



## BH-HC05 蓝牙透传模块 用户手册

### 修订历史

日期	版本	更新内容
2016/11/14	1.0.0	-
2017/8/5	1.0.1	新增 407、MINI 开发板说明





## 文档说明

本手册旨在帮助用户正确构建 BH-HC05 蓝牙模块的使用环境，引导用户快速使用该模块。

关于模块的原理图、机械尺寸等说明请参考《HC05-蓝牙模块原理图》文档。

关于 HC05 模块的 AT 指令说明，请参考《AT 指令说明》文档。



## 目录

BH-HC05 蓝牙透传模块 .....	1
用户手册.....	1
文档说明.....	2
目录 .....	3
1. HC05 简介 .....	4
1.1 HC05 简介 .....	4
1.2 特性参数.....	4
2. 硬件测试.....	5
2.1 硬件连接.....	5
2.2 测试流程.....	7
2.2.1 与安卓手机进行蓝牙通讯.....	8
2.2.2 两个开发板间蓝牙通讯.....	11
3. HC05 注意事项及常见问题 .....	14
4. 配套程序说明.....	15
4.1 HC05 驱动 .....	15
4.2 扫描蓝牙设备.....	17
4.3 main 函数执行流程.....	19
5. 产品更新及售后支持.....	20



## 1. HC05 简介

### 1.1 HC05 简介

BH-HC05 是秉火科技推出的蓝牙串口模块, 它采用蓝牙 2.0 协议, 可与任何版本的蓝牙兼容通讯, 包括与具有蓝牙功能的电脑、蓝牙主机、手机、PDA、PSP 等终端配对, 可实现串口透传功能。驱动 HC05 模块时只需要使用 TTL 电平标准的串口即可(5V/3.3V 电压均可), 支持的波特率范围为 4800~1382400, 非常适合用于单片机系统扩展蓝牙特性。

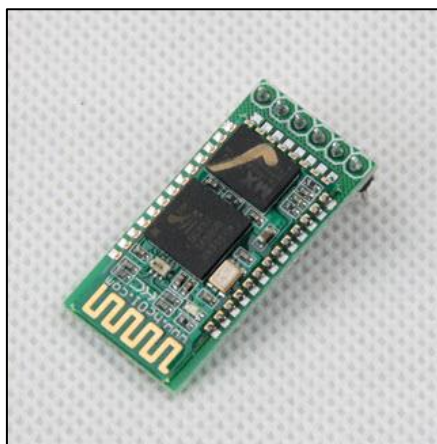


图 1-1 HC05 外观图

### 1.2 特性参数

参数	说明
供电	3.3V-5V
通讯接口	TTL 串口, 支持波特率 4800~1382400, 默认波特率为 9600。
通讯距离	10 米
频段	2.40GHz-2.48GHz
蓝牙协议	蓝牙 2.0。带 EDR, 2Mbps-3Mbps 调制度
存储器	8Mbit FLASH
天线	板载 2.4GHz 天线, 自适应跳频技术
无线发射功率	-4~6dBm 可调(蓝牙 Class2 功率级别)
无线接收灵敏度	-80dBm
误码率	0。但会在传输链路产生误码, 如 RS232 和 TTL 线路处理中。
应用领域	蓝牙转串口透传数据。
工作温度	-25~ +75℃
功耗	配对中: 30~40mA 通讯中: 5~20mA



已配对未通讯: 1~8mA 休眠电流: 小于 1mA
-------------------------------

## 2. 硬件测试

本模块配套 STM32 驱动程序, 可直接使用秉火 F103 指南者、霸道及 F429 挑战者开发板进行测试。按要求使用杜邦线把模块连接到开发板, 并下载对应的程序即可。

### 2.1 硬件连接

BH-HC05 模块外观见图 2-1, 模块引出了 6 个引脚, 在其背面标有引脚名称的丝印。





表 2-1 HC05 模块引脚说明

序号	引脚名称	说明	与 F103 霸道、指南者及 F407 霸天虎连接	与 F429 挑战者连接	与 F103-MINI 开发板连接
1	VCC	3.3/5V 电源输入	接 3.3V 或 5V	接 3.3V 或 5V	接 3.3V 或 5V
2	GND	地线	GND	GND	GND
3	TXD	串口数据发送引脚, TTL 电平	PA3 (注意跳帽)	PD6 (注意跳帽)	PA3 (注意跳帽)
4	RXD	串口数据接收引脚, TTL 电平	PA2 (注意跳帽)	PD5 (注意跳帽)	PA2 (注意跳帽)
5	KEY	模式引脚, 悬空时默认为低电平 <input type="checkbox"/> 高电平时模块进入 AT 命令模式 <input type="checkbox"/> 低电平时模块为串口透传模式	PB14	PB11	PA5
6	INT	配对状态输出 <input type="checkbox"/> 配对状态时输出为高电平 <input type="checkbox"/> 未配对时输出为低电平	PB13	PB10	PA7

**注意跳帽:** 在 F103 霸道及 F407 霸天虎开发板 USART2 的 PA2/PA3, F429 挑战者开发板 USART2 的 PD5/PD6 默认都通过跳线帽连接到了板载的 MAX3232 芯片, 存在引脚共用的问题, 使用蓝牙模块时, 必须把板子 JTAG 接口附近的这部分跳线帽拔掉, 见表 2-2。

表 2-2 F103 霸道、F407 霸天虎及 F429 挑战者需要拔掉的跳线帽

F103 霸道及 F407 霸天虎开发板	F429 挑战者开发板
拔掉跳帽: PA2<->T1IN、	拔掉跳帽: PD5<->T1IN、
拔掉跳帽: PA3<->R1OUT	拔掉跳帽: PD6<->R1OUT

在 F103 指南者及 F103-MINI 开发板没有这个问题, 只要连接好杜邦线即可。

按照表 2-1 把 BH-HC05 模块与各个开发板连接后的效果见图 2-2。



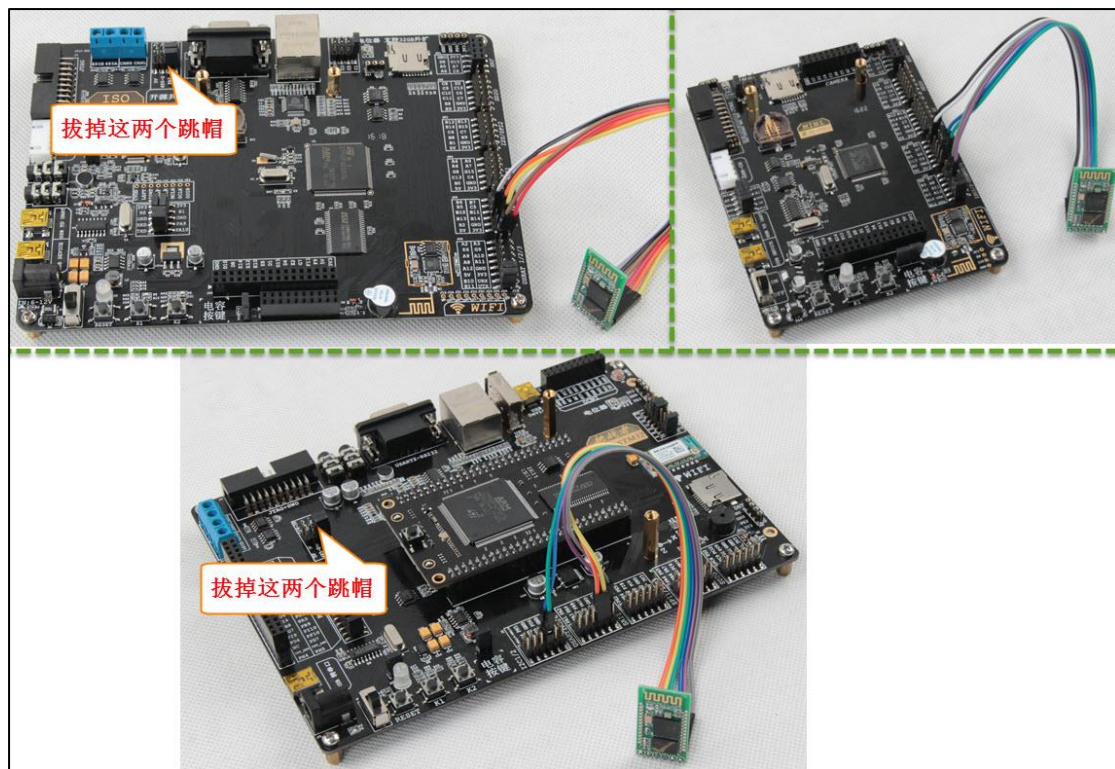


图 2-2 BH-HC05 模块与各个开发板的连接图 1

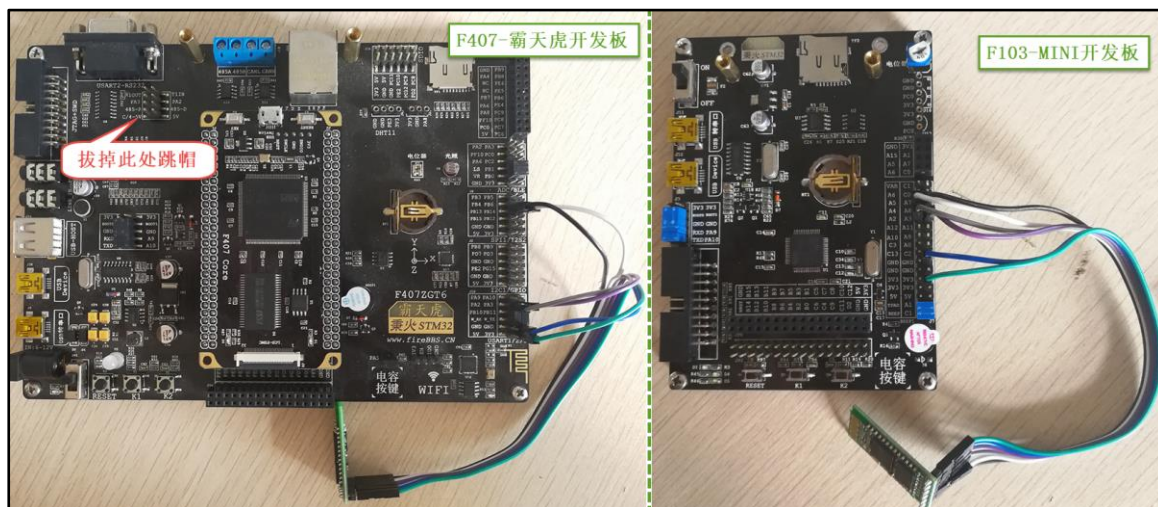


图 2-3 BH-HC05 模块与各个开发板的连接图 2

## 2.2 测试流程

HC05 模块支持两种测试方式:

- ❑ 方法 1: 使用 1 个开发板连接 1 个 HC05 模块。使用安卓手机与 HC05 模块进行蓝牙通讯。



- ❑ 方法 2: 使用 2 个开发板, 各连接 1 个 HC05 模块。两个开发板之间进行蓝牙串口透传通讯。

请用户根据自己的硬件环境选择测试方式, 以下对两种测试方式进行步骤说明。

## 2.2.1 与安卓手机进行蓝牙通讯

### 1. 准备测试环境

- ❑ 在安卓手机上安装“BTClient”APP。  
BTClient APP 所在目录: “配套软件\HC-PDA-ANDROID.apk”。
- ❑ 按表 2-1 连接好模块后, 找到配套资料里的例程“1. HC05 测试程序(带液晶)和 2. HC05 测试程序(不带液晶)”这两个程序均可, 根据自己的实验平台选择, 带液晶的程序需要给开发板接上配套的液晶, 不带液晶的程序通过串口输出运行状态信息。使用 MDK 编译并下载该程序到开发板, 复位开发板让程序运行。  
工程所在目录: “开发板配套例程\霸道版本[或指南者、挑战者版本]”。

### 2. 正常运行的实验现象

- ❑ 开发板的液晶屏会显示“HC05 BlueTooth demo, HC05 module detected! Device name :HC05\_SLAVE”三行文字, 见图 2-4, , 长按开发板的 KEY2 按键(注意不要按 KEY1), 会随机生成一个新名字;

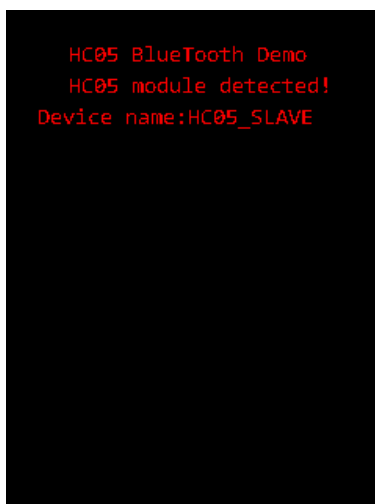


图 2-4 HC05 模块正常运行时的液晶显示

- ❑ 打开手机的 BTClient APP, 打开手机的蓝牙设备, 点击界面左下角的“连接”按钮, 会弹出“查找设备中...”的窗口, 稍等一下会出现名为“HC05\_SLAVE”的蓝牙设备, 这就是我们的 HC05 模块的名字, 点击该设备, APP 会与 HC05 尝试连接, 输入配对码“1234”, 即可连接成功。APP 的操作见图 2-5, 开发板的液晶显示见图 2-6。



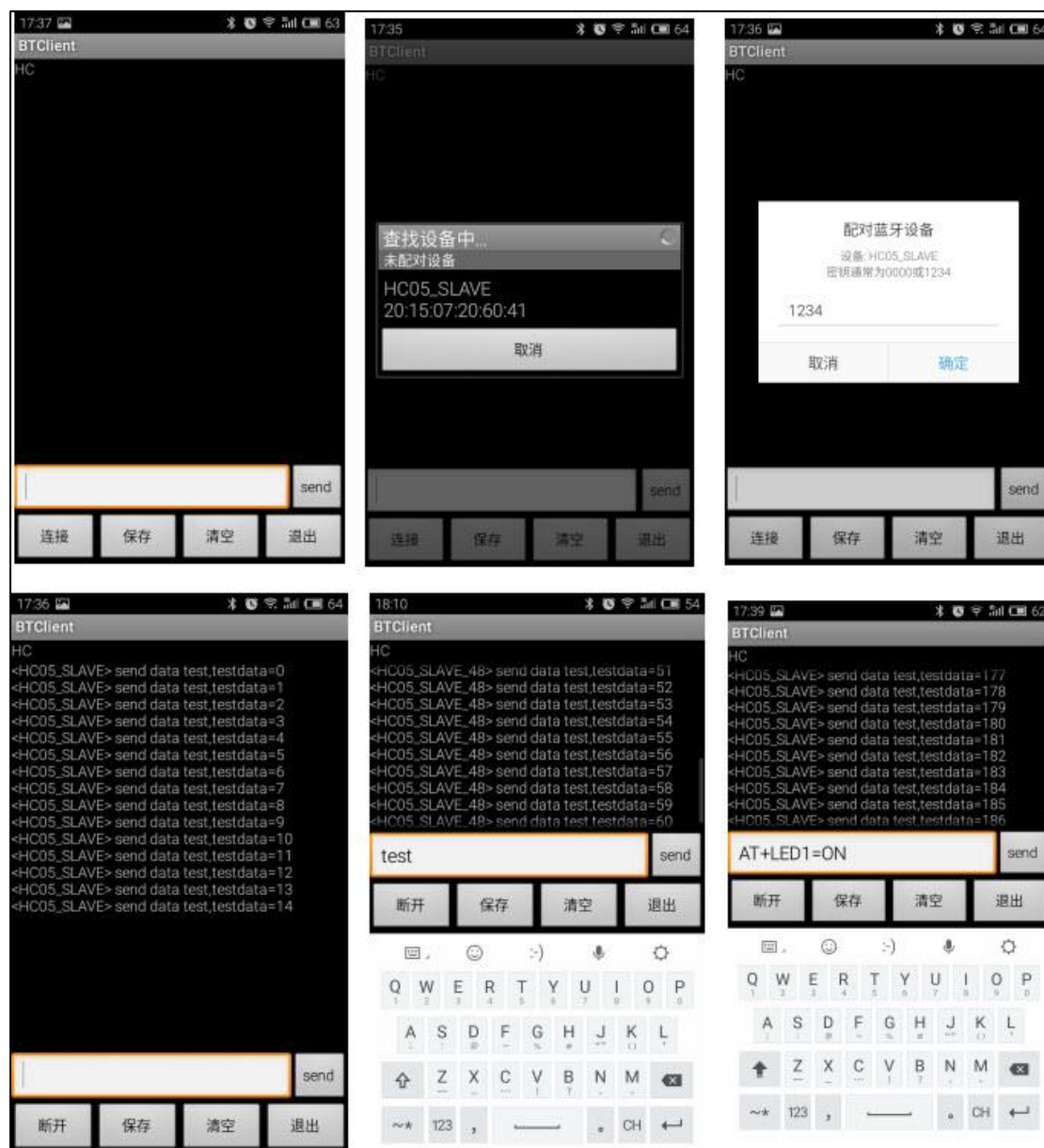


图 2-5 手机上 BTClient 的操作

- ❑ APP 与模块连接上后, HC05 模块会主动向 APP 发送类似 “<Device Name> send data test, test data=<x>” 的字样, 其中<Device Name>为 HC05 模块的名字, <x>则为一个自加的字符串编号。
- ❑ APP 与模块连接上后, 可以使用 APP 通过蓝牙向开发板发送数据。点击界面中的输入框, 输入字符串, 然后点击 “send” 发送数据, 开发板接收到数据后会在液晶上原样显示出来。(可输入回车, 但程序中的液晶处理只显示第一行数据。)
- ❑ 除了发送普通字符串, 还可以向开发板发送命令控制板子上的 LED 灯。开发板配套的程序支持两条命令: “AT+LED1=ON” 和 “AT+LED1=OFF”, 分别用于点亮开发板上的 LED1 和关闭 LED1。开发板接收到命令后, 会在液晶上



显示接收到的命令，并且控制 LED 灯。如果需要其它命令，用户可修改配套的 STM32 程序，实现自定义的功能。



图 2-6 与手机 APP 通讯时的液晶显示界面

实际运行时，液晶界面显示的“receive data”内容可能有细微差异，它不影响程序状态。

### 3. 不正常运行时故障排查

液晶屏显示“No HC05 detected!”，见图 2-7。这时说明开发板检测不到 HC05 模块，请参照引脚连接表重新检查模块与开发板之间的连线，并给开发板上电重新测试。

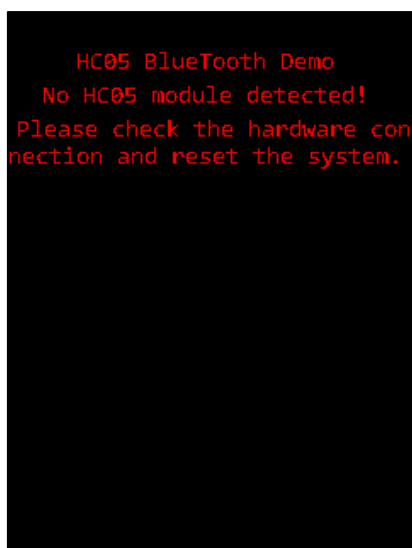


图 2-7 模块运行不正常时的液晶显示

## 2.2.2 两个开发板间蓝牙通讯

### 1. 准备测试环境

分别对两个开发板及模块按表 2-1 连接好，找到配套资料里的例程“**1. HC05 测试程序(带液晶)**和**2. HC05 测试程序(不带液晶)**”这两个程序均可，根据自己的实验平台选择，带液晶的程序需要给开发板接上配套的液晶，不带液晶的程序通过串口输出运行状态信息。使用 MDK 编译并下载该程序到两个开发板，复位开发板让程序运行。

工程所在目录：“开发板配套例程\霸道版本[或指南者、MINI、霸天虎、挑战者版本]”。

### 2. 正常运行的实验现象

- 开发板的液晶屏会显示“HC05 BlueTooth demo, HC05 module detected! Device name :HC05\_SLAVE”三行文字，确认两个开发板的液晶都显示这样的状态。见图 2-4；

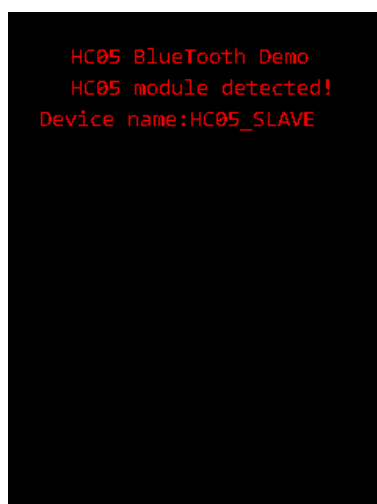


图 2-8 HC05 模块正常运行时的液晶显示

- 现在两个开发板的 HC05 模块都处于 **SLAVE** 模式，模块处在 SLAVE 模式时是不能发起蓝牙连接的。选择其中一个开发板，**长按它的 KEY1 按键**，然后它会自动执行后面的扫描、配对、连接流程。该开发板会把它的 HC05 模块设置为 **MASTER** 模式，并扫描周边的蓝牙设备，若扫描到有设备名称包含“HC05”字符串的设备，则发起蓝牙连接，连接成功后两个蓝牙模块互相发送数据，并在液晶屏上显示收到的最新数据。开发板的液晶显示见图 2-6。



图 2-9 两个开发板之间蓝牙通讯时 MASTER 的界面



实际运行时，液晶界面显示的“receive data”内容可能有细微差异，它不影响程序状态。

### 3. 不正常运行时故障排查

液晶屏以显示“No HC05 detected!”，见图 2-7。这时说明开发板检测不到 HC05 模块，请参照引脚连接表重新检查模块与开发板之间的连线，并给开发板上电重新测试。

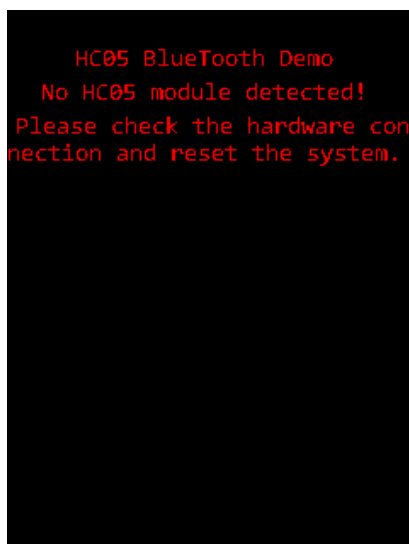


图 2-10 模块运行不正常时的液晶显示



## 3. HC05 注意事项及常见问题

### 1. 初始化流程(非常重要)

蓝牙模块上电后必须要执行 INIT 命令, 该命令会初始化蓝牙的 SPP 规范, 只有执行 INIT 命令后, 才能进行扫描、配对、连接、串口透传等操作。初始化后还必须要调用 INQ 命令发起查询, 才会被远端的蓝牙设备查找到。INQ 命令发送一次后就会一直执行查询, 如果模块配对、连接后 INQ 命令干扰了串口透传, 可使用 INQC 命令结束 INQ 查询, 它不会影响连接。

### 2. 主机(MASTER)与从机(SLAVE)有什么区别呢?

- ❑ 主机和从机在核心模块的硬件上是一样的, 区别只在于软件、使用 AT 命令可使模块切换主从机模式。
- ❑ HC05 模块只有主机才能主动发起配对、连接。连接后主机和从机都可以主动互相发送数据, 通信都是全双工的, 也就是收发可同时进行。
- ❑ 从机和主机配对使用, 也都可以和普通的蓝牙适配器配对使用, 分出主机或从机的目的之一是和别的产品配对, 比如有些 GPS 定位系统本身配有从机的, 那么就应该使用主机的蓝牙模块和他配对。
- ❑ 从机模块是相对于主机模块和适配器而言的, 但主机模块如果和电脑的适配器配对的话, 也可以认为是从机的。

### 3. 关于模块上的 LED 灯

模块上的 LED 是配对状态指示灯。

- ❑ 刚上电时模块进入 AT 状态: 指示灯 1 秒闪 1 次。
- ❑ 可配对状态: 指示灯 1 秒闪 2 次
- ❑ 配对成功状态: 指示灯 1 次闪 2 下, 2 秒闪一次。

### 4. 关于模块上的 KEY 引脚

模块上的 KEY 引脚是用于指示输入的是透传数据还是 AT 命令的。

- ❑ KEY 引脚为高电平: 向 HC05 模块发送的字符串都被当作 AT 命令来处理。
- ❑ KEY 引脚为低电平: 向 HC05 模块发送的数据都原样使用蓝牙协议发送到远端的蓝牙设备。(字符串、纯字节数据都一样, 原样输出, 具体格式由接收端处理。)





## 4. 配套程序说明

BH-HC05 模块一共配套了 2 个例程, 用户可根据需求选择相应的程序来学习。

例程所在目录: “开发板配套例程\霸道版本[或指南者、挑战者版本]”。

程序	说明
1.HC05 测试程序(带液晶)	HC05 模块测试程序, 带液晶状态显示, 方便测试
2.HC05 测试程序(不带液晶)	HC05 模块测试程序, 不带液晶状态显示, 可使用串口调试助手查看测试信息。由于这个程序不包含液晶显示驱动, 十分方便移植

### 4.1 HC05 驱动

配套的两个程序除了液晶信息输出, 其它部分是完全一样的, 这里我们以 F103 霸道开发板 “1.HC05 测试程序(带液晶)” 来讲解 (F429 挑战者程序中的液晶显示函数稍有区别, 其它内容一致)。

HC05 的驱动包含了 `bsp_usart_blt.c` 及 `bsp_hc05.c` 文件中。`bsp_usart_blt.c` 文件中主要是配置控制蓝牙模块的 `usart` 工作模式, 以及处理从 HC05 模块处接收到的数据, 进行基本处理。`bsp_hc05.c` 文件包含命令发送、设备管理等 HC05 功能函数。

`bsp_usart_blt.c` 程序中控制 STM32 使用 USART 与 HC05 模块通讯, 使用 USART 接收中断模式来处理 HC05 发送给 STM32 的数据。它在中断服务函数中把接收到的数据存储到一个静态缓冲区中, 核心代码见代码清单 4-1。

代码清单 4-1 USART2 中断接收缓冲 核心代码 (位于 `bsp_usart_blt.c` 文件)

```
1 //中断缓存串口数据
2 #define UART_BUFF_SIZE      1024
3 volatile uint16_t uart_p = 0;
4 uint8_t      uart_buff[UART_BUFF_SIZE];
5
6 void bsp_USART_Process(void)
7 {
8     if (uart_p < UART_BUFF_SIZE) {
9         if (USART_GetITStatus(BLT_USARTx, USART_IT_RXNE) != RESET) {
10             uart_buff[uart_p] = USART_ReceiveData(USART2);
11             uart_p++;
12         }
13     } else {
14         USART_ClearITPendingBit(BLT_USARTx, USART_IT_RXNE);
15         clean_rebuff();
16     }
17 }
18
19
20
21 //获取接收到的数据和长度
22 char *get_rebuff(uint16_t *len)
23 {
24     *len = uart_p;
25     return (char *)&uart_buff;
26 }
```



```
27
28 //清空缓冲区
29 void clean_rebuff(void)
30 {
31
32     uint16_t i=UART_BUFF_SIZE+1;
33     uart_p = 0;
34     while (i)
35         uart_buff[--i]=0;
36
37 }
38
```

其中的 `bsp_USART_Process` 函数直接在 USART 的接收中断服务函数中调用, 把每个接收到的字节数据都存储在静态变量 `uart_buff` 中, 并用 `uart_p` 表示接收到的数据长度。`get_rebuff` 函数则用于返回 `uart_buff` 的指针及其长度, 在需要处理接收数据的时候, 使用该函数获取 `uart_buff` 缓冲的数据。`clean_rebuff` 函数则用于清空 `uart_buff` 的数据, 一般在处理完缓冲数据后调用。

`bsp_hc05.c` 文件的核心函数为 `HC05_Send_CMD` 及 `HC05_SendString` 函数, 见代码清单 4-2。这两个函数分别用于向 HC05 模块发送命令及蓝牙透传数据。

代码清单 4-2 HC05 驱动核心代码 (位于 `bsp_hc05.c` 文件)

```
1
2 /**
3  * @brief 向 HC05 模块发送命令并检查 OK。只适用于具有 OK 应答的命令
4  * @param cmd:命令的完整字符串, 需要加\r\n。
5  * @param clean 命令结束后是否清除接收缓冲区, 1 清除, 0 不清除
6  * @template 复位命令: HC05_Send_CMD("AT+RESET\r\n",1);
7  * @retval 0, 设置成功; 其他, 设置失败。
8  */
9 uint8_t HC05_Send_CMD(char* cmd, uint8_t clean)
10 {
11     uint8_t retry=5;
12     uint8_t i,result=1;
13
14     while (retry-->0) {
15         BLT_KEY_HIGHT;
16         delay_ms(10);
17         Uart_SendString(HC05_USART, (uint8_t *)cmd);
18
19         for (i=0; i<20; i++) {
20             uint16_t len;
21             char * redata;
22
23             delay_ms(10);
24
25             redata = get_rebuff(&len);
26             if (len>0) {
27                 if (redata[0]!='\0') {
28                     HC05_DEBUG("send CMD: %s",cmd);
29
30                     HC05_DEBUG("receive %s",redata);
31                 }
32                 if (strstr(redata,"OK")) {
33                     if (clean==1)
34                         clean_rebuff();
35                     return 0;
36                 } else {
37                     //clean_rebuff();
38                 }
39             } else {
40                 } else {
```



```
41         delay_ms(100);
42     }
43 }
44     HC05_DEBUG("HC05 send CMD fail %d times",retry);
45 }
46
47     HC05_DEBUG("HC05 send CMD fail ");
48
49     if (clean==1)
50         clean_rebuff();
51
52     return result ;
53
54 }
55
56 /**
57  * @brief 使用 HC05 透传字符串数据
58  * @param str,要传输的字符串
59  * @retval 无
60  */
61 void HC05_SendString(char* str)
62 {
63     BLT_KEY_LOW;
64
65     Usart_SendString(HC05_USART,(uint8_t *)str);
66
67 }
```

HC05 模块根据它 KEY 引脚的电平来判断从串口接收到的是命令还是要进行串口透传的数据。所以 HC05\_Send\_CMD 函数和 HC05\_SendString 最本质的区别就是一个在发送命令前使用 **BLT\_KEY\_HIGHT** 语句控制 KEY 引脚为高电平，另一个在发送数据前使用 **BLT\_KEY\_LOW** 语句控制 KEY 引脚为低电平，实际上都是使用 Usart\_SendString 函数调用 USART2 向 HC05 发送字符串。由于需要判断命令响应，HC05\_Send\_CMD 函数发送命令后，还对 HC05 模块返回的响应进行判断，如果响应中包含“OK”，则返回 0 表示命令控制成功，否则将多次重发命令进行尝试。

## 4.2 扫描蓝牙设备

为了方便管理，我们定义了一个蓝牙设备列表的数据结构，用于存储扫描到的远程蓝牙设备数量、地址、名字，见代码清单 4-3。

代码清单 4-3 蓝牙设备列表数据结构

```
1  /*蓝牙地址，数字形式，分 NAP, UAP, LAP 段*/
2  typedef struct {
3      uint16_t NAP;
4      uint8_t UAP;
5      uint32_t LAP;
6  } BLTAddr;
7
8
9  typedef struct {
10     uint8_t num;                //扫描到的蓝牙设备数量
11
12     BLTAddr addr[BLTDEV_MAX_NUM]; //蓝牙设备地址，数字形式
13 }
```



```
14     char unpraseAddr[BLTDEV_MAX_NUM][50];    //蓝牙设备地址，字符串形式，方便扫描时和连接时使用
15
16     char name[BLTDEV_MAX_NUM][50];           //蓝牙设备的名字
17
18 } BLTDev;
```

HC05 模块初始化完毕，并设置为 MASTER 模式后，可定时调用 linkHC05 函数，它会自动扫描周边的蓝牙设备、获取蓝牙设备名字、打印搜索到的蓝牙设备信息，然后查找是否有蓝牙设备名称含有“HC05”字符串，若有则自动进行配对、连接，见代码清单 4-4。

代码清单 4-4 扫描、连接远程蓝牙设备

```
1 /**
2  * @brief  扫描蓝牙设备，并连接名称中含有"HC05"的设备
3  * @param  无
4  * @retval 0 获取成功，非 0 不成功
5  */
6 uint8_t linkHC05(void)
7 {
8     uint8_t i=0;
9     char cmdbuff[100];
10
11     parseBluetoothAddress(&bltDevList);
12     getRemoteDeviceName(&bltDevList);
13     printBLTInfo(&bltDevList);
14
15
16     for (i=0; i<=bltDevList.num; i++) {
17         if (strstr(bltDevList.name[i], "HC05") != NULL) {
18             //非 NULL 表示找到有名称部分为 HC05 的设备
19             HC05_INFO("搜索到远程 HC05 模块，即将进行配对连接...");
20
21             LCD_SetColors(YELLOW, BLACK);
22             ILI9341_Clear(0, 80, 240, 20);
23             ILI9341_DispString_EN( 5, 80, "Found a HC05 ,conecting..." );
24
25             strBLTAddr(&bltDevList, ',');
26
27             //配对
28             sprintf(cmdbuff, "AT+PAIR=%s,20\r\n", bltDevList.unpraseAddr[i]);
29             HC05_Send_CMD(cmdbuff, 0);
30
31             //连接
32             sprintf(cmdbuff, "AT+LINK=%s\r\n", bltDevList.unpraseAddr[i]);
33             return HC05_Send_CMD(cmdbuff, 0);
34         }
35     }
36
37     return 1;
38 }
39
40 }
```



## 4.3 main 函数执行流程

main 函数代码较长, 这里就不帖代码了, 请读者打开工程直接对照阅读。了解 main 函数的执行流程有助于读者掌握 HC05 模块的初始化及各种功能的使用方法。它具体的执行流程如下:

- ❑ 初始化 SysTick 定时器, 每 1ms 产生一次定时中断。
- ❑ 初始化液晶、USART1、板子上的 LED 灯、按键。
- ❑ 调用 HC05\_Init 函数初始化 HC05, 若检测到 HC05 模块, 则程序继续正常往下执行。
- ❑ 调用 HC05\_Send\_CMD 函数, 对模块进行复位(RESET)、恢复出厂设置(ORGL)。
- ❑ 调用 HC05\_Send\_CMD 函数, 演示如何向 HC05 模块发送各种命令, 默认这部分是演示是不在液晶屏上显示出来的, 如果想查看命令执行结果, 需要把 bsp\_hc05.h 文件中的 HC05\_DEBUG 宏设置为 1, 重新编译下载程序执行, 用串口调试助手即可查看命令执行结果。
- ❑ 执行 INIT 命令, 把模块初始化为蓝牙的 SPP 规范, 只有执行了这个命令后, 蓝牙模块才能进行扫描、配对、连接、串口透传等操作。
- ❑ 执行命令把模块的名字设置为 “HC05\_SLAVE”。

接着, 在 while 循环里定时执行各种操作:

- ❑ 若模块未处于连接状态, 根据主模式或从模式进行不同的操作(上电后默认为从模式, 按下 KEY1 后, 模块会进入主模式)。主模式下, 调用 linkHC05 函数, 尝试扫描周边的蓝牙设备, 若找到 “HC05” 名字的设备则进行主动连接。从模式下, 执行 INQ 命令, 只有执行这个命令, HC05 模块才会向周边发射蓝牙讯号, 不然不会被其它蓝牙设备找到的。
- ❑ 若模块处于连接状态, 定时检查接收缓冲区, 判断接收到的字符串是否为 “AT+LED1=ON” 和 “AT+LED1=OFF” 命令, 若是则进行相应的操作, 若不是则把接收缓冲区的第一行字符串显示出来。在这里用户可参照该范例自定义其它的命令。
- ❑ 若模块处于连接状态, 则每隔几秒向配对的蓝牙设备发送一段字符串, 用于演示。
- ❑ 若 KEY1 被长按下, 则转换 HC05 模块的 MASTER 模式/SLAVE 模式, 并更改设备的名字。
- ❑ 若 KEY2 被长按下, 则更换 HC05 模块的名字, 防止与周边的蓝牙设备名重复(其实名字重复也没问题, 只是方便用户识别而已)。



## 5. 产品更新及售后支持

秉火的产品资料更新会第一时间发布到论坛: <http://www.firebbs.com>

购买秉火产品请到秉火官方淘宝店铺: <https://fire-stm32.taobao.com>

在学习或使用秉火产品时遇到问题可在论坛发帖子与我们交流。