
com_rfid_uhf_Device.DLL 动态连接库使用手册
V1.0

目录

目录.....	2
1.介绍.....	3
2.应用程序集成.....	4
3.编程接口.....	5
3.1 连接/关闭读写器.....	5
3.1.1 OpenComPort【连接指定串口】.....	5
3.1.2 CloseSpecComPort【关闭指定串口】.....	6
3.1.3 OpenNetPort【打开 TCP 连接】.....	7
3.1.4 CloseNetPort【关闭 TCP 连接】.....	8
3.2. 18000-6C 标准命令.....	9
3.2.1. Inventory_G2【查询命令】.....	9
3.2.2. InventoryMix_G2【混合查询命令】.....	11
3.2.3. ReadData_G2【读数据命令】.....	13
3.2.4. WriteData_G2【写数据命令】.....	14
3.2.5. WriteEPC_G2【广播写 EPC 号命令】.....	15
3.2.6. KillTag_G2【销毁标签命令】.....	16
3.2.7. Lock_G2【设定存储区读写保护状态命令】.....	17
3.2.8. BlockErase_G2【块擦除命令】.....	19
3.3. 读写器自定义命令.....	20
3.3.1. GetReaderInformation【获取读写器信息命令】.....	20
3.3.2 SetAddress【设置读写器地址命令】.....	21
3.3.3 SetInventoryScanTime【查询命令最大响应时间设置命令】.....	22
3.3.4. SetRfPower【功率设置命令】.....	23
3.3.5. SetRegion【工作频率设置命令】.....	24
3.3.6. SetBaudRate【波特率设置命令】.....	25
3.3.7. SetAntennaMultiplexing【天线配置命令】.....	26
3.3.8. WriteRfPower【设置写功率命令】.....	27
3.3.9. ReadRfPower【读取写功率命令】.....	28
3.3.10. RetryTimes【写重复次数设置命令】.....	29
3.3.11. SetReadMode【实时模式设置命令】.....	30
3.3.12. SetCheckAnt【天线检测命令】.....	31
3.3.13. GetSeriaNo【获取读写器序列号命令】.....	32
3.3.14. SetBeepNotification【蜂鸣器设置命令】.....	33
3.3.15. SetDRM【设置 DRM】.....	34
3.3.16. GetDRM【读取 DRM】.....	35
3.3.17. GetReaderTemperature【读取温度】.....	36
3.3.18. MeasureReturnLoss【测量回损】.....	37
附录 1. 返回值错误代码表.....	38

1.介绍

该开发包支持 java 的开发。目前只支持 windows 操作系统（分 32 位和 64 位）。

上位机应用程序通过 com_rfid_uhf_Device.DLL 操作 R2000 系列 18000-6C 格式电子标签读写器，支持上位机应用程序同时在多个通讯端口打开多个读写器。

R2000 系列电子标签读写器包包括 UHFReader288MP、UHFReader288M、UHFReader82 等型号。

2. 应用程序集成

该开发包由以下个文件组成：

文件	使用
com_rfid_uhf_Device.dll	DLL 动态链接库, 包含所有 API 接口
UHFREADER288MP.dll	DLL 动态链接库, 由 com_rfid_uhf_Device.dll 调用
Device.java	Java 工程源文件

编程语言	说明
Java	com_rfid_uhf_Device.dll, UHFREADER288MP.dll 拷贝到 java 的工程目录, 或直接拷贝到 C:\WINDOWS\system32。

3.编程接口

3.1 连接/关闭读写器

3.1.1 OpenComPort【连接指定串口】

定义	public native int OpenComPort(int port,byte[]comAddr,byte baud,int[]PortHandle);		
说明	打开与读写器连接的串口。		
参数	名称	类型	备注
	port	Int	与读写器连接的串口号。1-COM1;2-COM2…
	ComAdr	byte[]	输入读写器地址, 如果在不清楚真实地址的情况下, 可以输入广播地址 0xFF, 函数调用成功后, 此参数将返回真实地址。
	baud	byte	串口通讯波特率。0-9600 ; 1-19200 ; 2-38400 ; 5-57600 ; 6-115200。
	PortHandle	Int[]	返回与读写器连接的句柄, 此后所有 API 操作都需要此句柄。
返回 (int)	成功 : 0 ; 失败 : 非 0 ; (查看返回值错误代码表)		
参考代码	<pre>int Port = 1;//com1 byte[]comAddr=new byte[1]; comAddr[0]=(byte)255; byte baud=5;//57600bps int[] PortHandle= new int[1]; int result = OpenComPort(Port, comAddr, baud, PortHandle); if(result==0) { //连接成功 }</pre>		

3.1.2 CloseSpecComPort 【关闭指定串口】

定义	public native int CloseSpecComPort(int PortHandle);		
说明	撤销指定串口和读写器的连接并释放相应资源。		
参数	名称	类型	备注
	PortHandle	int	与读写器连接的句柄, 打开串口 API 获得的句柄。
返回 (int)	成功 : 0 ; 失败 : 非 0 ; (查看返回值错误代码表)		
参考代码	无		

3.1.3 OpenNetPort 【打开 TCP 连接】

定义	public native int OpenNetPort(int Port, String IPAddr, byte[] comAddr, int[] PortHandle);		
说明	打开与读写器的 TCP 连接。		
参数	名称	类型	备注
	port	int	与读写器连接的端口号。
	IPAddr	String	读写器的 IP 地址。
	ComAddr	byte[]	输入读写器地址, 对于 TCP 连接, 可直接输入广播地址 255。
	PortHandle	int[]	返回与读写器连接的句柄, 此后所有 API 操作都需要此句柄。
返回 (int)	成功 : 0 ; 失败 : 非 0 ; (查看返回值错误代码表)		
参考代码	<pre>int Port = 27011; String ipAddr="192.168.0.250"; byte[] comAddr=new byte[1]; comAddr[0]=(byte)255; int[] PortHandle= new int[1]; int result = OpenNetPort(Port, ipAddr, comAddr, PortHandle); if(result==0) { //连接成功 }</pre>		

3.1.4 CloseNetPort 【关闭 TCP 连接】

定义	public native int CloseNetPort(int PortHandle);		
说明	关闭与读写器的 TCP 连接并释放相应资源。		
参数	名称	类型	备注
	PortHandle	int	与读写器连接的句柄, 打开 TCP 连接获得的句柄。
返回 (int)	成功 : 0 ; 失败 : 非 0 ; (查看返回值错误代码表)		
参考代码	无		

3.2. 18000-6C 标准命令

3.2.1. Inventory_G2【查询命令】

定义	public native int Inventory_G2(byte[] comAddr, byte QValue, byte Session, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, byte MaskFlag, byte AdrTID, byte LenTID, byte TIDFlag, byte Target, byte InAnt, byte Scantime, byte FastFlag, byte[] pEPCList, byte[] Ant, int[] Totallen, int[] CardNum, int PortHandle);			
说明	检查有效范围内是否有符合协议的电子标签存在。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	QValue	byte	[in]	1个字节。 bit7：统计数据包标志。 0 – 查询结束后，不发送查询过程统计数据包； 1 – 查询结束后，发送查询过程统计数据包。 Bit5：快速 ID 查询标志位，仅支持 Impinj Monza 系列部分标签，用于快速获取标签的 EPC 和 TID。 0 – 不启用；1 – 启用。 Bit4 ~ bit0：查询 EPC 标签时使用的初始 Q 值， 注：1、Q 值的设置应为场内的标签数量约等于 2 的 Q 次方。Q 值的范围为 0 ~ 15，若命令中出现了其它值，将返回参数出错的消息。 2、带统计包返回查询速度比不带慢。
	Session	byte	[in]	1个字节，查询 EPC 标签时使用的 Session 值。 0x00：Session 使用 S0； 0x01：Session 使用 S1； 0x02：Session 使用 S2； 0x03：Session 使用 S3。 0xff：读写器自动配置 Session（仅对 EPC 查询有效） 建议单张或者少量标签查询选 S0。
	MaskMem	byte	[in]	1个字节，掩码区。 1 – EPC 存储区； 2 – TID 存储区； 3 – 用户存储区。
	MaskAdr	byte []	[in]	2个字节，掩码的起始位地址（单位：Bits）。范围 0~16383。
	MaskLen	byte	[in]	1个字节，掩码的位长度（单位：Bits）。
	MaskData	byte []	[in]	n个字节，掩码数据。n 等于 MaskLen/8。如果 MaskLen 不是 8 的整数倍，则 n 为 [MaskLen/8] 取整再加 1。不够的在低位补 0。

	MaskFlag	byte	[in]	1个字节, 掩码标志位。0 – 不掩码；1 – 掩码。
	AdrTID	byte	[in]	1个字节, 询问 TID 区的起始字地址。
	LenTID	byte	[in]	1个字节, 询问 TID 区的数据字数, 范围 0~15。
	TIDFlag	byte	[in]	1个字节, TID 询问标志位。 0 – 读 EPC ; 1 – 读 TID。
	Target	byte	[in]	1个字节, 询问标签的 Target 值。 0 – Target A ; 1 – Target B。
	InAnt	byte	[in]	1个字节, 本次要进行查询的天线号。 0x80 – 天线 1 ; 0x81 – 天线 2 ; 0x82 – 天线 3 ; 0x83 – 天线 4 ; 单口读写器固定 0x80。
	Scantime	byte	[in]	1个字节, 本次命令查询的最大时间, 范围 0-255(*100ms), 当时间为 0 没, 表示口有限制。
	FastFlag	byte	[in]	1个字节, 快速查询的标志位。 0 – 不启用, 不需要要指定 Target, Inant, Scantime 3 个参数, 默认 0 即可 ; 1 – 启用, 此时要指定 Target, Inant, Scantime 3 个参数。
	pEPCList	byte []	[out]	返回的标签数据, 格式为 : EPC/TID 长度+EPC/TID 号+RSSI。当有多张标签时, 由若干个同格式数据按顺序排列返回。
	Ant	byte []	[out]	1个字节, 查询到标签的天线。 4 口读写器, 每个 bit 位表示天线号, 如 0x04, 二进制为 0000 0100, 表示天线 3 侦查到该标签。 12 口读写器, 按数字 0~11 分别对应天线 1 到天线 12。
	Totalen	int[]	[out]	pEPCList 收到数据的总长度。
	CardNum	int[]	[out]	本次查询到的标签张数。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0x01: 命令执行结束, 同时返回侦查到的电子标签数据; 0x02: 侦查时间结束, 命令执行强制退出, 同时返回已侦查到的标签数据 0xF8: 天线连接检测错误, 当前天线连接可能已经断开; 其余值, 请查看返回值错误代码表。			
参考代码	无			

3.2.2. InventoryMix_G2 【混合查询命令】

定义	public native int InventoryMix_G2(byte[] comAddr, byte QValue, byte Session, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, byte MaskFlag, byte ReadMem, byte[] ReadAdr, byte ReadLen, byte[] Psd, byte Target, byte InAnt, byte Scantime, byte FastFlag, byte[] pEPCList, byte[] Ant, int[] Totallen, int[] CardNum, int PortHandle);			
说明	检查有效范围内是否有符合协议的电子标签存在，并在查询到标签EPC号的同时读取指定的数据。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	QValue	byte	[in]	<p>1个字节。 bit7：统计数据包标志。 0 – 询问结束后，不发送查询过程统计数据包； 1 – 询问结束后，发送查询过程统计数据包。 bit6 ~ bit0：询问EPC标签时使用的初始Q值。 注：1、Q值的设置应为场内的标签数量约等于2的Q次方。Q值的范围为0 ~ 15，若命令中出现了其它值，将返回参数出错的消息。 2、带统计包发送速度比不带慢。</p>
	Session	byte	[in]	<p>1个字节，询问标签时使用的Session值。 0x00：Session使用S0； 0x01：Session使用S1； 0x02：Session使用S2； 0x03：Session使用S3。 建议单张或者少量标签查询选S0。</p>
	MaskMem	byte	[in]	<p>1个字节，掩码区。 1 – EPC存储区； 2 – TID存储区； 3 – 用户存储区。</p>
	MaskAdr	byte []	[in]	2个字节，掩码的起始位地址（单位：Bits）。范围0~16383。
	MaskLen	byte	[in]	1个字节，掩码的位长度（单位：Bits）。
	MaskData	byte []	[in]	n个字节，掩码数据。n等于MaskLen/8。如果MaskLen不是8的整数倍，则n为[MaskLen/8]取整再加1。不够的在低位补0。
	MaskFlag	byte	[in]	1个字节，掩码标志位。0 – 不掩码；1 – 掩码。
	ReadMem	byte	[in]	<p>1个字节，选择要读取的存储区， 0x00 – 保留区； 0x01 – EPC存储区； 0x02 – TID存储区； 0x03 – 用户存储区。其他值保留。若命令中出现了其它值，将返回参数出错的消息。</p>
	ReadAdr	byte []	[in]	2个字节，要读取存储数据的起始字地址，高字

				节在前。
ReadLen	byte	[in]	1 个字节, 要读取存储数据的字长度。	
Psd	byte []	[in]	4 个字节, 标签的访问密码。	
Target	byte	[in]	1 个字节, 查询标签的 Target 值。 0 – Target A ; 1 – Target B。	
InAnt	byte	[in]	1 个字节, 本次要进行查询的天线号。 0x80 – 天线 1 ; 0x81 – 天线 2 ; 0x82 – 天线 3 ; 0x83 – 天线 4 ; 单口读写器固定 0x80。	
Scantime	byte	[in]	1 个字节, 本次命令查询的最大时间, 范围 0-255(*100ms), 当时间为 0 时, 表示没有限制。	
FastFlag	byte	[in]	1 个字节, 快速查询的标志位。 0 – 不启用, 不需要指定 Target, Inant, Scantime 3 个参数, 默认 0 即可 ; 1 – 启用, 此时要指定 Target, Inant, Scantime 3 个参数。	
pEPCList	byte []	[out]	返回的标签数据, 格式为 : 滚码+EPC/数据长度+EPC/数据+RSSI。当有多张标签时, 由若干个同格式数据按顺序排列返回。滚码最高位为 0 表示当前为 EPC, 最高位为 1 表示存储数据, 同张标签数据先上传 EPC 号, 再上传存储数据 低 7 位如果相邻 可确认是同张数据, 不相邻, 非同张标签数据, 滚码低 7 位从 0 到 127, 每上传一次序号加一, 超过 127, 从 0 重新开始	
Ant	byte []	[out]	1 个字节, 查询到标签的天线。 4 口读写器, 每个 bit 位表示天线号, 如 0x04, 二进制为 0000 0100, 表示天线 3 侦查到该标签。 12 口读写器, 按数字 0-11 分别对应天线 1 到天线 12。	
Totalen	int[]	[out]	pEPCList 收到数据的总长度。	
CardNum	int[]	[out]	本次查询到的标签张数。	
PortHandle	int	[in]	连接读写器返回的句柄。	
返回 (int)	0x01: 命令执行结束, 同时返回侦查到的电子标签数据; 0x02: 侦查时间结束, 命令执行强制退出, 同时返回已侦查到的标签数据 0xF8: 天线连接检测错误, 当前天线连接可能已经断开; 其余值, 请查看返回值错误代码表。			
参考代码	无。			

3.2.3. ReadData_G2 【读数据命令】

定义	public native int ReadData_G2(byte[] comAddr, byte[] EPC, byte ENum, byte Mem, byte WordPtr, byte Num, byte[] Password, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, byte[] Data, int[] ErrorCode, int PortHandle);			
说明	该命令读取标签的整个或部分保留区 EPC存储器 TID存储器或用户存储器中的数据。从指定的地址开始读，以字为单位。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	EPC	byte []	[in]	ENum个字，标签的EPC号。
	ENum	byte	[in]	1个字节，标签的EPC号字长度，当Enum=255时，使用掩码方式。
	Mem	byte	[in]	1个字节，选择要读取的存储区， 0x00 – 保留区； 0x01 – EPC 存储区； 0x02 – TID 存储区； 0x03 – 用户存储区。其他值保留。若命令中出现了其它值，将返回参数出错的消息。
	WordPtr	byte	[in]	1个字节，要读取的起始字地址。
	Num	byte	[in]	1个字节，要读取的字长度。
	Password	byte []	[in]	4个字节，标签的访问密码。
	MaskMem	byte	[in]	1个字节，掩码区。 1 – EPC 存储区； 2 – TID 存储区； 3 – 用户存储区。
	MaskAdr	byte []	[in]	2个字节，掩码的起始位地址（单位：Bits）。范围0~16383。
	MaskLen	byte	[in]	1个字节，掩码的位长度（单位：Bits）。
	MaskData	byte []	[in]	n个字节，掩码数据。n等于MaskLen/8。如果MaskLen不是8的整数倍，则n为[MaskLen/8]取整再加1。不够的在低位补0。
	Data	byte []	[out]	Num个字，读取到的标签数据。
	ErrorCode	int []	[out]	返回值为0xFC时，返回标签错误信息，请查看ErrorCode错误代码表。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于0:表示失败，请查看返回值错误代码表。			
参考代码	无。			

3.2.4. WriteData_G2 【写数据命令】

定义	public native int WriteData_G2(byte[] comAddr, byte[] EPC, byte WNum, byte ENum, byte Mem, byte WordPtr, byte[] Writedata, byte[] Password, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, int[] ErrorCode, int PortHandle);			
说明	该命令可以一次性往保留内存、EPC存储器、TID存储器或用户存储器中写入若干个字。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	EPC	byte []	[in]	ENum个字, 标签的EPC号。
	Wnum	byte	[in]	1个字节, 要写入的字数, 当Enum=255时, 使用掩码方式。
	Enum	byte	[in]	1个字节, 标签的EPC号字长度。当Enum=255时, 使用掩码方式。
	Mem	byte	[in]	1个字节, 选择要写入的存储区, 0x00 – 保留区 ; 0x01 – EPC 存储区 ; 0x02 – TID 存储区 ; 0x03 – 用户存储区。其他值保留。若命令中出现了其它值, 将返回参数出错的消息。。
	WordPtr	byte	[in]	1个字节, 要写入的起始字地址。
	Writedata	byte []	[in]	WNum个字节, 要写入的数据。
	Password	byte []	[in]	4个字节, 标签的访问密码。
	MaskMem	byte	[in]	1个字节, 掩码区。 1 – EPC 存储区 ; 2 – TID 存储区 ; 3 – 用户存储区。
	MaskAdr	byte []	[in]	2个字节, 掩码的起始位地址 (单位: Bits)。范围0~16383。
	MaskLen	byte	[in]	1个字节, 掩码的位长度 (单位: Bits)。
	MaskData	byte []	[in]	n个字节, 掩码数据。n等于MaskLen/8。如果MaskLen不是8的整数倍, 则n为[MaskLen/8]取整再加1。不够的在低位补0。
返回 (int)	0:表示成功。不等于0:表示失败, 请查看返回值错误代码表。			
参考代码	无。			

3.2.5. WriteEPC_G2 【广播写 EPC 号命令】

定义	public native int WriteEPC_G2(byte[] ComAddr, byte[] Password, byte[] EPC, byte ENum, int[] ErrorCode, int PortHandle);					
说明	该命令向电子标签写入EPC号。写入的时候，天线有效范围内只能有一张电子标签。					
参数	名称	类型	方向	备注		
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。		
	Password	byte []	[in]	4个字节，标签的访问密码。		
	EPC	byte []	[in]	Enum 个字节，要写入的 EPC 号。		
	ENum	byte	[in]	1 个字节，写入的标签字长度。		
	ErrorCode	int[]	[out]	返回值为 0xFC 时，返回标签错误信息，请查看 ErrorCode 错误代码表。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败，请查看返回值错误代码表。					
参考代码	无。					

3.2.6. KillTag_G2 【销毁标签命令】

定义	public native int KillTag_G2(byte[] ComAddr, byte[] EPC, byte ENum, byte[] Killpwd, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, int[] ErrorCode, int PortHandle);			
说明	该命令用来销毁标签。标签销毁后，永远不会再处理读写器的命令。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	EPC	byte []	[in]	ENum个字，标签的EPC号。
	ENum	byte	[in]	1个字节，标签的EPC号字长度，当Enum=255时，使用掩码方式。
	Killpwd	byte []	[in]	4个字节，标签的杀死密码。
	MaskMem	byte	[in]	1个字节，掩码区。 1-EPC存储区； 2-TID存储区； 3-用户存储区。
	MaskAdr	byte []	[in]	2个字节，掩码的起始位地址（单位：Bits）。范围0~16383。
	MaskLen	byte	[in]	1个字节，掩码的位长度（单位：Bits）。
	MaskData	byte []	[in]	n个字节，掩码数据。n等于MaskLen/8。如果MaskLen不是8的整数倍，则n为[MaskLen/8]取整再加1。不够的在低位补0。
	ErrorCode	int []	[out]	返回值为0xFC时，返回标签错误信息，请查看ErrorCode错误代码表。
返回 (int)	0:表示成功。不等于0:表示失败，请查看返回值错误代码表。			
参考代码	无。			

3.2.7. Lock_G2 【设定存储区读写保护状态命令】

定义	public native int Lock_G2(byte[] comAddr, byte[] EPC, byte ENum, byte selectid, byte setprotect, byte[] Password, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, int[] ErrorCode, int PortHandle);			
说明	该命令可以设定保留区为可读写、永远可读写、带密码可读写、永远可读写，可以分别设定 EPC 存储器、TID 存储器和用户存储器为可写、永远可写、带密码可写、永远不可写。EPC 存储器、TID 存储器或用户存储器是永远可读的。而且，TID 存储器是只读的，永远都不可写。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	EPC	byte []	[in]	ENum个字节，标签的EPC号。
	ENum	byte	[in]	1个字节，标签的EPC号字长度，当Enum=255时，使用掩码方式。
	selectid	byte	[in]	1个字节， 0x00 – 控制 Kill 密码读写保护设定。 0x01 – 控制访问密码读写保护设定。 0x02 – 控制 EPC 存储器读写保护设定。 0x03 – 控制 TID 存储器读写保护设定。 0x04 – 控制用户存储器读写保护设定。 其它值保留，若出读写器接收到其他值，将返回参数出错的消息。
	setprotect	byte	[in]	1个字节， 当 Select 为 0x00 或 0x01， SetProtect 值代表的意义如下： 0x00 – 设置为可读写 0x01 – 设置为永远可读写 0x02 – 设置为带密码可读写 0x03 – 设置为永远不可读写 当 Select 为 0x02、0x03、0x04 时， SetProtect 值代表的意义如下： 0x00 – 设置为可写 0x01 – 设置为永远可写 0x02 – 设置为带密码可写 0x03 – 设置为永远不可写。
	Password	byte []	[in]	4个字节，标签的访问密码。
	MaskMem	byte	[in]	1个字节，掩码区。 1 – EPC 存储区； 2 – TID 存储区； 3 – 用户存储区。
	MaskAdr	byte []	[in]	2个字节，掩码的起始位地址（单位：Bits）。范围 0~16383。
	MaskLen	byte	[in]	1个字节，掩码的位长度（单位：Bits）。
	MaskData	byte []	[in]	n个字节，掩码数据。n 等于 MaskLen/8。如果

				MaskLen 不是 8 的整数倍，则 n 为 [MaskLen/8] 取整再加 1。不够的在低位补 0。
	ErrorCode	int []	[out]	返回值为 0xFC 时，返回标签错误信息，请查看 ErrorCode 错误代码表。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于 0:表示失败，请查看返回值错误代码表。			
参考代码	无。			

3.2.8. BlockErase_G2 【块擦除命令】

定义	public native int BlockErase_G2(byte[] ComAddr, byte[] EPC, byte ENum, byte Mem, byte WordPtr, byte Num, byte[] Password, byte MaskMem, byte[] MaskAdr, byte MaskLen, byte[] MaskData, int[] ErrorCode, int PortHandle);			
说明	该命令可以擦除 <u>保留</u> 内存、EPC存储器、TID存储器或用户存储器的若干字。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	EPC	byte []	[in]	ENum个字, 标签的EPC号。
	ENum	byte	[in]	1个字节, 标签的EPC号字长度, 当Enum=255时, 使用掩码方式。
	Mem	byte	[in]	1个字节, 选择要擦除的存储区, 0x00 – 保留区; 0x01 – EPC 存储区; 0x02 – TID 存储区; 0x03 –用户存储区。其他值保留。若命令中出现了其它值, 将返回参数出错的消息。。
	WordPtr	byte []	[in]	1个字节, 要擦除的起始字地址。
	Num	byte	[in]	1个字节, 要擦除的字长度。
	Password	byte []	[in]	4个字节, 标签的访问密码。
	MaskMem	byte	[in]	1个字节, 掩码区。 1 – EPC 存储区; 2 – TID 存储区; 3 –用户存储区。
	MaskAdr	byte []	[in]	2个字节, 掩码的起始位地址 (单位: Bits)。范围0~16383。
	MaskLen	byte	[in]	1个字节, 掩码的位长度 (单位: Bits)。
	MaskData	byte []	[in]	n个字节, 掩码数据。n等于MaskLen/8。如果MaskLen不是8的整数倍, 则n为[MaskLen/8]取整再加1。不够的在低位补0。
	ErrorCode	int []	[out]	返回值为0xFC时, 返回标签错误信息, 请查看ErrorCode错误代码表。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于0:表示失败, 请查看返回值错误代码表。			
参考代码	无。			

3.3.读写器自定义命令

3.3.1. GetReaderInformation 【获取读写器信息命令】

定义	public native int GetReaderInformation(byte[] comAddr, byte[] versionInfo, byte[] readerType, byte[] trType, byte[] dmaxfre, byte[] dminfre, byte[] powerdBm, byte[] ScanTime, byte[] Ant, byte[] BeepEn, byte[] OutputRep, byte[] CheckAnt, int PortHandle);			
说明	获得读写器的信息, 这其中包括读写器地址 (ComAdr) 和读写器软件版本 (VersionInfo) 的信息等多项信息			
参数	名称	类型	方向	备注
	ComAddr	byte[]	[in/out]	1个字节, 使用连接读写器返回的读写器地址。
	VersionInfo	byte []	[out]	2个字节, 读写器版本信息。第1个字节为版本号, 第2个字节为子版本号柄。
	ReaderType	byte[]	[out]	1个字节, 读写器类型代码。
	dmaxfre	byte[]	[out]	1个字节, Bit7–Bit6 用于频段设置用 ; Bit5–Bit0 表示当前读写器工作的最大频点。
	dminfre	byte []	[out]	1个字节, Bit7–Bit6 用于频段设置用 ; Bit5–Bit0 表示当前读写器工作的最小频点。
	powerdBm	byte []	[out]	1个字节, 读写器的输出功率。范围是 0 到 30, 单位 dBm。
	ScanTime	byte []	[out]	1个字节, 查询时间。读写器收到查询命令后, 在查询时间内, 会给上位机应答。
	Ant	byte []	[out]	1个字节, 天线配置信息。
	BeepEn	byte []	[out]	1个字节, 蜂鸣器开关信息。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

注 : 频段设置如下表:

MaxFre(Bit7)	MaxFre(Bit6)	MinFre(Bit7)	MinFre(Bit6)	FreqBand
0	0	0	1	Chinese band2
0	0	1	0	US band
0	0	1	1	Korean band
0	1	0	0	EU band
0	1	1	0	Ukraine band
1	0	0	0	Chinese band1
1	1	0	0	US band3

3.3.2 SetAddress 【设置读写器地址命令】

定义	public native int SetAddress(byte[]comAddr, byte ComAdrData,int PortHandle);			
说明	执行该命令后，读写器将会把读写器地址改为用户给定的值，并把这个值写入 EEPROM 保存。出厂时默认值是 0x00。允许用户的修改范围是 0x00~0xFE。当用户写入的值是 0xFF 时，读写器将会自动恢复成默认值 0x00。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	ComAdrData	byte	[in]	1 个字节，要修改的新地址。
返回 (int)	PortHandle	int	[in]	连接读写器返回的句柄。
	0:表示成功。不等于 0:表示失败，请查看返回值错误代码表。			
参考代码	无			

3.3.3 SetInventoryScanTime 【查询命令最大响应时间设置命令】

定义	public native int SetInventoryScanTime(byte[]comAddr, byte ScanTime,int PortHandle);			
说明	该命令用于设置查询命令的最大响应时间。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	ScanTime	byte	[in]	1 个字节,查询命令响应时间, 范围 0 到 255。 设置为 0 时, 该参数无效, 一直等待, 直到该轮读完。
PortHandle	int	[in]		连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

3.3.4. SetRfPower 【功率设置命令】

定义	public native int SetRfPower(byte[]comAddr, byte PowerDbm,int PortHandle);			
说明	设置读写器功率。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	PowerDbm	byte	[in]	1个字节, 读写器功率, 范围 0~30dBm。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

3.3.5. SetRegion 【工作频率设置命令】

定义	public native int SetRegion(byte[]comAddr, byte dmaxfre,byte dminfre,int PortHandle);			
说明	该命令设置读写器工作地上限频率, 下限频率。上限频率必须大于或等于下限频率。			
参数	名称	类型	方向	备注
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。
	dmaxfre	byte	[in]	1个字节, Bit7–Bit6 用于频段设置用 ; Bit5–Bit0 表示当前读写器工作的最大频点。
	dminfre	byte	[in]	1个字节, Bit7–Bit6 用于频段设置用 ; Bit5–Bit0 表示当前读写器工作的最小频点。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

注 : 频段设置如下表:

MaxFre(Bit7)	MaxFre(Bit6)	MinFre(Bit7)	MinFre(Bit6)	FreqBand
0	0	0	1	Chinese band2
0	0	1	0	US band
0	0	1	1	Korean band
0	1	0	0	EU band
0	1	1	0	Ukraine band
1	0	0	0	Chinese band1
1	1	0	0	US band

各频段计算公式 :

$$\text{Chinese band2: } Fs = 920.125 + N * 0.25 \text{ (MHz)} \text{ 其中 } N \in [0, 19].$$

$$\text{US band: } Fs = 902.75 + N * 0.5 \text{ (MHz)} \text{ 其中 } N \in [0, 49].$$

$$\text{Korean band: } Fs = 917.1 + N * 0.2 \text{ (MHz)} \text{ 其中 } N \in [0, 31].$$

$$\text{EU band: } Fs = 865.1 + N * 0.2 \text{ (MHz)} \text{ 其中 } N \in [0, 14].$$

$$\text{Ukraine band: } Fs = 868.0 + N * 0.1 \text{ (MHz)} \text{ 其中 } N \in [0, 6].$$

$$\text{Chinese band1: } Fs = 840.125 + N * 0.25 \text{ (MHz)} \text{ 其中 } N \in [0, 19].$$

$$\text{US band3: } Fs = 902 + N * 0.5 \text{ (MHz)} \text{ 其中 } N \in [0, 52].$$

3.3.6. SetBaudRate 【波特率设置命令】

定义	public native int SetBaudRate(byte[]comAddr, byte baud,int PortHandle);			
说明	设置读写器通讯波特率。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	baud	byte	[in]	1个字节, 波特率。 0 – 9600bps; 1 – 19200bps; 2 – 38400bps; 5 – 57600bps; 6 – 115200bps.
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

3.3.7. SetAntennaMultiplexing 【天线配置命令】

定义	public native int SetAntennaMultiplexing(byte[]comAddr, byte Ant,int PortHandle);			
说明	该命令用于4口读写器天线配置。			
参数	名称	类型	方向	备注
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。
	Ant	byte	[in]	1个字节, 天线配置信息。天线至少要配置一个。 Bit0 : 天线 1 配置位。 Bit0=0 天线 1 关闭 ; Bit0=1 天线 1 打开。 Bit1 : 天线 2 配置位。 Bit1=0 天线 2 关闭 ; Bit1=1 天线 2 打开。 Bit2 : 天线 3 配置位。 Bit2=0 天线 3 关闭 ; Bit2=1 天线 3 打开。 Bit3 : 天线 4 配置位。 Bit3=0 天线 4 关闭 ; Bit3=1 天线 4 打开。
	PortHandle	int	[in]	连接读写器返回的句柄。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

3.3.8. WriteRfPower 【设置写功率命令】

定义	public native int WriteRfPower(byte[]comAddr, byte PowerDbm,int PortHandle);					
说明	该命令用来设置写标签时的功率。					
参数	名称	类型	方向	备注		
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。		
	PowerDbm	byte	[in]	1 个字节, 读写器的写功率, 0-30 ; bit7=0 – 不使能 ; bit7=1 – 使能 ;。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。					
参考代码	无					

3.3.9. ReadRfPower 【读取写功率命令】

定义	public native int ReadRfPower(byte[] comAddr, byte[] PowerDbm,int PortHandle);					
说明	该命令用来读取写标签时的功率。					
参数	名称	类型	方向	备注		
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。		
	PowerDbm	byte []	[out]	1 个字节, 读写器的写功率, 0-30 ; bit7=0 – 不使能 ; bit7=1 – 使能 ;。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。					
参考代码	无					

3.3.10. RetryTimes 【写重复次数设置命令】

定义	public native int RetryTimes(byte[]comAddr, byte[] Times,int PortHandle);					
说明	该命令用来读取写标签时的功率。					
参数	名称	类型	方向	备注		
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。		
	Times	byte []	[out]	1 个字节, bit7=1 – 设置重写次数; bit7=0 – 读取重写次数 ; 范围 0-7。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。					
参考代码	无					

3.3.11. SetReadMode 【实时模式设置命令】

定义	public native int SetReadMode(byte[]comAddr, byte ReadMode,int PortHandle);					
说明	该命令用来设置读写器实时查询模式。					
参数	名称	类型	方向	备注		
	ComAddr	byte	[in/out]	使用连接读写器返回的读写器地址。		
	ReadMode	byte	[in]	1 个字节, 0 – 应答模式; 1 – 实时查询模式 ; 2 – 触发模式。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。					
参考代码	无					

3.3.12. SetCheckAnt 【天线检测命令】

定义	public native int SetCheckAnt(byte[]comAddr, byte CheckAnt,int PortHandle);					
说明	该命令用于设置读写标签的过程中是否进行天线检测。					
参数	名称	类型	方向	备注		
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。		
	CheckAnt	byte	[in]	1字节, 天线检测开关。 0x00 – 关闭读写标签过程中的天线检测。 0x01 – 开启读写标签过程中的天线检测。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于0:表示失败, 请查看返回值错误代码表。					
参考代码	无					

3.3.13. GetSeriaNo 【获取读写器序列号命令】

定义	public native int GetSeriaNo(byte[]comAddr, byte[] SeriaNo,int PortHandle);			
说明	该命令用于获取读写器4字节的唯一序列号。			
参数	名称	类型	方向	备注
	ComAddr	byte []	[in/out]	使用连接读写器返回的读写器地址。
	SeriaNo	byte []	[out]	4 字节, 读写器序列号。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。			
参考代码	无			

3.3.14. SetBeepNotification 【蜂鸣器设置命令】

定义	public native int SetBeepNotification(byte[] comAddr, byte BeepEn, int PortHandle);;					
说明	该命令用于设置蜂鸣器开关。					
参数	名称	类型	方向	备注		
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。		
	BeepEn	byte	[in]	1 个字节, 0x00 – 蜂鸣器关； 0x01 – 蜂鸣器开。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看返回值错误代码表。					
参考代码	无					

3.3.15. SetDRM【设置 DRM】

定义	public native int SetDRM(byte[] ComAddr, byte DRM, int FrmHandle);			
说明	该命令用来设置 DRM 状态。			
参数	名称	类型	方向	备注
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。
	DRM	byte	[in]	1 字节, DRM 状态, 0 – 关闭 ; 1 – 打开。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。			
参考代码	无			

3.3.16. GetDRM【读取 DRM】

定义	public native int GetDRM(byte[] ComAddr, byte[] DRM, int FrmHandle);			
说明	该命令用来读取 DRM 状态。			
参数	名称	类型	方向	备注
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。
	DRM	byte[]	[out]	1 字节, DRM 状态, 0 – 关闭 ; 1 – 打开。
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。			
参考代码	无			

3.3.17. GetReaderTemperature 【读取温度】

定义	public native int GetReaderTemperature (byte[]ComAddr, byte[] PlusMinus,byte[] Temperature, int PortHandle);					
说明	该命令用来读取读写器当前温度。					
参数	名称	类型	方向	备注		
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。		
	PlusMinus	byte[]	[out]	1字节, 温度状态, 0 - 零下; 1 - 零上。		
	Temperature	byte[]	[out]	1字节, 温度数值, 单位°c		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于0:表示失败, 请查看错误代码表。					
参考代码	无					

3.3.18. MeasureReturnLoss 【测量回损】

定义	public native int MeasureReturnLoss(byte[] ComAddr, byte[] TestFreq, byte Ant, byte[] ReturnLoss,int PortHandle);					
说明	该命令用来测量读写器指定端口的回波损耗。					
参数	名称	类型	方向	备注		
	ComAddr	byte[]	[in/out]	使用连接读写器返回的读写器地址。		
	TestFreq	byte[]	[in]	4 字节, 测量频点。		
	Ant	byte	[in]	1 字节, 测量天线端口, 0 – 天线 1 ; 1 – 天线 2。 2 – 天线 3 ; 3 – 天线 4。		
	ReturnLoss	byte[]	[out]	1 字节, 回波损耗, 单位 dB。		
PortHandle	int	[in]	连接读写器返回的句柄。			
返回 (int)	0:表示成功。不等于 0:表示失败, 请查看错误代码表。					
参考代码	无					

附录 1. 返回值错误代码表

错误代码	描述
0x00	无错误，表示 API 调用成功。
0x01	查询时间结束前返回。
0x02	指定的查询时间溢出。
0x05	访问密码错误。
0x09	销毁密码错误。
0x0A	销毁密码不能为全 0。
0x0B	电子标签不支持该命令。
0x0C	对该命令，访问密码不能为 0。
0x0D	电子标签已经被设置了读保护，不能再次设置。
0x0E	电子标签没有被设置读保护，不需要解锁。
0x10	有字节空间被锁定，写入失败。
0x11	不能锁定。
0x12	已经锁定，不能再次锁定。
0x13	参数保存失败,但设置的值在读写模块断电前有效。
0x14	无法调整。
0x15	查询时间结束前返回。
0x17	本条消息之后，还有消息。
0x18	读写模块存储空间已满。
0x19	电子不支持该命令或者访问密码不能为 0
0xF8	天线检测错误。
0xF9	命令执行出错。
0xFA	有电子标签，但通信不畅，无法操作。
0xFB	无电子标签可操作。
0xFC	电子标签返回错误代码。
0xFD	命令长度错误。
0xFE	不合法的命令。
0xFF	参数错误。
0x30	通讯错误。
0x33	通讯繁忙，设备正在执行其他命令。
0x35	端口已打开
0x37	无效句柄。

附录 2. ErrorCode 错误代码表

错误代码	描述
0x00	其它错误，全部捕捉未被其它代码覆盖的错误。
0x03	存储器超限或不被支持的 PC 值，规定存储位置不存在或标签不支持 PC 值。
0x04	存储器锁定，存储位置锁定永久锁定，且不可写入。
0x0B	电源不足，标签电源不足，无法执行存储写入操作。
0x0F	非特定错误，标签不支持特定错误代码。