



## ATS308X 蓝牙指标测试指南

版本号: 1.03  
2024-07-12

## 声明

### 免责

本文档中提供的信息仅作为使用炬芯产品的参考或示例，不能作为任何报价或销售合同的一部分。

炬芯产品可能包含被称为异常或勘误的设计缺陷或错误，这些缺陷或错误可能导致产品的功能偏离已发布的规格。设计者不得依赖炬芯产品上标有“保留”或“未定义”的说明。炬芯保留这些内容以供未来定义，未来对于这些内容的更改所产生的冲突或不兼容，炬芯不承担任何责任。

炬芯否认并排除任何及所有保证，包括但不限于对本文档及炬芯产品的信息的适销性、准确性、安全性、针对特定目的的适用性、所有权、以及防止知识产权侵权等方面的任何及所有明示或默示保证。

在任何情况下，炬芯不对任何直接、附带、间接、特殊、惩罚性或相应的损害负责，包括但不限于因使用本文档信息和炬芯产品而造成的任何类型的数据、利润、储蓄或收入的损失，无论诉讼形式如何，是否基于合同、侵权行为、疏忽行为或其他过失、严格责任、或其他归责方式；是否给予买方的任何补救措施未能达到其基本目的，以及是否已将此类损害的可能性告知炬芯。

炬芯产品不得用于任何可能导致人身伤害或严重财产损失的生命支持领域或其他应用。未经炬芯事先书面批准的任何及所有此类使用，以及进一步的测试和/或修改，将完全由客户承担风险。

### 信息获取方式

本文档和/或炬芯其他产品资料，以及销售协议的条款和条件的副本，可通过访问炬芯网站(<http://www.actions-semi.com>)或从炬芯授权代表处获取。

### 商标

“炬芯”、logo 及 “Actions” 为炬芯科技股份有限公司的商标。其他公司及其产品的名

---

称和品牌可能不时出现在本文档中，是其各自持有人的商标，除非有明确说明，不存在任何隶属关系、授权或背书。

## 权利保留

本文档的规定不应被视为授予买方关于炬芯或其他公司的专利、版权、商标、商业秘密、专有技术和任何其他知识产权的任何权利。

## 其它

本文档中所包含或描述的信息仅与炬芯产品有关，并在本文档发布之日废除并取代炬芯或任何其他意图传播该等信息的实体或人此前提提供的与该等产品有关的所有已发布的数据和规格。炬芯有权在任何时候更改本文档所述的信息，恕不另行通知。在订购产品之前，请与您的炬芯销售代表联系以获取最新信息。

## 额外支持

更多产品和公司信息可访问炬芯网站：<http://www.actions-semi.com>。

# 目 录

声明 .....	1
目 录.....	3
<b>1 版本历史.....</b>	<b>4</b>
<b>2 引 言.....</b>	<b>4</b>
2.1 编写目的.....	4
2.2 文档范围.....	4
2.3 术语和缩写词.....	4
<b>3 整机非信令测试.....</b>	<b>5</b>
3.1 固件生成方法.....	5
3.2 整机测试操作流程.....	5
3.3 整机非信令 TX 命令(BR/EDR/LE).....	7
3.4 整机非信令 RX 命令(BR/EDR/LE).....	8
3.5 整机精简 TX 命令(Only LE).....	9
3.6 整机精简 RX 命令(Only LE).....	10
3.7 整机 LE 低中高信道交替 TX 命令(压力测试).....	10
<b>4 PCBA UART 非信令测试.....</b>	<b>11</b>
4.1 固件生成方法(仅测试修改).....	11
4.2 UART 模式测试.....	11
<b>5 测试 API 定义 .....</b>	<b>12</b>

---

# 1 版本历史

日期	版本号	注释	作者
2024-04-26	1.0	建立初始版本	ATS308X 项目组
2024-07-12	1.03	修改 FCC 测试方法	ATS308X 项目组

# 2 引言

## 2.1 编写目的

本文主要向客户介绍如何制作测试蓝牙指标的固件，以及整机和PCBA测试蓝牙指标的方法。

## 2.2 文档范围

本文档包含如何生成测试固件，BLE 模式(整机)和 UART 模式(PCBA)的测试流程，以及对应的接口函数。

## 2.3 术语和缩写词

缩写和术语	解释
BLE	Bluetooth Low Energy

## 3 整机非信令测试

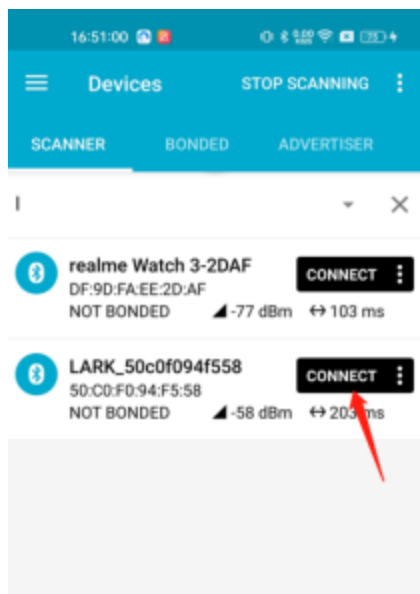
### 3.1 固件生成方法

➤ 做如下修改，然后编译烧录  
application/bt\_watch/prj.conf 打开  
CONFIG\_NSM\_APP=y  
将\application\bt\_watch\prebuild\bt\_production\_test\mp\_btc.bin  
拷贝到\application\bt\_watch\fs\_sdfs\sdfs\_k\mp\_btc.bin,  
然后删除\application\bt\_watch\fs\_sdfs\sdfs\_k\fcc.bin

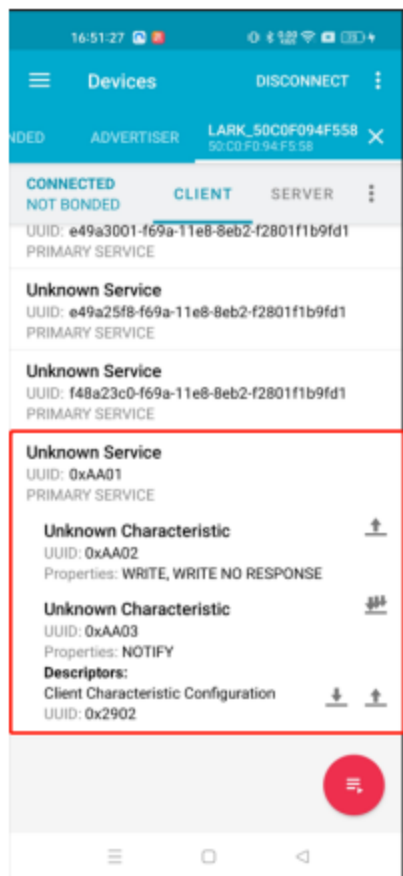
测试仪器操作说明参考：20220207\_如何用 CMW500 非信令模式测试蓝牙射频.pdf ——  
<https://redmine.actions-semi.com/attachments/1771>

### 3.2 整机测试操作流程

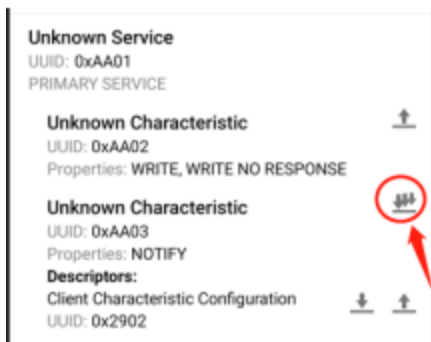
1. 先用手机下载第三方 APP 软件 nrf connect 连接 LE(截图以 realme GT 为例);



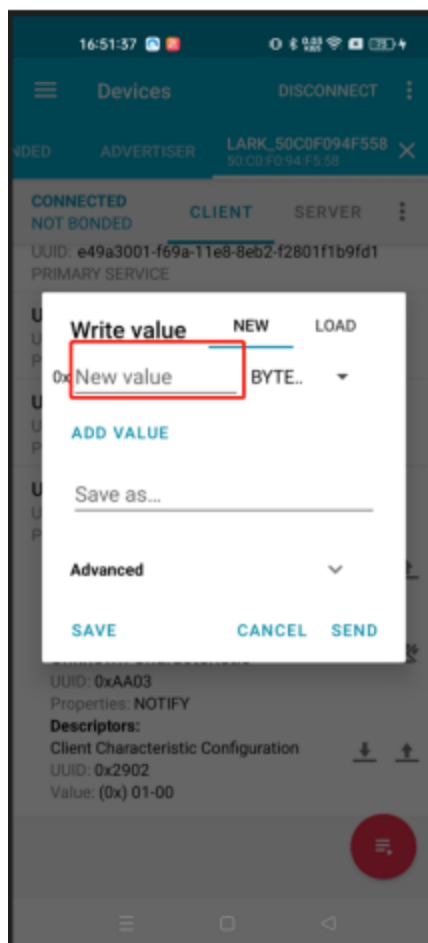
2. 测试对应的 UUID 是 0xAA01,0xAA02(Write),0xAA03(Notify);



3. 打开消息提醒，收到的数据会在 AA03 服务显示。



4. 发送数据，以下测试只需在这里发送对应测试命令即可。



### 3.3 整机非信令 TX 命令(BR/EDR/LE)

如发送命令"0xAAAA0A00FD000100012614100300", 发送测试命令数据格式解析如下:

同步字 AAAA, length 0A00, 命令 FD00

发送的命令后 8bytes 为测试参数, 即"0100022614100300"

8bytes 参数含义如下:

/\* Payload(8bytes):

byte0 : bt\_mode 0 or 1 //0: BR/EDR TEST, 1: BLE TEST

byte1 : BLE\_PHY 0 or 1 //0: BLE 1M, 1: BLE 2M

byte2 : channel //tx channel (0-79), 实际发送包落在 2402Mhz+byte2 Mhz

byte3 : tx\_power //tx power (20-43), 实际发射功率(dbm)是 byte3 减去 30dbm

byte4 : tx\_mode //tx mode, DH1/DH3/DH5 等 or power and ACP test(0x14), 调制特性 00001111(0x15), 调制特性 01010101(0x16), 具体定义见 <<ACTIONS 308x BT\_TEST\_command\_list\_20230605.xlsx>>里面 TX 0x09 命令定义;

byte5 : payload\_mode //payload mode(only BR/EDR, no use by BLE, BLE set 0x10), ,



具体定义见<<ACTIONS 308x BT\_TEST\_command\_list\_20230605.xlsx>>里面 TX 0x17 命令定义:

```
byte6 ~ byte7 : test time // unit : s 测试时间, 时间到自动重启
```

```
*/
```

所以, 上述命令的测试内容是“发送 BLE 1M 包, 2404Mhz(即 channel 1), TX power 8dbm, power and ACP test, 发包持续 3 秒后重启”

### 3.4 整机非信令 RX 命令(BR/EDR/LE)

如发送命令"0xAAAA0C00FE0001000012050029417671", 发送测试命令数据格式解析如下:

同步字 AAAA, length 0C00, 命令 FE00

发送的命令后 10bytes 为测试参数, 即"01000012050029417671"

10bytes 参数含义如下:

```
/* Payload(10bytes):
```

```
* byte0 : bt_mode 0 or 1 //0: BR/EDR TEST, 1: BLE TEST
```

```
* byte1 : BLE_PHY 0 or 1 //0: BLE 1M, 1: BLE 2M
```

```
* byte2 : channel //rx channel (0-79), 实际接收包落在 2402Mhz+byte2 Mhz
```

```
* byte3 : rx_mode //rx mode (0~11, 0x10~0x12), 比如 LE set 0x12(0x01010101),
```

具体定义见<<ACTIONS 308x BT\_TEST\_command\_list\_20230605.xlsx>>里面 RX 0x01 命令定义:

```
* byte4~5 : test time // unit : s, 测试时间, 时间到自动重启
```

```
* (option) byte6~9 : 仪器低 4bytes mac, 没有就使用默认地址
```

```
*/
```

所以, 上述命令的测试内容是“接收低四位 mac 是 71:76:41:29 的 BLE 1M 包, 2402Mhz(即 channel 0), 调制特性 01010101, 收包时间持续 5 秒后重启”。

RX 测试完成会自动重启, 如果想读取测试报告, 请再次连接 LE, 然后打开 0xAA03 通知, 0xAA02 通道发送读取 RX 测试结果数据的命令"AAAA0200E100";

例如读到的结果如下图内容:

```
aa aa 15 00 e1 00 00 0d 00 00 00 00 18 8a 18 00 00 00 00 00 51 02 00 00 B4
```

其中, 收到的数据格式解析, 同步字 AAAA, length 1500, 命令 E100

Payload 有 19bytes, 即"00 0d 00 00 00 00 18 8a 18 00 00 00 00 00 51 02 00 00 B4"

Payload 含义如下:

```
/* Payload(19bytes):
```

```
* byte0 : Test result 0 or 1 //0: success, 1: fail
```

```
* byte1 : Test times, 历史测试次数
```

```
* byte2~17: 测试报告,
```

报告解读例子:

```
00 00 00 00 68 21 37 00 00 00 00 00 57 41 00 00
```

第一个 word 4byte 为错误的 bit 数量(0x00000000)

第二个 word 4byte 为接收到的 bit 数量 (0x00372168)

第三个 word 4byte 为错误的包数量 (0x00000000)

第四个 word 4byte 为接收到的包数量(0x00004157)

\* byte18: 平均 RSSI 值

\*/

也可以接串口查看 log 获取测试报告, 如下图:

```
TX: 02 01 06 20 00 1c 00 04 00 1b 3a 00 aa aa 15 00 e1
00 00 02 70 00 00 00 28 01 00 00 01 00 00 00 01
00 00 00 b4
[os][I][650]: nsm cmd 0xe1 start!!
[os][I][385]: Pre test index 2 result 0 rssi -76
[os][I][387]: Pre test report 0 = 0x70
[os][I][387]: Pre test report 1 = 0x0
[os][I][387]: Pre test report 2 = 0x0
[os][I][387]: Pre test report 3 = 0x0
[os][I][387]: Pre test report 4 = 0x28
[os][I][387]: Pre test report 5 = 0x1
[os][I][387]: Pre test report 6 = 0x0
[os][I][387]: Pre test report 7 = 0x0
[os][I][387]: Pre test report 8 = 0x1
[os][I][387]: Pre test report 9 = 0x0
[os][I][387]: Pre test report 10 = 0x0
[os][I][387]: Pre test report 11 = 0x0
[os][I][387]: Pre test report 12 = 0x1
[os][I][387]: Pre test report 13 = 0x0
[os][I][387]: Pre test report 14 = 0x0
[os][I][387]: Pre test report 15 = 0x0
```

### 3.5 整机精简 TX 命令(Only LE)

如发送命令"0xAAAA0800C000050000260000", 发送测试命令数据格式解析如下:

同步字 AAAA, length 0800, 命令 C000

发送的命令后 6bytes 为测试参数, 即"050000260000"

6bytes 参数含义如下:

/\* Payload(6bytes):

- \* byte0~1: timeout, 持续接收 TX 数据的时间, 单位秒; 0xFFFF 为不重启一直发;
- \* byte2: channel, 信道编号 range 0~39, 实际发送包落在 2402Mhz+ 2\*byte2 Mhz
- \* byte3: power, 发射功率. range 20~43, 对应-10dbm~13dbm
- \* byte4: cmd 操作码
  - \* 0x00: 测试发送功率、带内辐射
  - \* 0x01: 测试调制特性 1
  - \* 0x02: 测试调制特性 2、频偏
- \* byte5: mode BLE mode, range 0~1, 0: BLE 1M, 1: BLE 2M

\*/

所以，上述命令的测试内容是“发送 BLE 1M 包， 2402Mhz， TX power 8dbm， 测试发送功率、带内辐射， 发包持续 5 秒后重启”

### 3.6 整机精简 RX 命令(Only LE)

如发送命令"0xAAAA0A00C1000500002941767100"， 发送测试命令数据格式解析如下：

同步字 AAAA， length 0A00， 命令 C100

发送的命令后 8bytes 为测试参数， 即"0500002941767100"

8bytes 参数含义如下：

/\* Payload(8bytes):

- \* byte0~1 : timeout, 持续接收 RX 数据的时间, 单位秒; 0xFFFF 为不重启一直收;
- \* byte2 : channel, 信道编号 range 0~39, 实际发送包落在 2402Mhz+ 2\*byte2 Mhz
- \* byte3~6 : mac, 发数仪器的低四位地址, 数据长度 4bytes, 小端模式.
- \* byte7 : mode, BLE mode, range 0~1, 0: BLE 1M, 1: BLE 2M

\*/

所以，上述命令的测试内容是“接收低四位 mac 是 71:76:41:29 的 BLE 1M 包， 2402Mhz， 收包时间持续 5 秒后重启”。

RX 测试完成会自动重启， 如果想读取测试报告， 请再次连接 LE， 然后打开 0xAA03 通知， 0xAA02 通道发送读取 RX 测试结果数据的命令“AAAA0200C200”；

能读到 6bytes 的结果：

/\*TX Payload(6bytes):

- \* byte0 : test result 1 fail or 0 success
- \* byte1-4 : correct package number, 接收到正确包数量
- \* byte5 : rx rssi

\*/

### 3.7 整机 LE 低中高信道交替 TX 命令(压力测试)

发送命令"0xaaaa0200c800"， 进入 BLE TX 压测模式， 断开 BLE， 发送不可连接广播， 2402MHz(信道 0 index 37), 2426MHz(信道 12 index 38), 2480MHz(信道 39 index 39)， 每 60s 交替发送广播数据， 发包间隔 100ms；

发送方式: |信道 0 发 60s|信道 12 发 60s|信道 39 发 60s|....., 持续 2 小时后自动重启；

## 4 PCBA UART 非信令测试

### 4.1 固件生成方法(仅测试修改)

➤ 做如下修改，然后编译烧录  
application/bt\_watch/prj.conf 打开  
CONFIG\_NSM\_APP=y  
CONFIG\_MGR\_TEST\_SAMPLE=y  
将\application\bt\_watch\prebuild\bt\_production\_test\mp\_btc.bin  
拷贝到\application\bt\_watch\fs\_sdfs\sdfs\_k\mp\_btc.bin,  
然后删除\application\bt\_watch\fs\_sdfs\sdfs\_k\fcc.bin  
Zephyr/boards/arm/版型/board\_cfg.h 修改 UART 波特率为 115200  
-#define CONFIG\_UART\_0\_SPEED 115200 // 2000000

### 4.2 UART 模式测试

- 开机后，串口输入“btmgr nsm\_uart”进入测试模式;
- 测试示例，参考附件<<3085L 非信令测试示例.pdf>> 进行命令交互;
- 命令含义参考附件 ACTIONS 308x BT\_TEST\_command\_list\_20230605.xlsx



3085X非信令测试 示例.pdf



ACTIONS 308x  
BT\_TEST\_comman

## 5 测试 API 定义

- 蓝牙非信令相关代码路径.\application\bt\_watch\src\bt\_nsm;
- .\framework\bluetooth\bt\_manager\ bt\_manager\_test\_sample.c 里面有通过 shell 命令进入非信令模式的示例代码，如下：  
static int shell\_cmd\_nsm\_uart(const struct shell \*shell, size\_t argc, char \*argv[]): 进入非信令 UART 示例;  
static int shell\_cmd\_nsm\_air\_tx(const struct shell \*shell, size\_t argc, char \*argv[]): 进入非信令 TX 示例;  
static int shell\_cmd\_nsm\_air\_rx(const struct shell \*shell, size\_t argc, char \*argv[]): 进入非信令 RX 示例;
- 相关的 API 如下.\application\bt\_watch\src\bt\_nsm\nsm\_interface.h 头文件所示



nsm\_interface.h

## 炬芯科技股份有限公司

地址：珠海市唐家湾镇高新区科技四路 1 号 1#厂房一层 C 区

电话：+86-756-3392353

传真：+86-756-3392251

邮政编码：519085

网址：<http://www.actions-semi.com>

电子邮件（业务）：[mp-sales@actions-semi.com](mailto:mp-sales@actions-semi.com)

（技术支持）：[mp-cs@actions-semi.com](mailto:mp-cs@actions-semi.com)