

利尔达科技集团股份有限公司

LIERDA SCIENCE & TECHNOLOGY GROUP CO., LTD

BLE_Uart_Transmission_说明



项目名称：
项目型号：
文件版本：V01
提交日期：2015-2-10

目 录

1. 简介.....1

2. 主机程序.....1

 2.1. 主体程序.....1

 2.1.1. 初始化.....1

 2.1.2. 主流程.....2

 2.1.3. BLE APP Call Back.....2

 2.2. 程序流程.....2

 2.3. 工程设置.....3

3. 从机程序.....5

 主体程序.....5

 3.1.1. 初始化.....5

 3.1.2. 主流程.....5

 3.2. 程序流程.....5

 3.3. 工程设置.....6

4. 敬告用户.....8



1. 简介

由 BLE_Uart_Transmission_Collector01 和 BLE_UART_Transmission_Server 组成一对主从机程序，实现串口透传功能。

硬件组成：基于 CY8CKIT-042-BLE Kit

BLE_Uart_Transmission_Collector01

BLE Pioneer Board + PRoC BLE Module

BLE_UART_Transmission_Server

BLE Pioneer Board + PSoC BLE Module

2. 主机程序

2.1. 主体程序

2.1.1. 初始化

```
00183:     CYBLE_API_RESULT_T apiResult;
00184:
00185:     CYBLE_LP_MODE_T lpMode;
00186:
00187:     CyGlobalIntEnable;
00188:
00189:     CommInit(); /* Start communication component */
00190:     printf("BLE Heart Rate Collector Example Project \r\n");
00191:
00192:     Scanning_LED_Write(LED_OFF);
00193:
00194:     apiResult = CyBle_Start(AppCallBack);
00195:     if (apiResult != CYBLE_ERROR_OK)
00196:     {
00197:         printf("CyBle_Start API Error: %xd \r\n", apiResult);
00198:     }
00199:     else
00200:     {
00201:         printf("CyBle_Start API ok \r\n");
00202:     }
00203:
00204:     /* Enable the Interrupt component connected to interrupt */
00205:     TC_CC_ISR_StartEx(InterruptHandler);
00206:
00207:     /* Start the components */
00208:     Timer_Start();
00209:
```

2.1.2. 主流程

```

00231: }
00232: /*****
00233:  * Wait for connection established with Central device
00234:  *****/
00235: if(CyBle_GetState() == CYBLE_STATE_CONNECTED)
00236: {
00237:     /*****
00238:     * Periodically measure a battery level and temperature and send
00239:     * results to the Client
00240:     *****/
00241:     CommMonitorUart();
00242:     CommMonitorBLE();
00243: }
00244: #if 0

```

```

00189: }
00190: /*****
00191:  /// \brief      CommMonitorUart
00192:  /// \param      none
00193:  /// \return      none
00194:  /// \deprecated monitor uart send use ble
00195:  *****/
00196: void CommMonitorUart(void)
00197: {

```

主机 BLE 发送是用 GattcWriteWithoutResponse

```

00169: /*****
00170:  /// \brief      MonitorBLE
00171:  /// \param      none
00172:  /// \return      none
00173:  /// \deprecated monitor ble send use uart tx
00174:  *****/
00175: void CommMonitorBLE(void)
00176: {

```

2.1.3. BLE APP Call Back

2.2. 程序流程

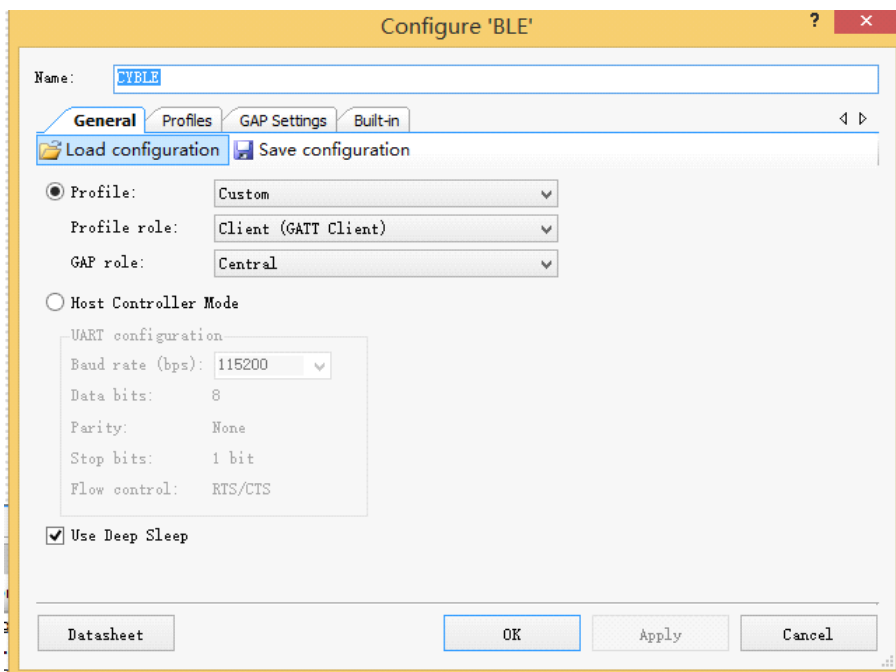
```

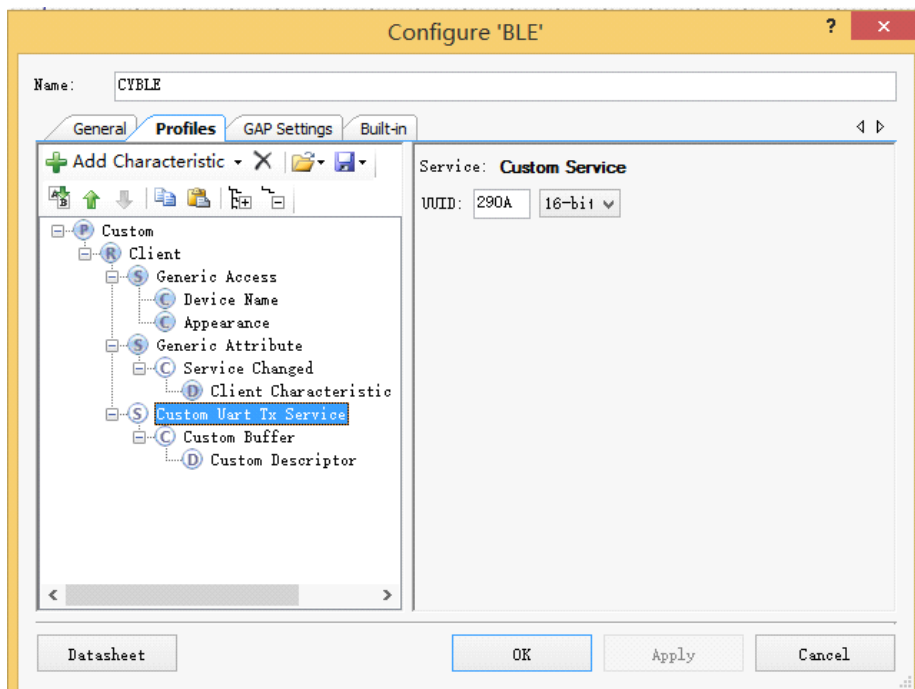
BLE Heart Rate Collector Example Project
CyBle_Start API ok 1
EVT_STACK_ON 2
Start Scan
EVT_GAPC_SCAN_START_STOP 3
SCAN_PROGRESS_RESULT: peerAddrType = 0, peerBdAddr = 0: 00a050000006, rssi = -39 dBm, data = 02 01 06 03 03 0a 29 newDevice = 2
This device contains Stop Scanning, waiting for Scanning event
SCAN_PROGRESS_RESULT: peerAddrType = 0, peerBdAddr = 0: 00a050000006, rssi = -39 dBm, data = 02 08 55 03 03 0a 29 newDevice = 2
This device contains Stop Scanning, waiting for Scanning event
EVT_GAPC_SCAN_START_STOP 5
Connect to the Device: 0
EVT_GATT_CONNECT_IND 6
EVT_GAP_DEVICE_CONNECTED 6
zozo test exchg mtu req send 7
CYBLE_EVT_GATTC_XCHNG_MTU_RSP

```

- 1 CyBle_Start(AppCallBack)
- 2 AppCallBack : CYBLE_EVT_STACK_ON: StartScan(CYBLE_UUID_CUSTOM_UART_TX_SERVICE)
- 3 AppCallBack : CYBLE_EVT_GAPC_SCAN_START_STOP (SCAN_START)
- 4 AppCallBack : CYBLE_EVT_GAPC_SCAN_PROGRESS_RESULT : ScanProgressEventHandler
发现已知设备 (UUID = CYBLE_UUID_CUSTOM_UART_TX_SERVICE)
- 5 AppCallBack : CYBLE_EVT_GAPC_SCAN_START_STOP (SCAN_STOP)
发起连接请求 CyBle_GapcConnectDevice
- 6 AppCallBack : CYBLE_EVT_GAP_DEVICE_CONNECTED
变更大数据包请求 512B CyBle_GattcExchangeMtuReq(cyBle_connHandle,0x200)
- 7 AppCallBack : CYBLE_EVT_GATTC_XCHNG_MTU_RSP
变更成功

2.3. 工程设置





其他默认即可。

lierda®
利尔达科技集团

3. 从机程序

3.1. 主体程序

3.1.1. 初始化

同主机

3.1.2. 主流程

同主机

```
00191: //////////////////////////////////////
00192: /// \brief      MonitorUart
00193: /// \param      none
00194: /// \return     none
00195: /// \deprecated send to ble tx
00196: //////////////////////////////////////
00197: void CommMonitorUart(void)
00198: {
00199:     CYBLE_API_RESULT_T apiResult;
00200:     CYBLE_GATT_HANDLE_VALUE_PAIR_T CustomNotificationhandle;
00201:
00202:     if(uCommState.Bit.UartRxFinished == ENABLED)
00203:     {
00204:         uCommState.Bit.UartRxFinished = DISABLED;
00205:
00206:         uCommState.Bit.BLERxRDY = DISABLED;
00207:
00208:         CustomNotificationhandle.attrHandle = UART_TX_HANDLE;
00209:         CustomNotificationhandle.value.val = Buffer;
00210:         CustomNotificationhandle.value.len = Buffer[0]+1;|
00211:     #if 1
00212:
00213:         /* Send notification to client using previously filled structure */
00214:         apiResult = CyBle_GattsNotification(cyBle_connHandle, &CustomNotificationhandle);
00215:     #else
00216:
00217:         apiResult = CyBle_GattcWriteWithoutResponse(cyBle_connHandle, &CustomNotificationhandle);
00218:     #endif
00219: }
```

从机 BLE 发送是用 GattsNotification

3.2. 程序流程

```
BLE Uart Transmission Server Example Project
EVT_STACK_ON 1
Start Advertisement with addr: 00a050000006
EVT_GAPP_ADVERTISEMENT_START_STOP 2
EVT_GATT_CONNECT_IND 3
EVT_GAP_DEVICE_CONNECTED: connIntv = 7500 us 4
EVT_GATTS_XCNHG_MTU_REQ 5
```

1 AppCallBack : CYBLE_EVT_STACK_ON: StartAdvertisement()

2 AppCallBack : CYBLE_EVT_GAPC_SCAN_START_STOP (SCAN_START)

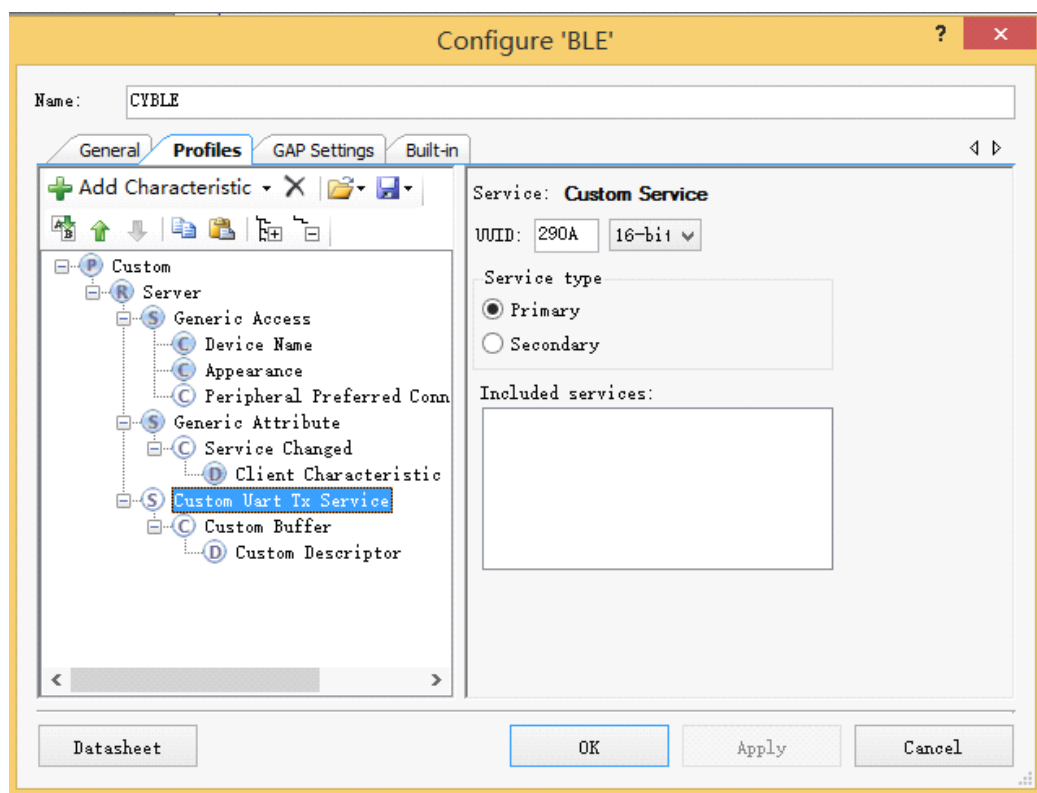
3&4 AppCallBack : CYBLE_EVT_GAP_DEVICE_CONNECTED

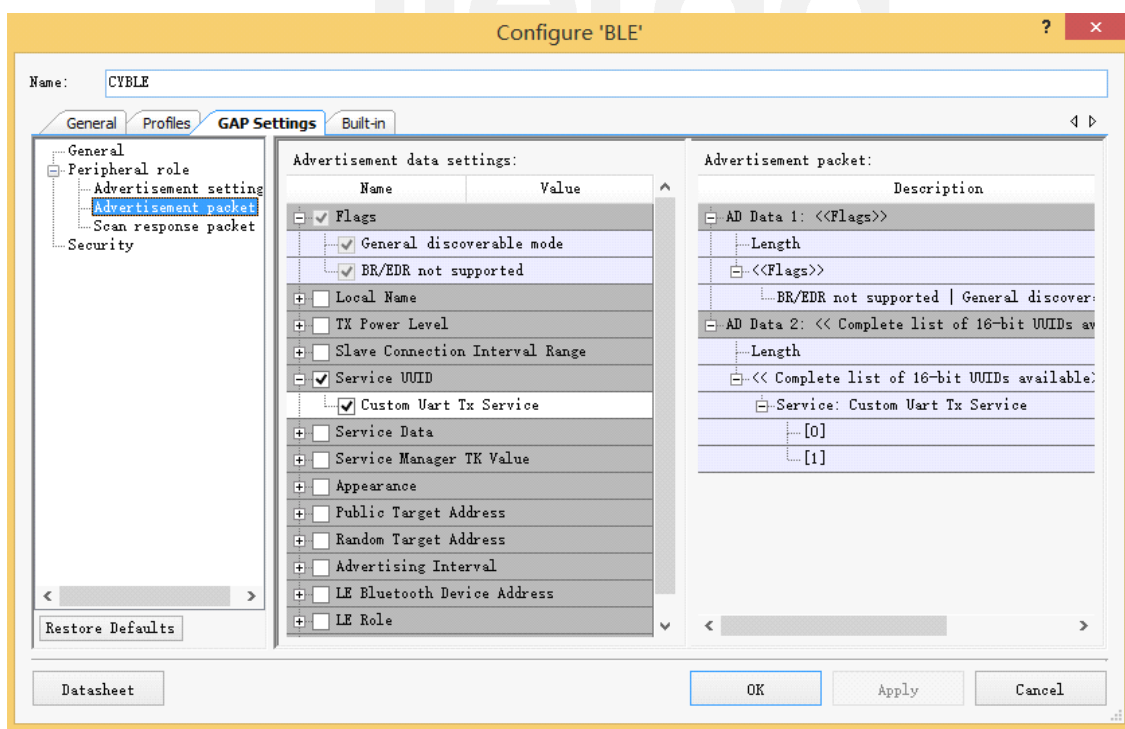
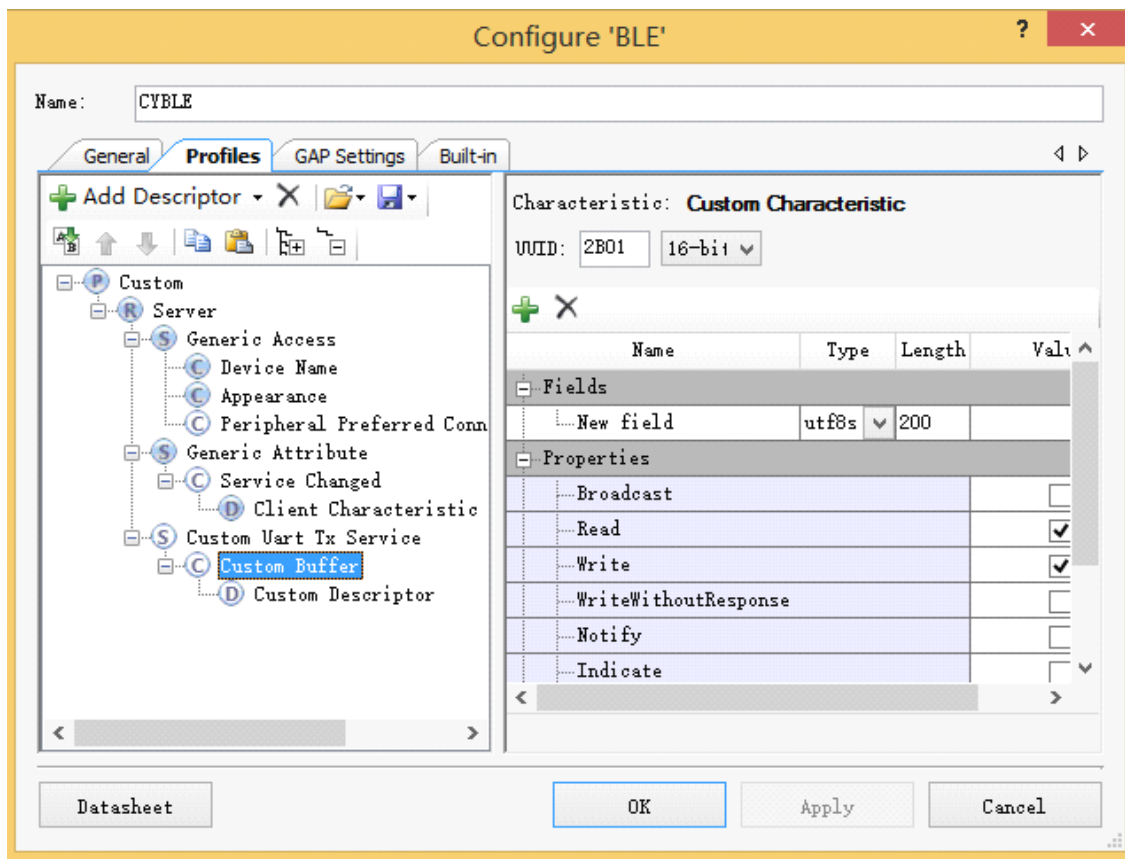
连接间隔 为 7.5ms

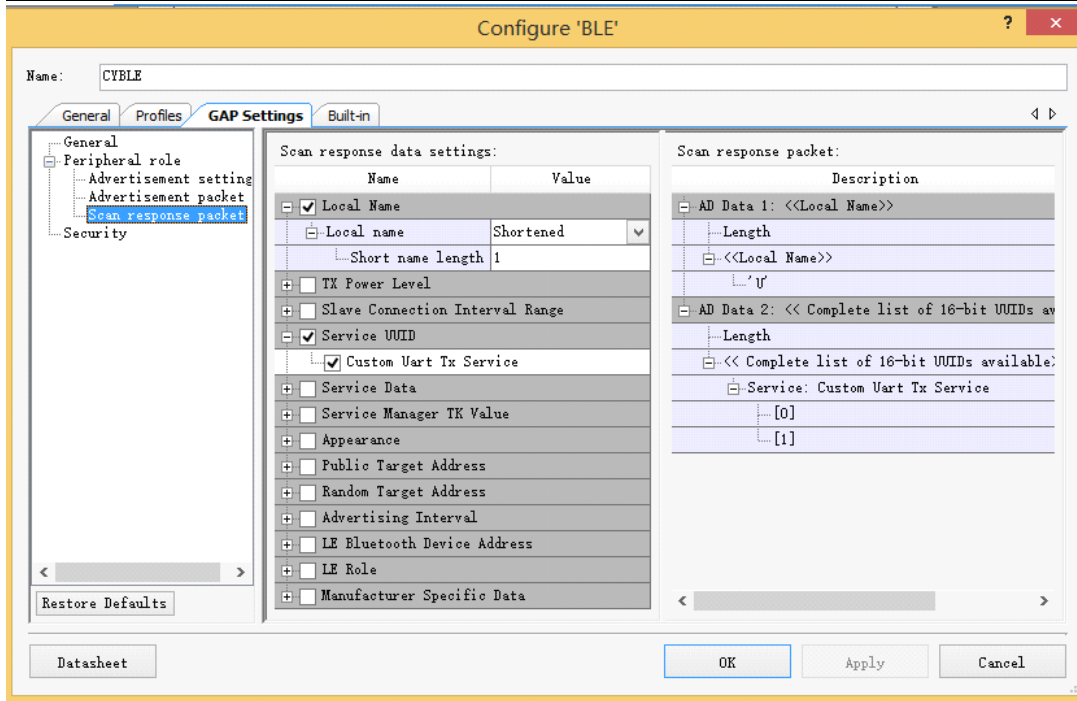
5 AppCallBack : CYBLE_EVT_GATTC_XCHNG_MTU_RSP

变更成功

3.3. 工程设置







4. 敬告用户

- 1、欢迎您使用利尔达科技集团股份有限公司的产品，在使用我公司产品前，请先阅读此敬告；如果您已开始使用说明您已阅读并接受本敬告。
- 2、利尔达科技集团股份有限公司保留所配备的全部资料的最终解释和修改权，如有更改恕不另行通知。

利尔达科技集团

利尔达科技集团股份有限公司

2015年 2月