
N32WB03x常用参数配置使用指南

简介

此文档的目的在于让使用者能够快速熟悉N32WB03x系列蓝牙SOC芯片的常用蓝牙参数配置，笔记讲解基于蓝牙透传rtdss例程进行讲解。

NATIONALS CONFIDENTIAL

目录

简介	1
目录	2
1. 设备名称修改	3
2. MAC地址配置	3
2.1 使用芯片默认的MAC地址	3
2.2 使用用户自定义MAC地址	4
3. 配置服务和特征值的UUID	4
4. 配置广播时间间隔范围	4
5. 配置连接参数	5
5.1 连接时间	5
5.2 从设备延时	5
5.3 超时时间	6
5.4 连接参数更新时间	6
6. 配置广播数据	7
6.1 默认广播模式—普通发现模式	7
6.2 beacon模式	7
7. 扫描应答数据	8
8. 发射功率配置	9
9. MUT变更	10
9.1 从机MTU修改	10
9.2 主机MTU值修改	10
历史 版本	11
声 明	12

1. 设备名称修改

```

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project: nfts
  N32WB03x
    STARTUP
    CMSIS
    FWLIB
    BLE_STACK
    BLE_PROFILE
    NS_DFU
    Crypto
    NS_LIB
    BLE_APP
      app_disc.c
      app_batt.c
      app_rdtsc.c
      app_ns_ius.c
    USER
      main.c
      app_uart.c
      app_gpio.c
    CONFIG
      app_user_config.h
      app_user_config.h
  main.c app_user_config.h app_ble.c
22  * OR PROFITS, OR BUSINESS INTERRUPTION, HOWEVER CAUSED AND ON ANY THEORY OF
23  * LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
24  * NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE.
25  * EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
26  */
27
28 /* File app_user_config.h
29 * Author: National Semiconductor Team
30 * Version: v1.0.1
31 *
32 * Copyright Copyright (c) 2019, Nations Technologies Inc. All rights reserved.
33 */
34
35 #ifndef APP_USER_CONFIG_H_
36 #define APP_USER_CONFIG_H_
37
38 #include "ns_adv_data_def.h"
39
40
41 // Device name
42 #define CUSTOM_DEVICE_NAME "NS_NUTSS" // 2、修改双引号内字符即可修改设备名称
43
44 /* ADV config */
45 #define CUSTOM_ADV_FAST_INTERVAL 160
46 #define CUSTOM_ADV_SLOW_INTERVAL 3200
47
48 #define CUSTOM_ADV_FAST_DURATION 0/30
49 #define CUSTOM_ADV_SLOW_DURATION 100
50
51 // advertise data
52 #define CUSTOM_USER_ADVVERTISE_DATA \
53     { \
54         .adv_type = ADV_TYPE_SERVICE_DATA_16BIT_UUID, \
55         .adv_uuid_device_information_service, \
56     }
57
58 // Scan response data length- maximum 31 bytes
59 #define CUSTOM_USER_ADV_SCANRESP_DATA \
60     { \
61         .adv_type_manufacturer_specific_data, \
62         .adv_if_id_nut, \
63         .scan_response_data_length, \
64     }
65
66 // Scan response data length- maximum 31 bytes
67 #define CUSTOM_USER_ADV_SCANRESP_DATA_LEN (sizeof(CUSTOM_USER_ADV_SCANRESP_DATA)-1)
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179

```

2. MAC地址配置

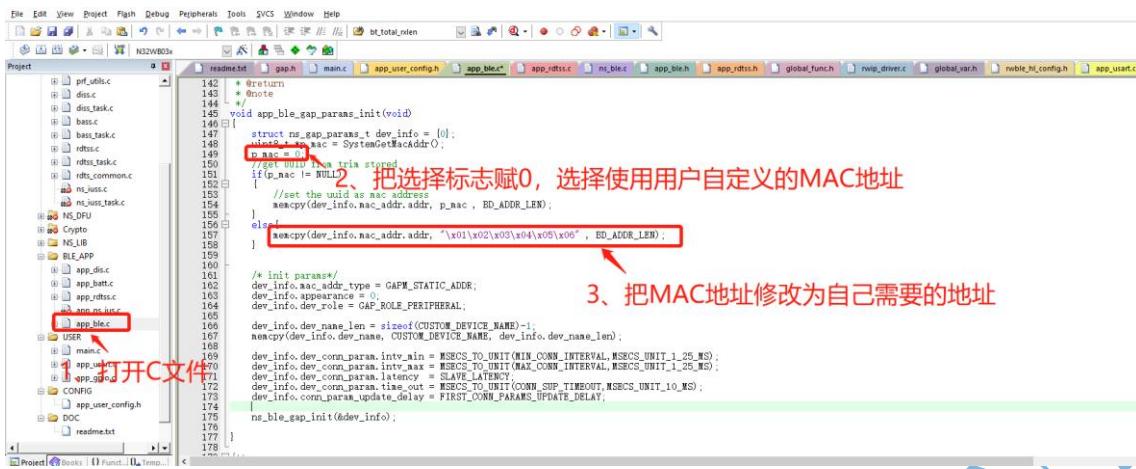
2.1 使用芯片默认的MAC地址

```

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project: nfts
  N32WB03x
    STARTUP
    CMSIS
    FWLIB
    BLE_STACK
    BLE_PROFILE
    NS_DFU
    Crypto
    NS_LIB
    BLE_APP
      app_disc.c
      app_batt.c
      app_rdtsc.c
      app_ns_ius.c
    USER
      main.c
      app_uart.c
      app_gpio.c
    CONFIG
      app_user_config.h
      app_user_config.h
  main.c app_user_config.h app_ble.c
142  /*@return
143  * @note
144  */
145 void app_ble_gap_params_init(void)
146 {
147     struct ns_gap_params_t dev_info = {0};
148     uint8_t *p_mac = SystemGetMacAddr();
149     //get UUID from firm stored
150     if(p_mac != NULL)
151     {
152         //set the uid as mac address
153         memcpy(dev_info.mac_addr.addr, p_mac, BD_ADDR_LEN); // 2、调用接口函数从芯片的ID中获取MAC地址，具有唯一性
154     }
155     else
156     {
157         memcpy(dev_info.mac_addr.addr, "\x01\x02\x03\x04\x05\x06", BD_ADDR_LEN);
158     }
159
160     /* init params*/
161     dev_info.mac_addr_type = GAFM_STATIC_ADDR;
162     dev_info.appearance = 0;
163     dev_info.dev_role = GAP_ROLE_PERIPHERAL;
164
165     dev_info.dev_name_len = sizeof(CUSTOM_DEVICE_NAME)-1;
166     memcpy(dev_info.dev_name, CUSTOM_DEVICE_NAME, dev_info.dev_name_len);
167
168     dev_info.dev_conn_param_intv_min = MSEC_TO_UNIT(MIN_CONN_INTERVAL_MSECS_UNIT_1_25_MS);
169     dev_info.dev_conn_param_intv_max = MSEC_TO_UNIT(MAX_CONN_INTERVAL_MSECS_UNIT_1_25_MS);
170     dev_info.dev_conn_param_latency = SLAVE_LATENCY;
171     dev_info.dev_conn_param_time_out = MSEC_TO_UNIT(CONN_SUP_TIMEOUT_MSECS_UNIT_10_MS);
172     dev_info.conn_param_update_delay = FIRST_CONN_PARAMS_UPDATE_DELAY;
173
174     ns_ble_gap_init(&dev_info);
175
176 }
177
178 /**
179 * @brief  ble advertising initialization

```

2.2 使用用户自定义MAC地址



```

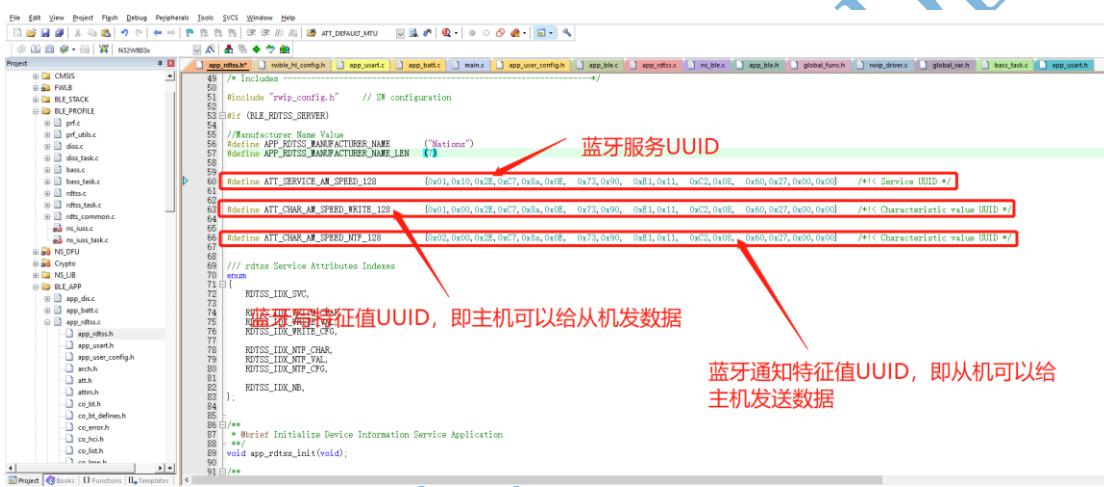
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project app_ble.c main.c app_user_config.h app_ble.h app_btsc.c ns_ble.c app_ble.h global_func.h nvip_driver.c global_var.h nble_hi_config.h app_user.c
142     #return
143     /* Note
144 */
145     void app_ble_gap_params_init(void)
146     {
147         struct nble_gap_params_t dev_info = {0};
148         uin8_t *p_mac = SystemGetMacAddress();
149         p_mac[0] = 0x02; // MAC address trim stored
150         if(p_mac != NULL)
151         {
152             //set the unit as mac address
153             memcpy(dev_info.mac_addr.addr, p_mac, BD_ADDR_LEN);
154         }
155         else
156         {
157             //memcpy(dev_info.mac_addr.addr, "\x01\x02\x03\x04\x05\x06", BD_ADDR_LEN);
158         }
159     }
160
161     /* init params*/
162     dev_info.acec_addr_type = GAPM_STATIC_ADDR;
163     dev_info.appearance = 0;
164     dev_info.dev_role = GAP_ROLE_PERIPHERAL;
165
166     dev_info.dev_name_len = sizeof(CUSTOM_DEVICE_NAME)-1;
167     memcpy(dev_info.dev_name, CUSTOM_DEVICE_NAME, dev_info.dev_name_len);
168
169     dev_info.dev_conn_params.intv_min = MSecsToUnitMinConnInterval(MSCS_UNIT_1_25_MS);
170     dev_info.dev_conn_params.intv_max = MSecsToUnitMaxConnInterval(MSCS_UNIT_1_25_MS);
171     dev_info.dev_conn_params.latency = 0;
172     dev_info.dev_conn_params.tiae_out = MSecsToUnitConnSupTimeout(MSCS_UNIT_10_MS);
173     dev_info.dev_conn_params_update_delay = FIRST_CONN_PARAMS_UPDATE_DELAY;
174
175     nble_gap_init(&dev_info);
176
177 }
178

```

2. 把选择标志赋0, 选择使用用户自定义的MAC地址

3. 把MAC地址修改为自己需要的地址

3. 配置服务和特征值的UUID



```

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project app_user.c main.c app_user_config.h app_ble.c ns_ble.c app_ble.h global_func.h nvip_driver.c global_var.h nble_hi_config.h app_user.h
1 // Includes
2
3 #include "rwp_config.h" // SW configuration
4 #include "OLE_RDTSS_SERVER"
5
6 /*Manufacturer Name Value
7 ndefine APP_RDTSS_MANUFACTURER_NAME ("Nations")
8 ndefine APP_RDTSS_MANUFACTURER_NAME_LBN {1}
9
10 #define ATT_SERVICE_AM_SPEED_128 {0x01,0x10,0x2B,0x07,0x0a,0x0E,0x73,0x90,0x01,0x01,0xC2,0x06,0x00,0x27,0x00,0x00} /*(C Service UUID */
11 #define ATT_CHAR_AM_SPEED_WRITE_128 {0x01,0x00,0x2B,0x07,0x0a,0x0E,0x73,0x90,0x01,0x01,0xC2,0x06,0x00,0x27,0x00,0x00} /*(C Characteristic value UUID */
12 #define ATT_CHAR_AM_SPEED_NTF_128 {0x02,0x00,0x2B,0x07,0x0a,0x0E,0x73,0x90,0x01,0x01,0xC2,0x06,0x00,0x27,0x00,0x00} /*(C Characteristic value UUID */
13
14 // rdtss Service Attributes Indexes
15
16 #include "RDTSS_IXC_SVC."
17
18 RDTSS_IXC_NTF_CHAR,
19 RDTSS_IXC_NTF_VAL,
20 RDTSS_IXC_NTF_CPN,
21
22 RDTSS_IXC_NTF_ND,
23
24
25 /**
26 * @brief Initialize Device Information Service Application
27 */
28 void app_rdtss_init(void);
29
30 /**
31 */
32
33
34
35
36 #ifndef _APP_USER_CONFIG_H_
37 #define _APP_USER_CONFIG_H_
38
39 #include "nble_adv_data_def.h"
40
41 /* Device name */
42 #define CUSTOM_DEVICE_NAME "NS_RDTSS"
43
44 /* ad config*/
45 #define CUSTOM_ADV_FAST_INTERVAL 160 /*** Fast advertising interval (in units of 0.625 ms). This value corresponds to 100 ms. */
46 #define CUSTOM_ADV_SLOW_INTERVAL 3200 /*** Slow advertising interval (in units of 0.625 ms. This value corresponds to 2 seconds). */
47 #define CUSTOM_ADV_FAST_DURATION 0/30 /*** The advertising duration of fast advertising in units of 1 seconds. maximum is 655 seconds */
48 #define CUSTOM_ADV_SLOW_DURATION 180 /*** The advertising duration of slow advertising in units of 1 seconds. maximum is 655 seconds */
49
50
51 // Advertising data
52 #define CUSTOM_USER_ADVVERTISE_DATA \
53     "\x00" \
54     ADV_TYPE_SERVICE_DATA_16BIT_UUID, \
55     ADV_TYPE_SERVICE_INFORMATION_SERVICE \
56
57
58 #define CUSTOM_USER_ADVVERTISE_DATA_LEN (sizeof(CUSTOM_USER_ADVVERTISE_DATA)-1)
59
60 // Scan response data
61 #define CUSTOM_USER_ADV_SCANSP_DATA \
62     "\x00" \
63     ADV_TYPE_MANUFACTURER_SPECIFIC_DATA \
64     /*Manufacturer information*/
65
66
67 // Scan response data length- maximum 31 bytes
68 #define CUSTOM_USER_ADV_SCANSP_DATA_LEN (sizeof(CUSTOM_USER_ADV_SCANSP_DATA)-1)
69
70
71 // Connection config */
72 #define MIN_CONN_INTERVAL 30 /*** Minimum connection interval (15 ms) */
73 #define MAX_CONN_INTERVAL 30 /*** Maximum connection interval (30 ms) */
74

```

蓝牙服务UUID

蓝牙写特征值UUID, 即主机可以给从机发数据

蓝牙通知特征值UUID, 即从机可以给主机发送数据

4. 配置广播时间间隔范围



```

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project app_user_config.h main.c app_user_config.h app_ble.c ns_ble.c app_ble.h global_func.h nvip_driver.c global_var.h nble_hi_config.h app_user.h
1 // Copyright Copyright (c) 2019, Nations Technologies Inc. All rights reserved.
2
3
4
5
6 #ifndef _APP_USER_CONFIG_H_
7 #define _APP_USER_CONFIG_H_
8
9 #include "nble_adv_data_def.h"
10
11 /* Device name */
12 #define CUSTOM_DEVICE_NAME "NS_RDTSS"
13
14 /* ad config*/
15 #define CUSTOM_ADV_FAST_INTERVAL 160 /*** Fast advertising interval (in units of 0.625 ms). This value corresponds to 100 ms. */
16 #define CUSTOM_ADV_SLOW_INTERVAL 3200 /*** Slow advertising interval (in units of 0.625 ms. This value corresponds to 2 seconds). */
17 #define CUSTOM_ADV_FAST_DURATION 0/30 /*** The advertising duration of fast advertising in units of 1 seconds. maximum is 655 seconds */
18 #define CUSTOM_ADV_SLOW_DURATION 180 /*** The advertising duration of slow advertising in units of 1 seconds. maximum is 655 seconds */
19
20
21 // Advertising data
22 #define CUSTOM_USER_ADVVERTISE_DATA \
23     "\x00" \
24     ADV_TYPE_SERVICE_DATA_16BIT_UUID, \
25     ADV_TYPE_SERVICE_INFORMATION_SERVICE \
26
27
28 #define CUSTOM_USER_ADVVERTISE_DATA_LEN (sizeof(CUSTOM_USER_ADVVERTISE_DATA)-1)
29
30 // Scan response data
31 #define CUSTOM_USER_ADV_SCANSP_DATA \
32     "\x00" \
33     ADV_TYPE_MANUFACTURER_SPECIFIC_DATA \
34     /*Manufacturer information*/
35
36
37 // Scan response data length- maximum 31 bytes
38 #define CUSTOM_USER_ADV_SCANSP_DATA_LEN (sizeof(CUSTOM_USER_ADV_SCANSP_DATA)-1)
39
40
41 // Connection config */
42 #define MIN_CONN_INTERVAL 30 /*** Minimum connection interval (15 ms) */
43 #define MAX_CONN_INTERVAL 30 /*** Maximum connection interval (30 ms) */
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74

```

广播时间间隔 = 160 * 0.625 = 100ms

最小时间单元为: 0.625ms

1. 打开头文件

5. 配置连接参数

5.1 连接时间

连接时间的时间范围为：7.5ms-4s (ble 协议规定的范围)

The screenshot shows the NS2WBD IDE interface with the project 'Project_01' open. The code editor displays the file `app_user_config.h`. Two annotations are present: a red box highlights the connection parameters `MIN_CONN_INTERVAL` and `MAX_CONN_INTERVAL`, with a red arrow pointing to the text "2、修改连接时间的MIN和MAX, 具体值根据主从机协商确定". A yellow box highlights the file `app_user_config.h` in the project tree, with a red arrow pointing to the text "1、打开头文件".

```
/* connection config */
#define MIN_CONN_INTERVAL 15 /* Minimum connection interval (15 ms) */
#define MAX_CONN_INTERVAL 30 /* Maximum connection interval (30 ms). */

#define SLAVE_LATENCY 0 /* Slave latency. */
#define CONN_SUP_TIMEOUT 5000 /* Connection supervisory timeout (5000ms). */

#define FIRST_CONN_PARAMS_UPDATE_DELAY (5000) /* Time of initiating event to update connection params (5 seconds). */

/* SEC_PARAM config */
#define SEC_PARAM_NO_CAPABILITIES GAP_TO_CAP_NO_INPUT_NO_OUTPUT
#define SEC_PARAM_OOB 0
#define SEC_PARAM_KEY_SIZE 16
#define SEC_PARAM_BOND 1
#define SEC_PARAM_MITM 1
#define SEC_PARAM_LESC 0
#define SEC_PARAM_KEYPRESS 0
#define SEC_PARAM_PKEY GAP_EDIST_NONE
#define SEC_PARAM_FREY GAP_EDIST_ENCKEY
#define SEC_PARAM_SEC_MODE_LEVEL GAP_NO_SEC

/* BOND config */
#define MAX_BOND_PEER 5
#define BOND_STORE_ENABLE 0
#define BOND_DBASE_ADDR 0x01020000
```

5.2 从设备延时

SLAVE_LATENCY—允许从机在没有数据要发送的情况下，跳过一定数目的主机连接事件，在这些被跳过的连接事件中，从机不必回复主机的数据包，这样可以降低功耗，**SLAVE_LATENCY = 0** 表示从机对于每一个来自主机的连接事件都进行回复，如果不回复，主机就会认为从机接收不正常。从设备延迟必须短于监控超时时间，且至少在超时时间内给从设备留出 6 次侦听的机会。从设备延迟 \leq (超时时间应 / 6 / 连接间隔) - 1，**SLAVE LATENCY** 的范围可以是 0-499。

5.3 超时时间

超时时间范围：10ms-32s，其必须要满足：超时时间应 \geq 连接间隔 \times (从设备延迟 + 1) \times 6，不管从设备延迟是多少，都应该保证至少 6 次数据发送机会。

The screenshot shows the N32VB03x IDE interface with the project 'nrtss' open. The code editor displays the file `app_user_config.h`. Two annotations are present: a red box highlights the line `#define FIRST_CONN_PARAMS_UPDATE_DELAY (5000)`, and a red arrow points from the text '2、修改超时时间' (Modify timeout) to the same line.

```
62 #define CUSTOM_USER_ADV_SCNRPD_DATA \
63     _DATA \
64     ((uint8_t){ \
65         0x0A, \
66         0x0B, \
67     }) \
68 // Scan response data length: maximum 31 bytes \
69 #define CUSTOM_USER_ADV_SCNRPD_DATA_LEN (gizeof(CUSTOM_USER_ADV_SCNRPD_DATA)-1) \
70 \
71 /* connection config */ \
72 #define MIN_CONN_INTERVAL 15 /*<< Minimum connection interval (15 ms) */ \
73 #define MAX_CONN_INTERVAL 30 /*<< Maximum connection interval (30 ms). */ \
74 \
75 #define SLAVE_LATENCY 0 /*<< Slave latency. */ \
76 \
77 #define CONN_SUP_TIMEOUT 5000 /*<< Connection supervision timeout (5000ms). */ \
78 \
79 #define FIRST_CONN_PARAMS_UPDATE_DELAY (5000) /*<< Time of initiating event to update connection params (5 seconds). */ \
80 \
81 /*sec config */ \
82 #define SEC_PARAM_IO_CAPABILITIES GAP_IO_CAP_NO_INPUT_NO_OUTPUT \
83 #define SEC_PARAM_APPL_ID 0 \
84 #define SEC_PARAM_KEY_SIZE 16 \
85 #define SEC_PARAM_BOND 1 \
86 #define SEC_PARAM_MITM 1 \
87 #define SEC_PARAM_LTKC 0 \
88 #define SEC_PARAM_PKEYPRESS 0 \
89 #define SEC_PARAM_KIDKEY \
90 #define SEC_PARAM_RKID \
91 #define SEC_PARAM_EKID \
92 #define SEC_PARAM_RKID \
93 #define SEC_PARAM_SEC_MODE_LEVEL \
94 \
95 /* bonding */ \
96 #define MAX_BOND_PEER 5 \
97 #define BOND_STORE_ENABLE 0 \
98 #define BOND_DATA_BASE_ADDR 0x01020000 \
99 \
100 /* profiles config */ \
101 #define CFG_APP_DIS 1 \
102 #define CFG_PRF_DIS 1 \
103 // #define CFG_APP_BATT 1 \
104 // #define CFG_PRF_BATT 1
```

5.4 连接参数更新时间

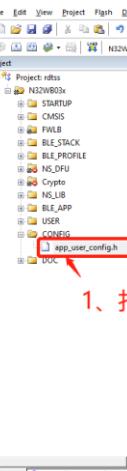
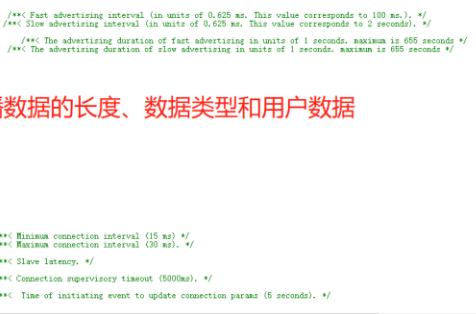
在主从机建立连接之后会进行连接参数的更新，从连接到参数更新有一段延时，这段延时就是连接参数更新时间。

```
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
BLE_RECEIVED_DATA_EVT
Project central_ntsc
  N32VB03x
    STARTUP
    CMSIS
    FWLIB
    BLE_STACK
    BLE_PROFILE
    NS_LIB
      BLE_APP
        app_drtcc.c
        app_ble.c
        app_user_config.h
      USER_APP
        main.c
        app_gpio.c
        app_usart.c
      CONFIG
        app_user_config.h
      DOC
        README.txt
main.c readme.txt app_drtcc.c app_ble.c ns_ble.c app_user_config.h
26  * ****
27  */
28  /*@file app_user_config.h
29  * @author Nations Firmware Team
30  * @version v1.0.0
31  * @copyright Copyright (c) 2019, Nations Technologies Inc. All rights reserved.
32  */
33
34 */
35
36 #ifndef _APP_USER_CONFIG_H_
37 #define _APP_USER_CONFIG_H_
38
39 /* Device name */
40 #define CUSTOM_DEVICE_NAME "NS_RUTS_CLIENT"
41
42 /* connection config */
43 #define MIN_CONN_INTERVAL 15
44 #define MAX_CONN_INTERVAL 30
45 #define SLAVE_LATENCY 0
46 #define CONN_SUP_TIMEOUT 5000
47
48 #define FIRST_CONN_PARAMS_UPDATE_DELAY (5000)
49 /* sec config */
50 #define SEC_PARAM_IO_CAPABILITIES GAP_NO_INPUT_NO_OUTPUT
51 #define SEC_PARAM_OOB 0
52 #define SEC_PARAM_KEY_SIZE 16
53 #define SEC_PARAM_BOND 1
54 #define SEC_PARAM_SECURE_CONN 0
55 #define SEC_PARAM_L2CAP 0
56 #define SEC_PARAM_PKEYPRESS 0
57 #define SEC_PARAM_KPRESS 0
58 #define SEC_PARAM_IKEY GAP_KDIST_NONE
59 #define SEC_PARAM_RKEY GAP_RDIST_NONE
60 #define SEC_PARAM_SSEC_MODE_LEVEL GAP_NO_SSEC
61
62 /* bond config */
63 #define MAX_BOND_PEER 5
64 #define BOND_STORE_ENABLE 0
65 #define BOND_DATA_BASE_ADDR 0x01020000
66
67 /**
68 * @brief Application configuration header file
69 */
70
71 /**
72 * @brief The device name of this device*
73 */
74
75 /**
76 * @brief Minimum connection interval (15 ms).
77 */
78 /**
79 * @brief Maximum connection interval (30 ms).
80 */
81 /**
82 * @brief Slave latency.
83 */
84 /**
85 * @brief Connection supervisory timeout (5000ms).
86 */
87
88 /**
89 * @brief Time of initiating event to update connection params (5 seconds).
90 */
91
92 /**
93 * @brief No I/O capabilities. (@enum gap_io_cap)
94 */
95 /**
96 * @brief Out Of Band data not available. (@enum gap_oob)
97 */
98 /**
99 * @brief Initiator Key Distribution. (@enum gap_kdist)
100 */
101 /**
102 * @brief Responder Key Distribution. (@enum gap_rdist)
103 */
104 /**
105 * @brief Device security requirements (minimum security level). (@enum see gap_sec_req)
106 */
```

6. 配置广播数据

6.1 默认广播模式—普通发现模式

最大数据长度为 31，由于蓝牙在默认情况下，蓝牙会占用 3 个字节的广播数据广播设备信息，所以用户数据只能使用 $31 - 3 = 28$ 个字节。

```

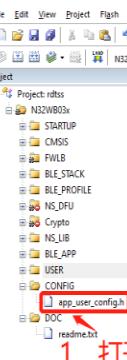
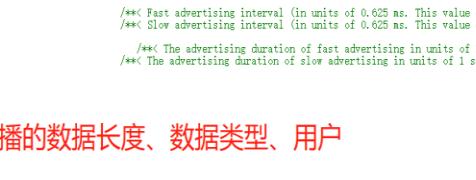
File Edit View Project Flight Debug Peripherals Tools SCS Window Help
Project ns2wb03x
  STARTUP
  CMSIS
  FWLB
  BLE_STACK
  BLE_PROFILE
  NS_DFU
  Crypto
  NS_LIB
  BLE_APP
  USER
  CONFIG
    app_user_config.h
  DOC
  README.TXT

main.c app_user_config.h app_ble.c app_rdtss.c ns_ble.h app_ble.h app_rdtss.h

34 /*
35  * Define the device name.
36 #ifndef _APP_USER_CONFIG_H_
37 #define _APP_USER_CONFIG_H_
38 #include "ns_adv_data_def.h"
39 #endif
40 /* Device name */
41 #define CUSTOM_DEVICE_NAME "NS_RDTSS"
42
43 /* Advertiser config */
44 #define CUSTOM_ADV_FAST_INTERVAL 160
45 #define CUSTOM_ADV_SLOW_INTERVAL 3200
46
47 #define CUSTOM_ADV_FAST_DURATION 0/30
48 #define CUSTOM_ADV_SLOW_DURATION 180
49
50 // Advertise data
51 #define CUSTOM_USER_ADVERTISE_DATA \
52   (ADV_TYPE_SERVICE_DATA_16BIT_UUID \
53    | ADV_TYPE_DEVICE_INFORMATION_SERVICE \
54    | CUSTOM_USER_ADVERTISE_DATA_LEN)
55
56 #define CUSTOM_USER_ADVERTISE_DATA_LEN (sizeof(CUSTOM_USER_ADVERTISE_DATA)-1)
57
58 #define CUSTOM_USER_ADV_SNRP_DATA \
59   (CUSTOM_USER_ADV_SNRP_DATA_LEN \
60    | ADV_TYPE_MANUFACTURER_SPECIFIC_DATA \
61    | ADV_TYPE_VERIFICATIONS \
62    | CUSTOM_USER_ADV_SNRP_DATA_LEN)
63
64 // Scan response data length: maximum 31 bytes
65 #define CUSTOM_USER_ADV_SNRP_DATA_LEN (sizeof(CUSTOM_USER_ADV_SNRP_DATA)-1)
66
67 /* Connection config */
68 #define MIN_CONN_INTERVAL 30
69 #define MAX_CONN_INTERVAL 30
70 #define SLAVE_LATENCY 0
71 #define CONN_SUP_TIMEOUT 5000
72 #define FIRST_CONN_PARAMS_UPDATE_DELAY (5000)
73 #define FIRST_CONN_PARAMS_UPDATE_TIMEOUT (5 seconds).
74
75 on // Local name

```

广播数据CUSTOM_USER_ADVERTISE_DATA的总长度要小于等于28。

```

File Edit View Project Flight Debug Peripherals Tools SCS Window Help
Project ns2wb03x
  STARTUP
  CMSIS
  FWLB
  BLE_STACK
  BLE_PROFILE
  NS_DFU
  Crypto
  NS_LIB
  BLE_APP
  USER
  CONFIG
    app_user_config.h
  DOC
  README.TXT

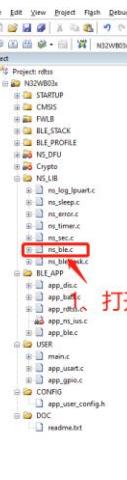
main.c app_user_config.h ns_adv_data_def.h

37 #define _APP_USER_CONFIG_H_
38
39 #include "ns_adv_data_def.h"
40
41 /* Device name */
42 #define CUSTOM_DEVICE_NAME "NS_RDTSS"
43
44 /* Advertiser config */
45 #define CUSTOM_ADV_FAST_INTERVAL 160
46 #define CUSTOM_ADV_SLOW_INTERVAL 3200
47
48 #define CUSTOM_ADV_FAST_DURATION 0/30
49 #define CUSTOM_ADV_SLOW_DURATION 180
50
51 // Advertise data
52 #define CUSTOM_USER_ADVERTISE_DATA \
53   (\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x00)
54
55 #define CUSTOM_USER_ADV_SNRP_DATA \
56   (CUSTOM_USER_ADV_SNRP_DATA_LEN \
57    | ADV_TYPE_MANUFACTURER_SPECIFIC_DATA \
58    | ADV_TYPE_VERIFICATIONS \
59    | CUSTOM_USER_ADV_SNRP_DATA_LEN)
60
61 // Scan response data length: maximum 31 bytes
62 #define CUSTOM_USER_ADV_SNRP_DATA_LEN (sizeof(CUSTOM_USER_ADV_SNRP_DATA)-1)
63
64 /* Connection config */
65 #define MIN_CONN_INTERVAL 30
66 #define MAX_CONN_INTERVAL 30
67 #define SLAVE_LATENCY 0
68 #define CONN_SUP_TIMEOUT 5000
69 #define FIRST_CONN_PARAMS_UPDATE_DELAY (5000)
70 #define FIRST_CONN_PARAMS_UPDATE_TIMEOUT (5 seconds).
71
72 on // Local name

```

6.2 beacon模式

可以去掉广播数据中被占用的02 01 06三个字节的固定广播数据，从而使广播数据可以达到31个字节




```

File Edit View Project Flight Debug Peripherals Tools SCS Window Help
Project ns2wb03x
  STARTUP
  CMSIS
  FWLB
  BLE_STACK
  BLE_PROFILE
  NS_DFU
  Crypto
  NS_LIB
  BLE_APP
  USER
  CONFIG
    app_user_config.h
  DOC
  README.TXT

main.c app_user_config.h gpm_task.h

114 static void app_create_advertising(void)
115 {
116     NS_LOG_DEBUG("Advertising...\n");
117     if (app_env.adv_state == APP_ADV_STATE_IDLE && app_env.adr_mode != APP_ADV_MODE_STOP)
118     {
119         // Prepare the GPM_ACTIVITY_CREATE_CMD message
120         struct gpm_activity_create_adv_cmd *p_cmd = KB_MSG_ALLOC(GPM_ACTIVITY_CREATE_CMD,
121                                                               TASK_GPM, TASK_APP,
122                                                               gpm_activity_create_adv_cmd);
123
124         // Set operation code
125         p_cmd->adv_param_type = GPM_CREATE_ADV_ACTIVITY;
126
127         // Fill the allocated kernel message
128         p_cmd->own_addr_type = app_env.own_addr_type; // GPM_STATIC_ADDR;
129         p_cmd->adv_param.prop = ADV_PROP_INDIRECT;
130         p_cmd->adv_param.second_cft phy = app_env.adv_phy; //PHY_IMPS_VALUE, GPM_PHY_TYPE_LR_WI;
131         p_cmd->adv_param.filter_pol = ADV_ALLOW_SCAN_ANY_CONN;
132         p_cmd->adv_param_disc_mode = GPM_ADV_MODE_BEACON; //修改为beacon 模式可以去掉广播数据中被占用的02 01 06三个字节的固定广播数据，从而使广播数据可以达到31个字节
133         p_cmd->adv_param_disc_mode = adv_env.adv_phy;
134
135         if(adv_env.ex_adv_stable)
136         {
137             p_cmd->adv_param.prop = GPM_ADV_TYPE_EXTENDED;
138             p_cmd->adv_param.prop = ADV_CONN;
139             p_cmd->adv_param.second_cft phy = adv_env.adv_phy; //PHY_IMPS_VALUE, GPM_PHY_TYPE_LR_WI;
140             p_cmd->adv_param.filter_pol = ADV_ALLOW_SCAN_ANY_CONN;
141             p_cmd->adv_param_disc_mode = 0;
142         }
143
144         switch (app_env.adr_mode)
145         {
146             case APP_ADV_MODE_DIRECTED:
147                 /* If the peripheral is already bonded with a central device, use the direct advertising
148                  procedure (BD Address of the peer device is stored in NWD.
149
150                  #if (BLE_APP_SEC)
151                      if (nme_sec_get_bond_status())
152                          p_cmd->adv_param.prop = GPM_ADV_PROP_DIR_CONN_MASK;
153                  #endif
154
155                 */
156
157         }
158     }
159 }

```

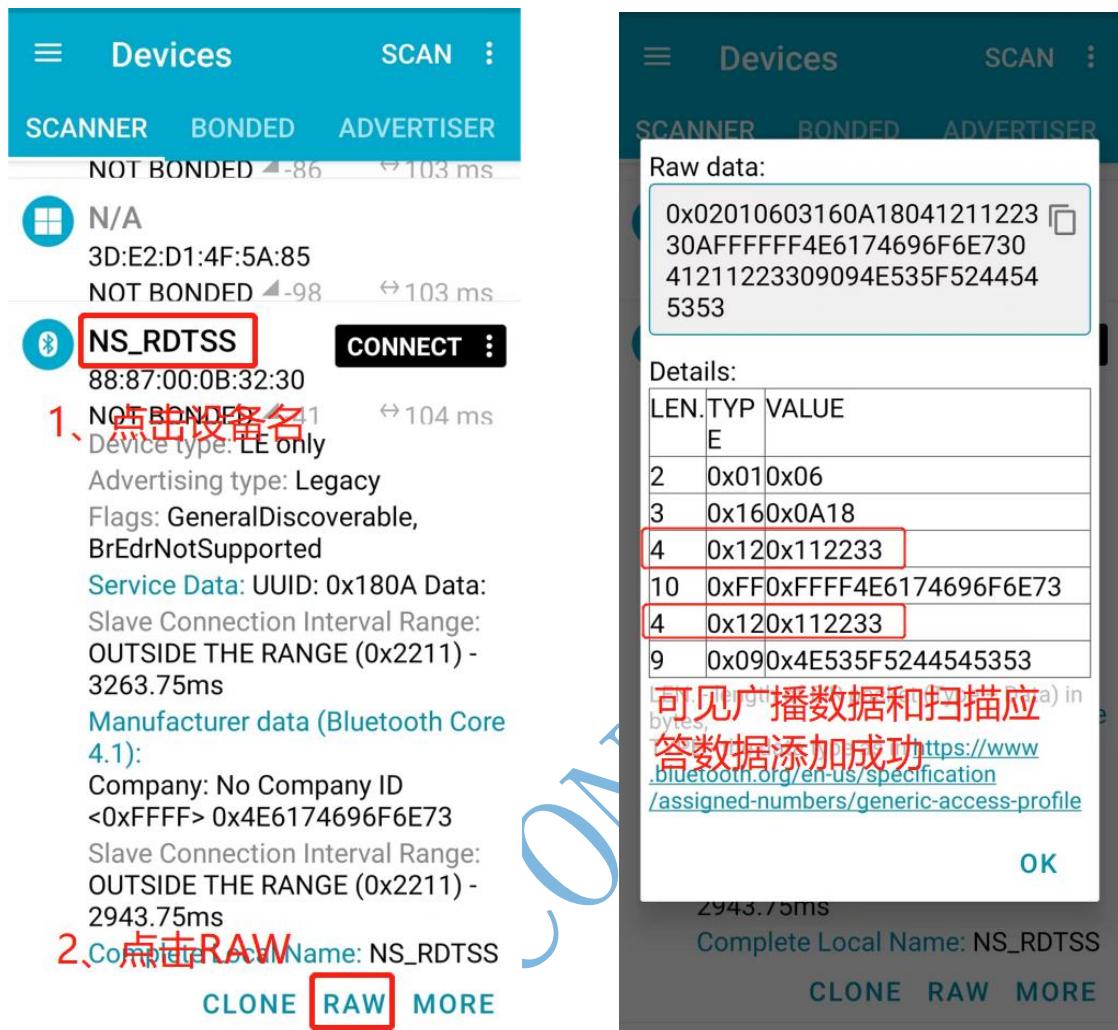
7. 扫描应答数据

最大数据长度为 31 个字节，用户可以使用数据长度为 31 个字节。

扫描应答数据CUSTOM_USER_ADV_SCNRSP_DATA长度要小于等于31个，用户可以使用31个字节长度。

The screenshot shows the Nsight Embedded Studio interface. The left pane displays the project structure for 'rdts' under 'Project: rdts'. The right pane shows the code editor with the file 'app_user_config.h' open. The code defines a manufacturer-specific data structure with a length of 3 bytes, containing fields for flags, major, minor, and UUID. A red box highlights the line containing the manufacturer-specific data definition.

添加广播数据和扫描应答数据之后，可在APP扫描到的设备名字下看到具体数据



8. 发射功率配置

The screenshot shows a software development environment with a project named 'N32WB03x'. The 'USER' folder is selected, and the 'main.c' file is open. The code is as follows:

```

64 // brief main function
65 *
66 * @param Note
67 */
68
69 #include "global_func.h"
70
71 int main(void)
72 {
73     /*for hold the SW before sleep
74     delay_n10us(200*1000);
75     */
76
77     NS_LOG_INIT_C();
78
79     if (CFG_APP_N5_JUS)
80     {
81         if(CURRENT_APP_START_ADDRESS == NS_APP1_START_ADDRESS)
82             NS_LOC_INFO("application 1 start new ... \r\n");
83         else if(CURRENT_APP_START_ADDRESS == NS_APP2_START_ADDRESS)
84             NS_LOC_INFO("application 2 start new ... \r\n");
85     }
86
87     /*endif
88     app_ble_init();
89
90     rf_tx_power_set(TR_POWER_Pos0_DEM); //设置发射功率为-16dbm
91
92     NS_LOC_INFO(DEMO_STRING);
93
94     //pinhigh init
95     LedInit(LED1_PORT, LED1_PIN); // power led
96     LedInit(LED2_PORT, LED2_PIN); //connection state
97     LedOn(LED1_PORT, LED1_PIN);
98     app_start_rx_enable(ENABLE);
99
100    //init text
101    usart_tx_da_send((uint8_t*)DEMO_STRING, sizeof(DEMO_STRING));
102
103    delay_n10us(500);
104    //disable uart for enter sleep
105
106}

```

Annotations highlight specific parts of the code:

- A red box surrounds the '#include "global_func.h"' line.
- A red box surrounds the 'rf_tx_power_set(TR_POWER_Pos0_DEM);' line.
- A green box surrounds the 'app_ble_init();' line.
- A red box surrounds the 'delay_n10us(500);' line.
- A green box surrounds the '//disable uart for enter sleep' comment.

Annotations with numbers 1, 2, and 3 point to these highlighted areas respectively.

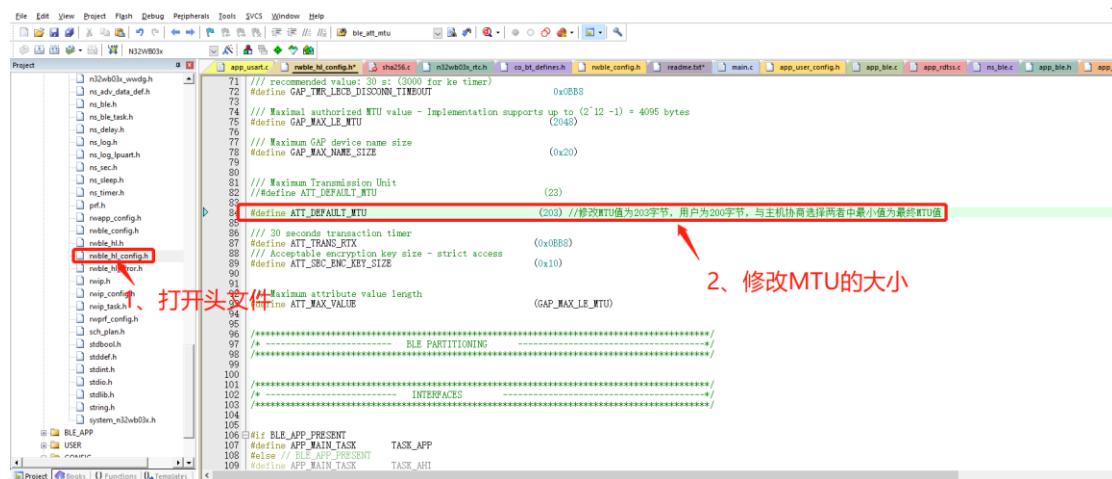
3、调用接口修改芯片的发射功率

9. MUT变更

从机修改ATT_DEFAULT_MTU值，主机也需要修改MTU值，主从协商取两者中最小的值为通信的MTU值。

9.1 从机MTU修改

从机最大支持MTU值，有效数据为MTU-3个字节，ATT的opcode占用一个字节以及ATT的handle占用2个字节，所以客户能使用的有效数据为MTU-3个字节。



```

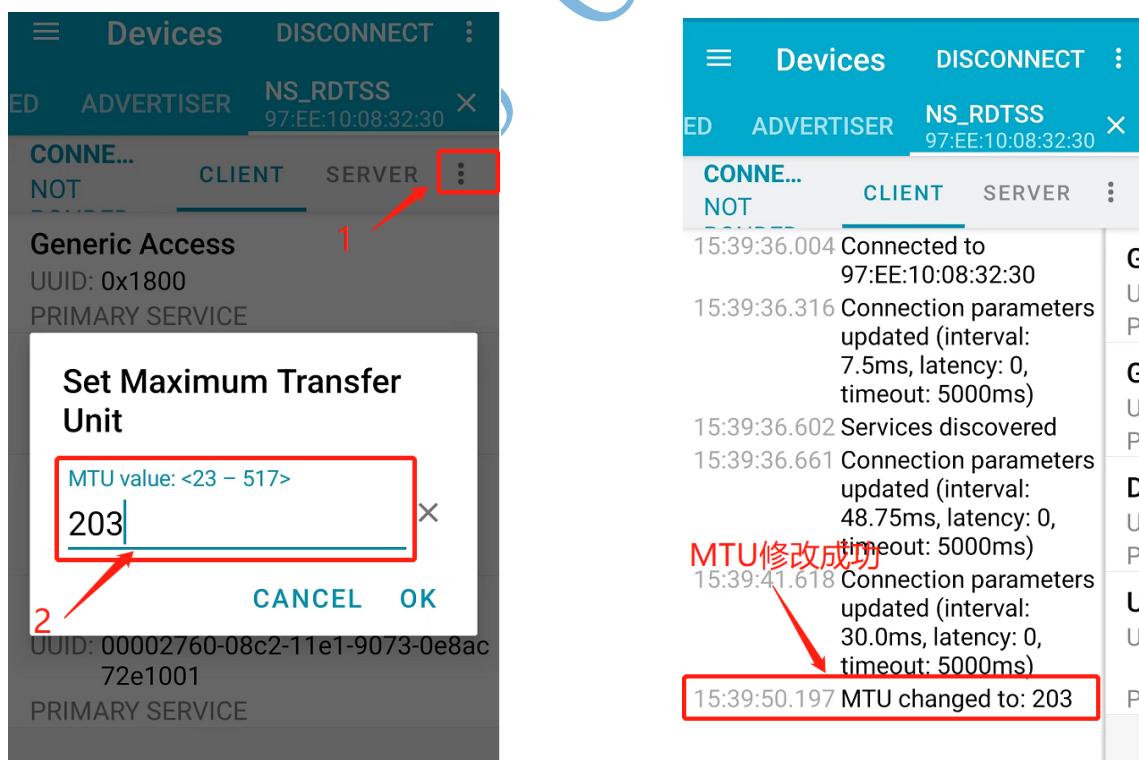
File Edit View Project Flight Debug Peripherals Tools SIVCS Window Help
Project n32wb03x
nrf_bt.h nrf_bt_config.h sha256.c nrf_bt03_rx.h os_bt_defines.h nrf_bt_config.h README.txt main.c app_user_config.h app_bt.c app_rdtss.c nrf_bt.h app_bt.h app_rdtss.h
nrf_bt.h nrf_bt_config.h sha256.c nrf_bt03_rx.h os_bt_defines.h nrf_bt_config.h README.txt main.c app_user_config.h app_bt.c app_rdtss.c nrf_bt.h app_bt.h app_rdtss.h
71 // recommended value: 30 s (3000 for ke timer)
72 #define GAP_TMR_LCDB_DISCONNECT_TIMEOUT 0x0B8
73 // Maximal authorized MTU value - Implementation supports up to (2^12 - 1) = 4095 bytes
74 #define GAP_MAX_MTU
75 // Maximum GAP device name size
76 #define GAP_MAX_NAME_SIZE (0x20)
77 // Maximum transmission unit
78 #define ATT_DEFAULT_MTU (23)
79 // Maximum attribute value length
80 #define ATT_MAX_VALUE_LENGTH (GAP_MAX_MTU)
81 // 30 seconds transaction timer
82 #define ATT_TRANS_RTX (0x0B8)
83 // Acceptable encryption key size - strict access
84 #define ATT_SEC_ENC_KEY_SIZE (0x10)
85 // Maximum attribute value length
86 #define ATT_MAX_VALUE_LENGTH (GAP_MAX_MTU)
87 // Maximum GAP device name size
88 #define GAP_MAX_NAME_SIZE (0x20)
89 // Maximum transmission unit
90 #define ATT_DEFAULT_MTU (203) //修改MTU值为203字节, 用户为200字节, 与主机协商选择两者中最小值为最终的MTU值
91 // Maximum attribute value length
92 #define ATT_MAX_VALUE_LENGTH (GAP_MAX_MTU)
93 // BLE PARTITIONING
94 // INTERFACES
95 // SERVICES
96 // APPS
97 // If BLE_APP_PRESENT
98 #define APP_MAIN_TASK TASK_APP
99 // else
100 #define APP_MAIN_TASK TASK_AHI
101 // APPS
102 // SERVICES
103 // INTERFACES
104 // APPS
105 // SERVICES
106 // If BLE_APP_PRESENT
107 #define APP_MAIN_TASK TASK_APP
108 // else
109 #define APP_MAIN_TASK TASK_AHI

```

2、修改MTU的大小

9.2 主机MTU值修改

主从协商选择两者中最小值作为通信的MTU值



历史版本

日期	版本	修改
2022.09.07	V1.0.0	初始版本

NATION'S CONFIDENTIAL

声 明

国民技术股份有限公司（以下简称国民技术）保有不事先通知而修改的权利。国民技术认为提供的信息准确可信，尽管这样，国民技术对准确性和可靠性不承担任何责任。购买前请获取器件说明的最新版本。在法律允许的最大范围内，任何明示、暗示或保证，包括但不限于适销性、特定用途适用性和第三方知识产权侵权责任，国民技术概不承担不承认。在任何情况下，国民技术均不对因使用本产品而产生的任何直接、间接、偶然、特殊、惩戒性或后果性损害负责，即使已告知可能发生此类损害。不建议应用于与生命相关的设备和系统。国民技术对本手册拥有专属产权。未经明确许可，任何人不得以任何理由对本手册的全部或部分进行使用、复制、修改、抄录、传播。

NATION'S CONFIDENTIAL