

# Coex 评估测试

使用手册

版本: 1.0

版权 @ 2020

www.bouffalolab.com

### Contents

1	准备	f	4
2	烧录	ξ	5
	2.1	连接	5
	2.2	软件下载	6
	2.3	串口工具配置	7
3	测试	代评估一	9
	3.1	wifi ping + ble adv	9
4	测试	【准备	12
5	测试	代评估二	13
	5.1	wifi ping + ble connect(1s 发送 5 次数据包,长度为 23byte) .............	13
6	测试	【评估三	19
	6.1	wifi running iperf + ble connect(1s 发送 5 次数据包,长度为 23byte) ........	19

# List of Figures

正面	5
背面	6
烧写工具界面	7
串口工具	8
模块成功连接 WiFi	9
模块开启 ping	10
APP 扫描到 BLE	14
BLE 连接成功	15
PC 端 lperf 开启 sever 模式 ...................................	20
Ble 开启 ADV	21
APP 扫描到 BLE	22
BLE 连接成功	23
	正面          背面          烧写工具界面          串口工具          模块成功连接 WiFi          模块开启 ping          APP 扫描到 BLE          BLE 连接成功          PC 端 lperf 开启 sever 模式          APP 扫描到 BLE          Ble 开启 ADV          APP 扫描到 BLE          BLE 连接成功

1

1. 硬件:芯片模块一个,Windows PC 一台,USB 转串口线一根。

2. 软件: 烧写工具, 烧录 bl602\_demo\_event.bin 文件, 路径: bouffalolab\_release\_bl\_iot\_sdk.zip/App\_Demos/bl602\_ demo\_event/build\_out/bl602\_demo\_event.bin, 选择任意一款串口工具

3. 手机上下载任意一款蓝牙调试 APP.



## 2.1 连接

芯片模块的相关引脚连接如下图所示,其中图 1 是模块的正面图,其标号 1 处用跳线帽短接,标号 2 处将左边两根排 针短接,标号 3 处将上面的两根排针短接;图 2 是模块的背面图,烧录时将 IO8 和 HI 两根排针短接,烧录完成后将 IO8 和 LOW 两根排针短接并重新上电。用 USB 线连接 PC 和模块,此时模块上的电源灯常亮,表明模块通电正常。



图 2.1: 正面





图 2.2: 背面

## 2.2 软件下载

打开解压后文件中的烧写工具 flash\_tool 目录,双击 BLDevCube.exe, chip type 选择对应的芯片类型,打开后界面参数参考下图配置:



Bouffalo Lab Dev Cube 1.3.4 - BL602/604

· □ >

View Help	0			
Simple Flas	sher			
	1			2
		Firmware Config		
Interface	Uart $\lor$	Factory Params	Z:\bouffalolab_r	release_bl_iot_sdk_1.5.12-5-g6a078b47_bins\flash_tool\bl602\device_tree\bl_factory_params_loTKitA_40M.dts
COM Port	COM6 ~	Partition Table	Browse	_iot_sdk_1.5.12-5-g6a078b47_bins\flash_tool\bl602\partition\partition_cfg_2M.toml
		Boot2 Bin	Browse	hl jot sdk 1.5.12.5.g6a078b47 bins/flash tool/bl602/builtin imgs/blsn boot2 bin
Uart Rate	2000000	U DOOLE DIN	biowsc	
JLink Rate	1000	Firmware Bin	Browse	j-g6a078b47_bins\App_Demos\bl602_demo_event\build_out\bl602_demo_event.bin
Board	IoTKitA ~	🗌 Media	Browse	
Xtal	40M ~	Romfs	Browse	
Chin Frase	Falce	MFG Bin	Browse	
cinp cluse	Taise *		Koy (16 Putor)	IV (16 Puter)
	Refresh	Single Download C	onfig	Browse
	cicui			UT OT LC
		Success		Log Create & Download Open U
	· · ·			
hed				
caled by ho	st: 80/a8/c5f/c182/5208cb89d5	eu i ast/4debtc9ce3t7cb58	suaaetc16d/de4ea	au
sha256/52	07			
xip readsh	a time cost(ms): 15.8591308593	75		
hed	a and cost(ma). 19/0991000990	-		
- caled by de	v: 867a87c5f7c18275208cb89d5	e01a3f74debfc9ce3f7cb58	0aaefc16d7de4ea	a0

图 2.3: 烧写工具界面

其中图 3 的框 1 中 COM Port 选项根据实际串口情况选择(右击我的电脑-> 管理-> 设备管理器-> 端口,查看端口号, 模块是双串口,选择端口号较小的),框 2 中的相关路径依据实际情况选择。配置完成后点击 Download 按钮下载。

### 2.3 串口工具配置

将 IO8 和 LOW 两根排针短接并重新上电,打开串口工具,设置对应的端口号,波特率设定为 2000000 bps。



🕵 PuTTY Configuration		?	×
Putty Configuration         Category:         □ Session         □ Logging         □ Terminal         □ Keyboard         □ Rell         □ Features         □ Window         □ Appearance         □ Behaviour         □ Translation         □ Colours         □ Colours         □ Proxy         □ Telnet         □ Proxy         □ Telnet         □ SSH         □ Serial	Basic options for your PuTTY set         Specify the destination you want to connect         Serial line         COM3         Connection type:         O Raw       Telnet         O Raw       Telnet         Load, save or delete a stored session         Saved Sessions         Default Settings         Close window on exit:         O Always       Never         Image: Only on close	? ssion ct to Speed 2000000 I • Seria Load Save Delete ean exit	×
About Help	Open	Cancel	

图 2.4: 串口工具

3 测试评估—

#### 3.1 wifi ping + ble adv

重启板子, chip 作为 client, PC 作为 server, APP 以 nRF Master Control Panel / nRF Connect 为例

- 1. router ssid: bl\_test\_081, passwd: 12345678
- 2. 在串口中运行 wifi 相关命令:

#stack\_wifi

#wifi\_sta\_connect bl\_test\_081 12345678 (连接成功后会获取 IP 地址)

[lwip] netif status callback IP: 192.168.8.193 MK: 255.255.255.0 GW: 192.168.8.1 [WF][SM] Exiting wifiConnected\_ipObtaining state [WF][SM] IP GOT IP:192.168.8.193, MASK: 255.255.255.0, Gateway: 192.168.8.1, dns1: 192.16 8.8.1, dns2: 0.0.0.0 [WF][SM] State Action ###wifiConnected\_ipObtaining### --->>> ###wifiConnected\_IPOK### [WF][SM] Entering wifiConnected\_IPOK state [APP] [EVT] GOT IP 24583 [SYS] Memory left is 132664 Bytes

图 3.1: 模块成功连接 WiFi

3. 在 PC 的 cmd 界面运行命令: \$ping 192.168.81.103 -t (默认 1s ping 一次,192.168.81.103 是设备端获取的 IP 地址)



C:∖Us C:\Us	sers\paul> sers\paul>ping 1	192, 168, 3	81.103 -1	t		
正左	$D_{122}^{*} = 100 160 0^{\circ}$	1 102 目:	古 20 空話	劳动粉捉.		
莱茵	192. 168. 81. 103	的回复:	字节=32		TTL=255	
米目来自	192. 168. 81. 103 192. 168. 81. 103	的回复: 的回复:	字节=32 字节=32	मन्(8)=4ms 8न् 8]=2ms	TTL=255 TTL=255	
来自	192.168.81.103	的回复: 的回复:	字节=32 字节=32	ਸ਼ੀ(0)=3πs ਸ਼ੀ(0)=3πs	TTL=255 TTL=255	
来目	192. 168. 81. 103	的回复:	字节=32		TTL=255	
来自来自	192. 168. 81. 103	的回复: 的回复:	子卫=32 字节=32	ចុំ(២)=3ms អូវ៉ី(២)=2ms	TTL=255 TTL=255	
来自 来自	192. 168. 81. 103 192. 168. 81. 103	的回复: 的回复:	字节=32 字节=32	时间=2ms 时间=6ms	TTL=255 TTL=255	
来自	192.168.81.103	的回复:	字节=32 字节=32	धर्ने(ध)=3ms धर्ने(ध)=3ms	TTL=255	
来目	192. 168. 81. 103	的回复:	字节=32	时间=5ms	TTL=255	
米目来自	192. 168. 81. 103 192. 168. 81. 103	的回复: 的回复:	子节=32 字节=32	_मन्(म)=3ms  मन्(म)=26ms	TTL=255 s TTL=255	

图 3.2: 模块开启 ping

4. 在串口中运行 ble 相关命令:

#### #stack\_ble

#ble\_init

#ble\_start\_adv 0 0 0xa0 0xa0 (发起 adv,interval 为 100ms)

Advertising started				
<pre># [ 4064829][[][32mINF0 ]][0m: b1_sec.c: 126] random number is 7ed9292f [ 4065655][][32mINF0 ]][0m: b1_sec.c: 126] random number is 8730dad4 [ 4066496][][32mINF0 ]][0m: b1_sec.c: 126] random number is 1d81e61b [ 4067362][][32mINF0 ]][0m: b1_sec.c: 126] random number is b00a72db [ 4068204][][32mINF0 ]][0m: b1_sec.c: 126] random number is 9b23dbc6 [ 4069035][][32mINF0 ]][0m: b1_sec.c: 126] random number is 017665d8 [ 4069887][][32mINF0 ]][0m: b1_sec.c: 126] random number is 0e7aabc0</pre>				
proc_hellow_entry: RISC-V rv32imafc [ 4070710][[[32mINF0 ]][0m: b1 sec.c: 126] random number is a53d5ec9				

5. 手机端打开 APP, 查看是否扫描到设备 BL602-BLE-DEV, 如果扫描到说明测试成功:



28.all 58.all 🗟 🎗	a 🕲 🗖	Ö	\$ <b>101 69</b> 1 4:09	)
≡ De	evices	STOP SCA	NNING	
SCANNER	BONDED	ADVE	RTISER	
No filter				-
NUT			7 043 1118	
C N/A 78:67 NOT	7:E4:35:26:DA BONDED -	C 67 dBm 🗧	ONNECT →276 ms	3
<b>3 N/A</b> 5A:A8 NOT	8:92:86:48:29 BONDED -	C 87 dBm 🗧	ONNECT →267 ms	3
8 BL60 18:B9 NOT	D2-BLE-DEV 9:05:DE:97:CA BONDED	<b>C</b> 28 dBm <del>«</del>	ONNECT →108 ms	3
	E:29:E6:08:0E			



1.PC 与路由器通过有线连接。



## 5.1 wifi ping + ble connect(1s 发送 5 次数据包,长度为 23byte)

- 重启板子, chip 作为 client, PC 作为 server, APP 以 BLE 调试助手为例。
- 1-3. 步骤如同测试评估一中所示:
- 4. 在串口中运行 ble 相关命令:
  - #stack\_ble
  - #ble\_init

#ble\_start\_adv 0 0 0xa0 0xa0 (发起 adv,interval 为 100ms)

5. 打开 APP, 扫描到 BL602-BLE-DEV 并连接:



≡	Devices	STOP	SCANNING	:
SCA	NNER BONE	DED 4	DVERTISER	<b>BL</b> 18
No filt	er			*
0	<b>N/A</b> 75:86:20:15:04:B1 NOT BONDED	<b>-</b> 91 dBr	$\mathbf{n}$ $\leftrightarrow$ N/A	
8	BL602-BLE-DEV 18:B9:05:DE:96:E0 NOT BONDED	-55 dBr	OPEN TA n ↔ 48 ms	В
0	<b>N/A</b> 0B:56:5A:EC:B5:51 NOT BONDED	-79 dBr	n ⇔103 ms	
8	N/A EC:FA:5C:58:C5:CB NOT BONDED	-79 dBr	n ↔202 ms	
0	<b>N/A</b> 3A:4E:DE:8E:FD:2B NOT BONDED	-66 dBr	$n \leftrightarrow N/A$	
	N/A (4:05:40:05:00:05 图 5.1: AP	P 扫描到 BLI	≣	



≡ Dev	vices		DISCONNECT	:
BONDED	ADVER	TISER	BL602-BLE-DE\ 18:B9:05:DE:96:E0	′ ×
CONNECTED NOT BONDED		CLIENT	SERVER	•
Generic Att UUID: 0x1801 PRIMARY SER	<b>ribute</b>			
Generic Acc UUID: 0x1800 PRIMARY SER	Cess ) RVICE			
Unknown So UUID: 07af27 PRIMARY SER	ervice a6-9c22-7 RVICE	11ea-9afe	e-02fcdc4e7412	

图 5.2: BLE 连接成功

#### 6. 在串口中运行 ble 相关命令:

#ble\_conn\_update 0x6 0x6 0x0 0x1f4 (连接 interval 为 7.5ms)

7. 在串口中查看连接参数已更新:



proc\_hellow\_entry: RISC-V rv32imafc pa 37447893d, ce trk 7.09, action: capcode 46 -> 45 ble\_conn\_update 0x6 0x6 0x0 0x1f4 [btsnoop]:opcode =[0x2013],1en =[0xe], data=[000006000060000000f40100000000] [btsnoop]:Stop [btsnoop]:pkt\_type =[0x3],1en =[0x4], data=[00011320] [btsnoop]:Stop conn update initiated # [btsnoop]:pkt\_type =[0x4],1en =[0xa], data=[03000000060000000f401] [btsnoop].Stop LE conn param updated: int 0x0006 1at 0 to 500 pa 420707550, ce trk 0.74, action: capcode 45 -> 44

8. 在 APP 中找到服务特性,并且写入相应的数据:



÷	BL602-BLE-DEV	断开连接	:
	Services	Log	
G > u P	eneric Attribute UID: 00001801-0000-1000-8000 RIMARY SERVICE	-00805f9b34fb	
G > U P	eneric Access UID: 00001800-0000-1000-8000 RIMARY SERVICE	-00805f9b34fb	
V V P	nknown Service UID: 07af27a6-9c22-11ea-9afe-0 RIMARY SERVICE	)2fcdc4e7412	
	Unknown Characteristic UUID: 07af27a7-9c22-11ea-9a 4e7412 Properties:NOTIFY Descriptors: Client Characteristic Configura UUID: 00002902-0000-1000- 805f9b34fb	fe-02fcdc ation -8000-00	Ŧ
	<b>Unknown Characteristic</b> UUID: 07af27a8-9c22-11ea-9a 4e7412 Properties:	fe-02fcdc	



	10 e ",111 ".111 🕤 & 🔀	(C) ×	\$ <b>101 B2</b> 1 2:17
	← BL602-BLE-	DEV 断	开连接 🚦
	Services	Lo	
	Generic Attribute		_
	Write value ○ 单次发送		
/	<ul> <li>连续发送</li> <li>文件发送</li> </ul>		
	发送:2254 Byte 实时速度:69 B/s	HEX	
	发送间隔 _ 200 _ r	ns	
	12345678901234 90123456789012	5678901234 3456 <mark></mark>	45678
		清空	停止

9. 查看 wifi 与 ble 是否稳定连接

## 6.1 wifi running iperf + ble connect(1s 发送 5 次数据包,长度为 23byte)

重启板子, chip 作为 client, PC 作为 server, APP 以 BLE 调试助手为例。

1-2. 步骤如同测试评估一中所示:

3. 在串口中运行命令: \$ipc 192.168.81.101 (192.168.81.101 是 PC 的 IP 地址)

MK: 255.255.255.0
GW: 192.168.81.1
<pre>[WF][SM] Exiting wifiConnected_ipObtaining state [WF][SM] IP GOT IP:192.168.81.105, MASK: 255.255.255.0, Gatewa [WF][SM] State Action ###wifiConnected_ipObtaining###&gt;&gt;&gt; # [WF][SM] Entering wifiConnected_IPOK state [APP] [EVT] GOT IP 23503 [SYS] Memory left is 85632 Bytes proc_bellow_entry: RISC-V rv32imafc ipc 192.168.81.101</pre>
# Connect to iperf server successful! 5.0898(5.0898 5.0898 5.0898) Mbps! push back

4. 在 PC 的 cmd 界面运行命令: \$iperf.exe -s -i 1



):\Paul\tools\iperf>iperf -s -i1	
Sorror listoning on TCP nort 5001	
TCP window cize: 64 0 KBwte (default)	
window Size. 04.0 Mbyte (defadit)	
[380] local 192.168.81.101 port 5001 -	connected with 192.168.81.103 port 51649
[ID] Interval Transfer Ban	dwidth
[380] 0.0- 1.0 sec 442 KBytes 3.6	2 Mbits/sec
[380] 1.0-2.0 sec 223 KBytes 1.83	3 Mbits/sec
[380] 2.0-3.0 sec 523 KBytes 4.2	3 Mbits/sec
[380] 3.0- 4.0 sec 538 KBytes 4.4	1 Mbits/sec
[380] 4.0-5.0 sec 548 KBytes 4.4	9 Mbits/sec
[380] 5.0-6.0 sec 574 KBytes 4.70	) Mbits/sec
[380] 6.0-7.0 sec 559 KBytes 4.53	3 Mbits/sec
[380] 7.0- 8.0 sec 559 KBytes 4.53	3 Mbits/sec
[380] 8.0- 9.0 sec 524 KBytes 4.24	9 Mbits/sec
[380] 9.0-10.0 sec 553 KBytes 4.53	3 Mbits/sec
[380] 10.0-11.0 sec 533 KBytes 4.3	7 Mbits/sec
[380] 11.0-12.0 sec 5.08 KBytes 41.0	ð Kbits/sec
[380] 12.0-13.0 sec 452 KBytes 3.70	) Mbits/sec
[380] 13.0-14.0 sec 498 KBytes 4.03	3 Mbits/sec
[380] 14.0-15.0 sec 533 KBytes 4.3	7 Mbits/sec
[380] 15.0-16.0 sec 574 KBytes 4.7	l Mbits/sec
[380] 16.0-17.0 sec 147 KBytes 1.20	) Mbits/sec

图 6.1: PC 端 Iperf 开启 sever 模式

5. 在串口中运行 ble 相关命令:

#stack\_ble

#ble\_init

#ble\_start\_adv 0 0 0xa0 0xa0 (发起 adv,interval 为 100ms)



# pa 300603924d, ce trk 6.04, action: capcode 50 -> 49 2.5021(2.2100 26.2022 4.8002) Mbps!
ple_start_adv 0 0 0xa0 0xa0
adv_type 0x0
tmp 0x0
interval min OxaO
interval max OxaO
Advertising started
# [ 305646][[][32mINFO ][0m: b1_sec.c: 126] random number is 983611b2
$\sim 204900104$ as the E 70 set on and 40 $\sim 40$
pa 304269910u, CE LIK 5.76, action. capcoue 49 7/ 46
[ 308171][L][32m1NFU L][0m: bl_sec.c: 126] random number is 58d8b249
[
4.5207(2.2109_22.0388_4.8902) Mbps!
L 309852][L][32mINFO L][0m: b1_sec.c: 126] random number is 67a500b0
proc_hellow_entry: RISC-V rv32imafc
[ 310689][[][32mINFO ][0m: b1_sec.c: 126] random number is 507c4047
pa 307669003d, ce trk 5.43, action: capcode 48 -> 47
[ 311526][[[32mINFO [][0m: b] sec.c: 126] random number is 984f5937
$\begin{bmatrix} 312368 \end{bmatrix} \begin{bmatrix} \boxed{32m} \\ \boxed{312368} \end{bmatrix} \begin{bmatrix} \boxed{32m} \\ \boxed{32m} \\ \boxed{32m} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \boxed{32m} \\ \boxed{32m} \\ \boxed{32m} \end{bmatrix} \begin{bmatrix} \boxed{32m} \\ \boxed{32m} \\ \boxed{32m} \\ \boxed{32m} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \boxed{32m} \\ 32m$
$[$ 312189][ $\Box$ [32mINF0] $\Box$ [0m; b1 sec.c: 126] random number is 3f238cb7
$\begin{bmatrix} 214022 \end{bmatrix} \begin{bmatrix} 0.2m \text{INFO} \\ 0.2m \text{INFO} \end{bmatrix} \begin{bmatrix} 0.m \text{ b} \\ 0.2m \text{ c} \end{bmatrix} \begin{bmatrix} 0.2m \text{INFO} \\ 0.2m \text{ c} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\ 0.2m \text{ c} \end{bmatrix} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 0.2m \text{ c} \\$
L SIAUSSILLISZMINTO LIUM. DI_SEC.C. IZOJ TANUOM NUMBER IS 9DEUDU4D
push back

图 6.2: Ble 开启 ADV

6. 手机打开 APP, 扫描到设备 BL602-BLE-DEV 并连接:



≡	Devices	STOP	SCANNING	:
SCA	NNER BON	DED	ADVERTISER	<b>BL</b> 18
No filt	ter			Ŧ
0	<b>N/A</b> 75:86:20:15:04:B1 NOT BONDED	<b>-</b> 91 dB	$m \leftrightarrow N/A$	
8	BL602-BLE-DE 18:B9:05:DE:96:E0 NOT BONDED	/ ) -55 dB	OPEN TA m ↔ 48 ms	В
0	N/A 0B:56:5A:EC:B5:5 NOT BONDED	1 -79 dB	m ⇔103 ms	
8	N/A EC:FA:5C:58:C5:C NOT BONDED	B -79 dB	m ↔202 ms	
•	N/A 3A:4E:DE:8E:FD:2 NOT BONDED	B 📕 -66 dB	$m \leftrightarrow N/A$	
	N/A (4:05:40:05:00:22 图 6.3: Al	- PP 扫描到 BL	E	



≡ Dev	ices	DISCONNECT	:
BONDED	ADVERTISER	BL602-BLE-DEV 18:B9:05:DE:96:E0	×
CONNECTED NOT BONDED	CLIE	NT SERVER	:
Generic Att UUID: 0x1801 PRIMARY SER	<b>ribute</b> VICE		
Generic Acc UUID: 0x1800 PRIMARY SER	vice		
Unknown Se UUID: 07af27a PRIMARY SER	e <b>rvice</b> a6-9c22-11ea-9a VICE	afe-02fcdc4e7412	

图 6.4: BLE 连接成功

7. 连接成功后,在串口中运行 ble 连接参数更新命令:

#ble\_conn\_update 0x28 0x28 0x0 0x1f4 (连接 interval 为 50ms)



2.6620(2.2109 5.2423 4.8902) Mbps!	
conn update initiated	
# LE conn param updated: int 0x0028 lat 0 to 500	
proc hellow entry: RISC-V rv32imafc	
push back	

8. 在 APP 中找到服务特性,并且写入相应的数据:



÷	BL602-BLE-DEV	断开连接	:
	Services	Log	
Ge > UU PR	eneric Attribute IID: 00001801-0000-1000-8000- RIMARY SERVICE	00805f9b34fb	
Ge > uu PR	eneric Access IID: 00001800-0000-1000-8000- IMARY SERVICE	00805f9b34fb	
Ur V UU PR	nknown Service IID: 07af27a6-9c22-11ea-9afe-0 IMARY SERVICE	2fcdc4e7412	
	Unknown Characteristic UUID: 07af27a7-9c22-11ea-9af 4e7412 Properties:NOTIFY Descriptors: Client Characteristic Configura UUID: 00002902-0000-1000- 805f9b34fb	fe-02fcdc tion 8000-00	Ŧ
	Unknown Characteristic UUID: 07af27a8-9c22-11ea-9af 4e7412 Properties:	fe-02fcdc	



	🔀 & 🤶 🕪 🛍	(3) \$ \$ 1□1 182□ 2:17
	← BL602-BLE-	DEV 断开连接 :
	Services	Log
,	Generic Attribute	
	Write value ○ 单次发送	
`	<ul> <li>         连续发送         <ul> <li>             文件发送         </li> </ul> </li> </ul>	
	发送:2254 Byte 实时速度:69 B/s	HEX
	发送间隔 200 n 12345678901234 901234567890123	ns 56789012345678 3456
		清空  停止

9. 查看 ble 是否稳定连接, iperf 的速率是否正常