，下载安装使用。如果需要自行开发 APP 软件，可以联络我司业务人员索取相关平台的 SDK。APP 使用范围 Android4.3 以上的版本或 iPhone 4s 以上的版本。



5.1.2. 数据加密选项设置

点击设置，可取消勾选“加密”“数据头”。



5.1.3. 读取/设置模组参数

操作说明，如下图：

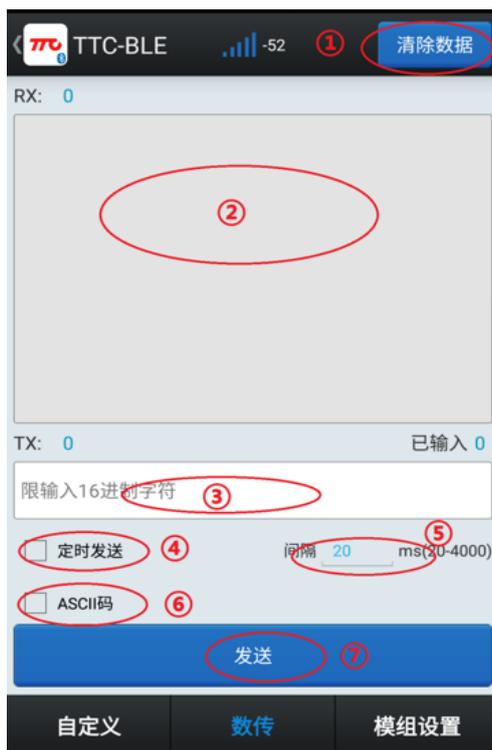
1. APP 会向模组获取当前页面的信息。
2. APP 会读取当前参数信息。
3. 选择或填入需要设置的参数。
4. APP 会设置该参数到模组中。



5.1.4. 接收/发送透传数据

操作说明，如下图：

1. 清空发送计数值，清空接收内容和接收计数值。
2. 显示接收到的数据内容。
3. 输入需要发送的数据。
4. 开启/关闭定时发送数据功能。
5. 修改定时发送数据的间隔时间。
6. 选择发送及显示的格式（ASCII/HEX）。
7. 单次发送数据。



5.2. PC 端软件 (BleConfig)

PC 端软件(BleConfig)用于对我司蓝牙模块进行设置、验证的配套软件，通过图形化的界面避免了记忆多条指令，方便客户进行调试。通过本软件可以轻松的对模块进行主从机连接、参数配置、数据收发、连续发送数据等操作。如有必要，本软件也可以临时作为串口工具来使用。

具体使用说明，请参见《BleConfig 使用说明》。

6. 文件修订说明

版本号	时间	修订内容
V1.0	2016-05-04	透传规格书发布
V1.1	2016-05-10	1.修改规格书中描述有误的内容 2.修改 APP 图片
V1.2	2016-05-15	1.修改功耗测试结果 2.修改公司地址
V1.3	2016-05-25	1.修改串口波特率设置范围 2.添加实物图及尺寸
V1.4	2016-06-01	1.移除实物图及尺寸部分 2.添加 5x5 脚位表格
V1.5	2016-06-23	1.修改官网地址链接
V1.6	2016-07-16	1.新增透传多连说明 2.修改指令表 3.修改波特率序号，与 CC2541 透传对应
V1.7	2016-08-01	1.修改 SPI 命令号，与 C2541 透传对应
V1.8	2016-10-20	1.修改 5x5 模组型号
V1.9	2017-05-31	1.修改脚位定义
V2.0	2017-07-01	1.修改脚位图、脚位定义等 2.规格书分为新旧脚位 2 种版本

7. 联系我们

深圳市昇润科技有限公司

ShenZhen ShengRun Technology Co.,Ltd.

Tel: 0755-86233846 Fax: 0755-82970906

官网地址: www.tuner168.com

阿里巴巴网址: <http://shop1439435278127.1688.com>

E-mail: marketing@tuner168.com

地址: 广东省深圳市南山区西丽镇龙珠四路金谷创业园 B 栋 6 楼 601-602

