SPP for Dongle 使用说明

目　　录

安装驱动程序.....2
默认串口设置.....3
串口指令.....3
　　指令格式.....3
　　指令束约.....4
具体指令说明.....4
操作示例.....9
错误操作码.....10

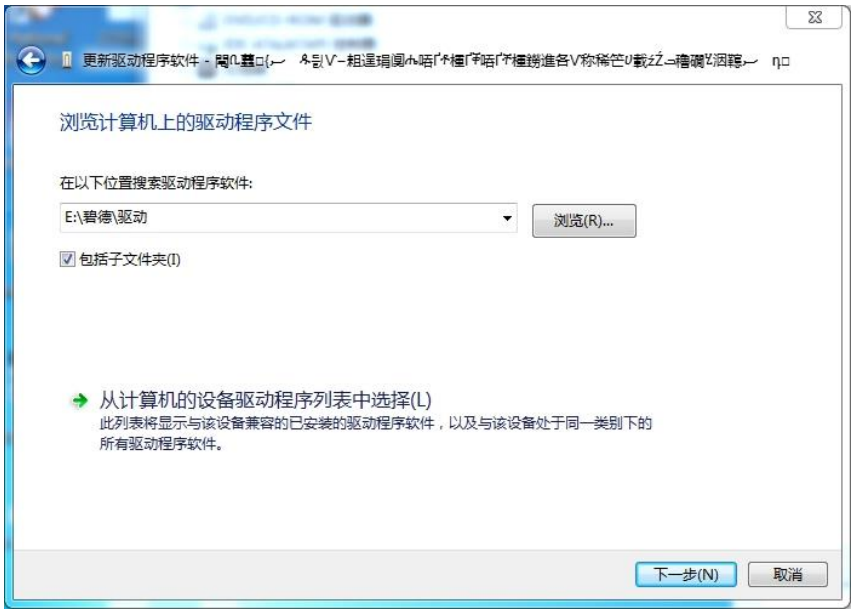
安装驱动程序

第一次使用 Dongle 时，需要安装驱动。安装驱动的步骤如下：

1. 在设备管理器找到未识别的 Dongle，右键，在弹出的菜单中选择“更新驱动程序软件”



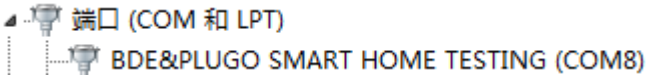
2. 选择“浏览计算机以查找驱动程序软件”



选择“bde spp for dongle.inf”文件所在的文件夹，点击下一步



弹出安全警告，选择“始终安装此驱动程序软件“。等待驱动安装完成。



Dongle 便会被识别为一个 COM 口。

默认串口设置

Dongle 的默认串口设置为：115200bps,8bit 数据位，1bit 停止位，没有校验位。

串口指令

用户可以通过串口指令控制 Dongle 的行为，指令采用 ASCII 码格式。

指令格式

指令由四部分组成，具体如下：

指令开始标志	指令码	参数列表	指令结束标志

指令开始标志为：”SPP:”这四个连续的字符。

指令码代表不同的操作：

指令码	指令作用
setBR	设置 Dongle 波特率
getBR	获取 Dongle 波特率
setDBM	设置发射功率
getDBM	获取发射功率
setScan	开启/关闭扫描
connect	连接指定的从设备

disconnect	断开连接
setConnInt	设置连接参数
getConnInt	获取连接参数
getStatus	查询 Dongle 运行状态
sendData	发送指定长度的透传数据
getVersion	获取 Dongle 固件版本

参数列表：参数列表根据不同的指令而不同，用于指定指令的具体参数。具体的参数请参考具体指令说明。

指令结束结束标志：由三个字符组成，分别为'\r'回车符，'\n'换行符和'\0'空字符。

指令的各部分由空格来区分，但指令结束标志与最后一个参数之间不需要空格。空格的个数没有严格的规定。但需要保证"SPP:"和"\r\n\0"之间的字节个数不超过 50 个字节。但为了执行速度，建议在需要使用空格的地方只用一个空格符来填充。下面是获取 Dongle 固件版本的指令：

"SPP: getVersion\r\n\0"

又比如设置连接参数的指令：

"SPP: setConnInt 10 20 0 100\r\n\0"

这个例子中的参数有 4 个，均以空格隔开。

指令束约

串口指令有三个约束，这三个约束用于 Dongle 从串口数据流里区分正确的指令和透传数据。三个约束如下：

a) 一条完整的指令应该要连续的发送到 Dongle，或保证一条完整的指令的发送时间不超过 100ms。

b) "SPP:"和"\r\n\0"之间的数据不能超过 50 个字节。

c) 指令码必须为正确的已定义了的指令码。如：

"SPP: getversion\r\n\0"

以上数据包会被理解为透传数据，因为 getversion(v 没有大写)不是正确的指令码。

具体指令说明

- setBR

作用：设置 Dongle 的波特率

参数个数：1 个

具体参数：

参数(波特率)	含义
9600	设置 Dongle 波特率为 9600bps

19200	设置 Dongle 波特率为 19200bps
38400	设置 Dongle 波特率为 38400bps
57600	设置 Dongle 波特率为 57600bps
115200	设置 Dongle 波特率为 115200bps

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## getBR

作用：获取模式当前波特率

参数个数：无

响应：

“SPP: ok 115200\r\n\0”(具体参数和 Dongle 当前波特率相关)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## setDBM

作用：设置 Dongle 的发射功率

参数个数：1 个

具体参数：

参数(发射功率)	含义
0	设置 Dongle 的发射功率为 0dbm
4	设置 Dongle 的发射功率为 4dbm
-6	设置 Dongle 的发射功率为-6dbm
-23	设置 Dongle 的发射功率为-23dbm

说明：

默认值为 0dbm。Dongle 只支持上表中四种功率值。

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## getDBM

作用：获取 Dongle 的发射功率

参数个数：无

响应：

“SPP: ok n\r\n\0”(n 表示具体的功率值)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- setScan

作用：开启或关闭扫描

参数个数：1 个

具体参数：

参数(动作)	含义
on	开启扫描
off	关闭扫描

说明：

当开启扫描成功后，Dongle 会先响应 ok 指令，当 Dongle 扫描到从设备时，会响应 dev 指令。

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: dev addr 87:BC:D6:13:33:88 public name =bde spp dev\r\n\0”

“SPP: dev addr 87:BC:D6:13:33:88 public dat =0123456789\r\n\0”

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- connect

作用：连接指定的从设备

参数个数：1 个

具体参数：

参数 1(从设备地址)	参数 2(地址类型)	含义
87:BC:D6:13:33:88	static	连接地址为 87:BC:D6:13:33:88，地址类型为 static 的从设备

说明：

地址以 ASCII 码表示，每两个地址字节间用’:’分隔。地址类型有四种：

地址类型	说明
public	Use the BD_ADDR
static	Static address
prinon	Generate Non-Resolvable Private Address
prires	Generate Resolvable Private Address

Dongle 的 dev 响应中会带有地址类型，用户必须根据这个地址类型来连接从设备。如：

收到 dev 响应 “SPP: dev addr 87:BC:D6:13:33:88 public dat =\r\n\0”

连接该设备的指令为

“SPP: connect 87:BC:D6:13:33:88 public\r\n\0”

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## disconnect

作用：断开连接

参数个数：无

说明：

如果 Dongle 是主动的断开连接，断开连接成功后，Dongle 会回到空闲状态。如果 Dongle 是被动的断开连接，即由从设备断开。Dongle 会自动扫描搜索上次连接的设备，并自动建立连接，而这时的扫描是没有响应的。因此当 Dongle 被动的动开连接，需要连接新的设备时，应先用 setScan off 来关闭扫描，再 setScan on 来开启扫描，这时便会重新响应 dev 指令。

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## setConnInt

作用：设置连接参数

参数个数：4 个

具体参数：

参数 1(最小连接间隔 单位:1.25ms)	参数 2(最大连接间隔 单位:1.25ms)	参数 3(latency)	参数 4(连接超时 单位:10ms)	含义
min	max	lat	timeout	设置最小连接间隔为 min*1.25ms，设置最大连接间隔为 max*1.25ms，设置 latency 为 lat，设置连接超时为 timeout*10ms

说明：

最小连接间隔和最大连接间隔的取值范围均为：0x0006~0x0C80，而且最大连接间隔不能小于最小连接间隔。

latency 取值范围为：0x0000~0x03E8。

连接超时的取值范围为：0x000A~0x0C80。

用户设置的连接间隔必须小于连接超时时间。即：[max*1.25 + (1 + latency)] < 10 * timeout。

连接间隔越大，功耗就越低，不过传输速率也会越低。

更新连接参数需要另一方的接受，因此，有时会存在延迟几秒才响应的现象。当更新连接参数成功后，Dongle 会响应 ok 指令，当更新成败或更新超时时，Dongle 会响应 err 指令，超时时间为 10s。由于 Dongle 指令的执行具有单步性，因此在得到响应之前再发送其他的指令，都会返回 err，因此，当用户发送了更新连接参数指令后，应该要等待接收到相应的响应后才执行其他操作。

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## getConnInt

作用：获取连接参数

参数个数：无

响应：

“SPP: ok interval latency timeout\r\n\0”(interval 为实际的连接间隔，timeout 为连接超时)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## getStatus

作用：获取 Dongle 的运行状态

参数个数：无

说明：

Dongle 的运行状态有五种：空闲、有响应的扫描、无响应的扫描、正在连接和已经连接。对应的值为”idle”、”scan rsp”、”scan norsp”、”connecting”、”connected”。 ”scan norsp”和”scan rsp”的区别是：前者是 Dongle 被动的断开了连接(即由从设备断开连接或由于其他因素断开连接)，Dongle 会尝试重新连接上次连接的从设备，因此 Dongle 会开启扫描，搜索上次连接的从设备，但此时的扫描内容是不会回传的。后者是用户向 Dongle 发送了 setScan 指令，主动的开启了扫描，此时的扫描内容会上传给用户。

响应：

“SPP: ok idle\r\n\0”(Dongle 处于空闲状态)

“SPP: ok connected\r\n\0”(Dongle 处于连接状态)

“SPP: ok scan rsp\r\n\0”(带响应扫描)

“SPP: ok scan norsp\r\n\0”(不带响应扫描)

“SPP: ok connecting\r\n\0”(正在连接)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ## sendData

作用：发送指定长度的透传数据

参数个数：1 个

指令格式：

“SPP: sendData 10\r\n\0123456789”

“sendData”与”\r\n\0”之间的”10”代表需要发送的透传数据为 10 个字节，即”\r\n\0”后的 10 个字节。当 Dongle 接收到该指令时，将不会对”\r\n\0”后指定长度的数据进行指令分析，而是将这些数据全部认为是透传数据。因此可以利用这条指令来发送指令数据包。如下：

“SPP: sendData 18\r\n\0SPP: setBR 9600\r\n\0”

响应：

“SPP: ok\r\n\0”(操作成功)

“SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- getVersion

  作用：获取 Dongle 固件版本

  参数个数：无

  响应：

  “SPP: ok v1.1.0 -D\r\n\0”(当前固件版本为 v1.1.0，-D 表示设备为 Dongle 设备)

  “SPP: err reason\r\n\0” (操作失败，reason 的具体值请查看操作错误码)

- ok

  作用：操作成功响应

  说明：

  ok 指令表示操作成功，参数根据操作的不同而不同。

- err

  作用：操作失败响应

  说明：

  err 指令表示操作失败，参数见操作错误码

- dev

  作用：扫描响应

  说明：

  dev 指令包含从设备的相关信息，包括两种类型，一种是名字，另一种是数据。如下：

  “SPP: dev addr FF:11:22:33:66:FF static name =bde spp dev\r\n\0”(名字信息)

  “SPP: dev addr FF:11:22:33:66:FF static dat =0123456789\r\n\0”(数据信息)

操作示例

1. 将 Dongle 插入电脑后，用串口软件打开串口，设置波特率为 115200bps。
2. 发送 setScan 指令，开启扫描：
   “SPP: setScan on\r\n\0”
   接收到
   “SPP: ok\r\n\0”

表示开启扫描成功。如果 Dongle 附近有 SPP 从设备在广播，则会响应 dev 指令：
“SPP: dev addr FF:22:33:44:55:FF static dat=\r\n\0”

3. 接收到 dev 响应后，发送 connect 指令连接从设备：
   “SPP: connect FF:22:33:44:55:FF static\r\n\0”
   Dongle 返回
   “SPP: ok\r\n\0”
   但此时并不一定已经连接成功，Dongle 正在发起连接。
4. 多次发送 getStatus 指令来查看 Dongle 的状态:
   “SPP: getStatus\r\n\0”
   当返回：
   “SPP: ok connected\r\n\0”
   时，表明模块已经连接成功。
5. 成功建立连接后，便可以传输数据。
6. 发送 disconnect 指令断开连接：
   “SPP: disconnect\r\n\0”
   断开成功后，返回：
   “SPP: ok\r\n\0”
7. 再次发送 setScan on 指令来扫描新的 SPP 从设备。

错误操作码

操作错误码是 err 响应的参数，当用户发送的指示不被 Dongle 接受时会返回 err 响应，示例如下：
“SPP: err procNotDone\r\n\0”
具体的值和含义如下表所示：

value	significance
“procNotDone”	上次操作还没有完成
“invalidParam”	指令的参数不合法
“notConn”	Dongle 还没有连接
“alreadyConn”	Dongle 已经处于连接状态
“procFailure”	操作失败