KLM600 UhfReader_API_HID 函数库说明文档

目录：

1. 简介

《UhfReader_API_HID（免驱）函数库说明文档》主要介绍 KLM 模块的每个功能函数的定义、说明。该文档主要包括：简介、通用命令集、标签盘点命令集、标签数据存取命令集、固件升级命令集、其他命令集和附录部分。

HID 是 Human Interface Device 的简称，即人机接口设备，KLM600 是符合 HID 类别规范的超高频模块。用户在使用 KLM600 模块时，先将 KLM600 设备与计算机连接，然后调用 UhfSearchHids 函数搜索本台计算机上可用的 HID 序列号，最后以 HID 序列号作为参数（char* cPort）通过 UhfReaderConnect 或者 UhfOpenPort 函数与模块建立通信连接，在通信过程中这两个函数的返回参数（HANDLE &hCom）将作为其他函数的输入参数使用，详见示例代码：UHF_HID_Example。

2. 通用命令集

2.1 UhfReaderConnect ()

2.1.1 功能简介

该函数打开串口并与 KLM 建立链接。

2.1.2 函数原型

int WINAPI UhfReaderConnect (HANDLE &hCom, char* cPort, UCHAR flagCrc);

2.1.3 返回值

1：打开端口成功并连接到 KLM；

其他：打开端口或连接 KLM 失败；

2.1.4 输入参数

HANDLE &hCom：返回参数，通信端口句柄，初始化为 NULL；

char* cPort：输入参数，串口（如 COM1、COM2 等）或 HID 序列号（如 \\?\hid#vid_0483&pid_5750#6&20bfa97&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}等）；

UCHAR flagCrc：输入参数，选择通信波特率以及是否使用 CRC16 验证功能，

0x00：不使用 CRC 功能；

0x01：使用 CRC 功能；

2.2 UhfReaderDisconnect ()

2.2.1 功能简介

该函数关闭 KLM 通信端口。

2.2.2 函数原型

int WINAPI UhfReaderDisconnect (HANDLE &hCom, UCHAR flagCrc);

2.2.3 返回值

1：关闭 KLM 通信端口成功；

其他：关闭 KLM 通信端口失败；

2.2.4 输入参数

HANDLE &hCom：输入参数，通信端口句柄。

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.3 UhfOpenPort ()

2.3.1 功能简介

该函数打开串口。

2.3.2 函数原型

int WINAPI UhfOpenPort (HANDLE &hCom, char* cPort, UCHAR flagCrc);

2.3.3 返回值

1：打开串口成功；

其他：打开串口失败；

2.3.4 输入参数

HANDLE &hCom：返回参数，通信端口句柄，初始化为 NULL；

char* cPort：输入参数，串口（如 COM1、COM2 等）或 HID 序列号（如 \\?\hid#vid_0483&pid_5750#6&20bfa97&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}等）；

UCHAR flagCrc：输入参数，选择通信波特率以及是否使用 CRC16 验证功能，

0x00：不使用 CRC 功能；

0x01：使用 CRC 功能；

2.4 UhfClosePort ()

2.4.1 功能简介

该函数关闭串口。

2.4.2 函数原型

int WINAPI UhfClosePort ();

2.4.3 返回值

1：关闭串口成功；

其他：关闭串口失败；

2.4.4 输入参数

无。

2.5 UhfGetPaStatus ()

2.5.1 功能简介

该函数读取 KLM 连接状态。

2.5.2 函数原型

int WINAPI UhfGetPaStatus (HANDLE hCom, UCHAR* uStatus, UCHAR flagCrc);

2.5.3 返回值

1：连接成功；

其他：连接失败；

2.5.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uStatus：返回参数，KLM 的连接状态（1 字节，0：连接成功；其他：连接失败）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.6 UhfGetPower ()

2.6.1 功能简介

该函数读取 KLM 的功率。

2.6.2 函数原型

int WINAPI UhfGetPower (HANDLE hCom, UCHAR* uPower, UCHAR flagCrc);

2.6.3 返回值

1：读取 KLM 的功率成功；

其他：读取 KLM 的功率失败；

2.6.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uPower：返回参数，KLM 的功率（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.7 UhfSetPower ()

2.7.1 功能简介

该函数设置 KLM 的功率。

2.7.2 函数原型

int WINAPI UhfSetPower (HANDLE hCom, UCHAR uOption, UCHAR uPower, UCHAR flagCrc);

2.7.3 返回值

1：设置 KLM 的功率成功；

其他：设置 KLM 的功率失败；

2.7.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR uOption：输入参数，设置功率命令的 OPTION 字节，必须设置为 0x01；

UCHAR uPower：输入参数，要设置的功率值；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.8 UhfGetFrequency ()

2.8.1 功能简介

该函数读取 KLM 的频率。

2.8.2 函数原型

int WINAPI UhfGetFrequency (HANDLE hCom, UCHAR* uFreMode, UCHAR* uFreBase, UCHAR* uBaseFre, UCHAR* uChannNum, UCHAR* uChannSpc, UCHAR* uFreHop, UCHAR flagCrc);

2.8.3 返回值

1：读取 KLM 的频率成功；

其他：读取 KLM 的频率失败；

2.8.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uFreMode：返回参数，KLM 返回的 FREMODE（1 字节）；

UCHAR* uFreBase：返回参数，KLM 返回的 FREBASE 字节（1 字节）；

UCHAR* uBaseFre：返回参数，KLM 返回的 BF 字节（2 字节）；

UCHAR* uChannNum：返回参数，KLM 返回的 CN 字节（1 字节）；

UCHAR* uChannSpc：返回参数，KLM 返回的 SPC 字节（1 字节）；

UCHAR* uFreHop：返回参数，KLM 返回的 FREHOP 字节（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.9 UhfSetFrequency ()

2.9.1 功能简介

该函数设置 KLM 的频率。

2.9.2 函数原型

int WINAPI UhfSetFrequency(HANDLE hCom, UCHAR uFreMode, UCHAR uFreBase, UCHAR* uBaseFre, UCHAR uChannNum, UCHAR uChannSpc, UCHAR uFreHop, UCHAR flagCrc);

2.9.3 返回值

1：设置 KLM 的频率成功；

其他：设置 KLM 的频率失败；

2.9.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR uFreMode：输入参数，频率设置的 FREMODE（1 字节）；

UCHAR uFreBase：输入参数，频率设置的 FREBASE 字节（1 字节）；

UCHAR* uBaseFre：输入参数，频率设置的 BF 字节（2 字节）；

UCHAR uChannNum：输入参数，频率设置的 CN 字节（1 字节）；

UCHAR uChannSpc：输入参数，频率设置的 SPC 字节（1 字节）；

UCHAR uFreHop：输入参数，频率设置的 FREHOP 字节（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.10 UhfGetRegister ()

2.10.1 功能简介

该函数读取寄存器数据。

2.10.2 函数原型

int WINAPI UhfGetRegister (HANDLE hCom, int RADD, int RLEN, UCHAR* STATUS, UCHAR* REG, UCHAR flagCrc)

2.10.3 返回值

1：读取寄存器数据成功；

其他：读取寄存器数据失败；

2.10.4 参数

HANDLE hCom：输入参数，通信端口句柄；

int RADD：输入参数，读取寄存器数据的起始地址；

int RLEN：输入参数，读取寄存器数据长度（单位：字节）；

UCHAR* STATUS：返回参数，返回的操作状态信息（1 字节）；

UHCAR* REG：返回参数，返回的寄存器数据；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.11 UhfSetRegister ()

2.11.1 功能简介

该函数用于设置寄存器值。

2.11.2 函数原型

int WINAPI UhfSetRegister (HANDLE hCom, int RADD, int RLEN, UCHAR* REG, UCHAR* STATUS, UCHAR flagCrc);

2.11.3 返回值

1：写入寄存器数据成功；

其他：写入寄存器数据失败；

2.11.4 参数

HANDLE hCom：输入参数，通信端口句柄；

int RADD：输入参数，写入寄存器数据的起始地址；

int RLEN：输入参数，写入寄存器数据长度（单位：字节）；

UHCAR* REG：输入参数，写入的寄存器数据；

UCHAR* STATUS：返回参数，返回的操作状态信息（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.12 UhfResetRegister ()

2.12.1 功能简介

该函数用于恢复寄存器出厂默认值。

2.12.2 函数原型

int WINAPI UhfResetRegister (HANDLE hCom, UCHAR flagCrc);

2.12.3 返回值

1：恢复寄存器出厂默认值成功；

其他：恢复寄存器出厂默认值失败；

2.12.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.13 UhfSaveRegister ()

2.13.1 功能简介

该函数用于保存当前设置到相应的寄存器。

2.13.2 函数原型

int WINAPI UhfSaveRegister (HANDLE hCom, UCHAR flagCrc);

2.13.3 返回值

1：保存当前设置成功；

其他：保存当前设置失败；

2.13.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.14 UhfGetVersion ()

2.14.1 功能简介

该函数读取 KLM 的硬件序列号和软件版本号。

2.14.2 函数原型

int WINAPI UhfGetVersion (HANDLE hCom, UCHAR* uSerial, UCHAR* uVersion, UCHAR flagCrc);

2.14.3 返回值

1：读取 KLM 的硬件序列号和软件版本号成功；

其他：读取 KLM 的硬件序列号和软件版本号失败；

2.14.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uSerial：返回参数，KLM 的硬件序列号（6 个字节）；

UCHAR* uVersion：返回参数，KLM 的软件版本号（3 个字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.15 UhfGetReaderUID ()

2.15.1 功能简介

该函数用于读取 KLM 的 UID 信息。

2.15.2 函数原型

int WINAPI UhfGetReaderUID (HANDLE hCom, UCHAR* uUid, UCHAR flagCrc);

2.15.3 返回值

1：读取 UID 成功；

其他：读取 UID 失败；

2.15.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uUid：返回参数，KLM 的 UID 信息（12 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.16 UhfEnterSleepMode ()

2.16.1 功能简介

该函数用于进入睡眠模式。

2.16.2 函数原型

int WINAPI UhfEnterSleepMode (HANDLE hCom, UCHAR flagCrc);

2.16.3 返回值

1：进入睡眠模式成功；

其他：进入睡眠模式失败；

2.16.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

2.17 UhfStopOperation ()

2.17.1 功能简介

该函数停止 KLM 的当前操作。

2.17.2 函数原型

int WINAPI UhfStopOperation (HANDLE hCom, UCHAR flagCrc);

2.17.3 返回值

1：停止当前操作成功；

其他：停止当前操作失败；

2.17.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

3. 标签盘点命令集

3.1 UhfStartInventory ()

3.1.1 功能简介

该函数启动 KLM 的识别循环。

3.1.2 函数原型

int WINAPI UhfStartInventory (HANDLE hCom, UCHAR flagAnti, UCHAR initQ, UCHAR flagCrc);

3.1.3 返回值

1：启动 KLM 的识别循环成功；

其他：启动 KLM 的识别循环失败；

3.1.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR flagAnti：输入参数，是否使用防碰撞识别功能（1：防碰撞识别；0：单标签识别）；

UCHAR initQ：输入参数，防碰撞识别过程的初始 Q 值，flagAnti 为 1 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

【注】

UhfStartInventory()函数用于启动 KLM 的识别循环，当 flagAnti=1 时启动防碰撞功能，用于循环识别多张标签，当 flagAnti=0 时不启动防碰撞功能，用于循环识别单张标签。该函数与 UhfReadInventory()、UhfStopOperation()两个函数配合使用，UhfStartInventory()只是开启识别循环，之后 KLM 将识别到的标签号上传到缓冲区，UhfReadInventory()用于从缓冲区读取一个标签号数据，开启循环识别之后，模块只能响应 UhfStopOperation()函数，UhfStopOperation()停止识别循环。

3.2 UhfReadInventory ()

3.2.1 功能简介

该函数读取 KLM 返回的标签 UII。

3.2.2 函数原型

int WINAPI UhfReadInventory (HANDLE hCom, UCHAR* uLenUii, UCHAR* uUii);

3.2.3 返回值

1：读取标签的 UII 成功；

其他：读取标签的 UII 失败；

3.2.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uLenUii：返回参数，标签 UII 的长度，1 个字节；

UCHAR* uUii：返回参数，标签 UII，至少 66 个字节。

3.3 UhfInventorySingleTag ()

3.3.1 功能简介

该函数单步识别标签，一次只返回一个 UII。

3.3.2 函数原型

int WINAPI UhfInventorySingleTag (HANDLE hCom, UCHAR* uLenUii, UCHAR* uUii , UCHAR flagCrc);

3.3.3 返回值

1：识别标签成功；

其他：识别标签失败；

3.3.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uLenUii：返回参数，标签 UII 的长度，1 个字节；

UCHAR* uUii：返回参数，标签 UII，至少 66 个字节；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4. 标签数据存取命令集

4.1 UhfReadDataByEPC ()

4.1.1 功能简介

该函数读取标签数据（指定 UII）。

4.1.2 函数原型

int WINAPI UhfReadDataByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uUii, UCHAR* uReadData, UCHAR* uErrorCode, UCHAR flagCrc);

4.1.3 返回值

1：读取标签数据成功；

其他：读取标签数据失败；

4.1.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，读取数据的长度（Word 为单位，不能为 0）；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uReadData：返回参数，读取的标签数据，至少为 uCnt * 2 个字节；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.2 UhfReadDataFromSingleTag ()

4.2.1 功能简介

该函数读取标签数据（不指定 UII）。

4.2.2 函数原型

int WINAPI UhfReadDataFromSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uReadData, UCHAR* uUii, UCHAR* uLenUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.2.3 返回值

1：读取标签数据成功；

其他：读取标签数据失败；

4.2.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，读取数据的长度（Word 为单位），不能为 0；

UCHAR* uReadData：返回参数，读取的标签数据，至少为 uCnt * 2 个字节；

UCHAR* uUii：返回参数，被读取的标签的 UII；

UCHAR* uLenUii ：返回参数，UII 长度；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc： 输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.3 UhfReadMaxDataByEPC ()

4.3.1 功能简介

该函数用于读取标签某个存储区的所有数据（指定 UII）。

4.3.2 函数原型

int WINAPI UhfReadMaxDataByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank,

UCHAR* uPtr, UCHAR* uUii, UCHAR* Data_len, UCHAR* uReadData, UCHAR* uErrorCode, UCHAR flagCrc);

4.3.3 返回值

1：读取数据成功；

其他：读取数据失败；

4.3.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* Data_len：返回参数，读取到的数据长度，单位：字节；

UCHAR* uReadData：返回参数，读取的标签数据；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

【注】

目前 KLM100 最大支持 90 个字长，KLM200、KLM300 仅支持存储区大小最长 233 个字长，当存储区大小大于 KLM 支持的最大字长时，函数返回失败。

4.4 UhfReadMaxDataFromSingleTag ()

4.4.1 功能简介

该函数用于读取标签某个存储区的所有数据（不指定 UII）。

4.4.2 函数原型

int WINAPI UhfReadMaxDataFromSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR* Data_len, UCHAR* uReadData, UCHAR* uUii, UCHAR* uLenUii, UCHAR*

uErrorCode, UCHAR flagCrc);

4.4.3 返回值

1：读取数据成功；

其他：读取数据失败；

4.4.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR* Data_len：返回参数，读取到的数据长度，单位：字节；

UCHAR* uReadData：返回参数，读取的标签数据；

UCHAR* uUii：返回参数，标签的 UII；

UCHAR* uLenUii：返回参数，返回的 UII 数据长度，单位：字节；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

【注】

目前 KLM100 最大支持 90 个字长，KLM200、KLM300 仅支持存储区大小最长 233 个字长，当存储区大小大于 KLM 支持的最大字长时，函数返回失败。

4.5 UhfWriteDataByEPC ()

4.5.1 功能简介

该函数向标签写入单字长（1 个 Word）的数据（指定 UII）。

4.5.2 函数原型

int WINAPI UhfWriteDataByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uUii, UCHAR* uWriteData, UCHAR* uErrorCode, UCHAR flagCrc);

4.5.3 返回值

1：写入标签数据成功；

其他：写入标签数据失败；

4.5.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，写入数据的长度（uCnt 必须取值为 1）；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uWriteData：输入参数，需要写入的数据；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.6 UhfWriteDataToSingleTag ()

4.6.1 功能简介

该函数向标签写入单字长（1 个 Word）的数据（不指定 UII）。

4.6.2 函数原型

int WINAPI UhfWriteDataToSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uWriteData, UCHAR* uUii, UCHAR* uLenUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.6.3 返回值

1：写入标签数据成功；

其他：写入标签数据失败；

4.6.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，写入数据的长度（uCnt 必须取值为 1）；

UCHAR* uWriteData：输入参数，需要写入的数据；

UCHAR* uUii：返回参数，被写入的标签的 UII；

UCHAR* uLenUii ：返回参数，UII 长度；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc： 输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.7 UhfBlockWriteDataByEPC ()

4.7.1 功能简介

该函数用于向标签写入多字长的数据（指定 UII）。

4.7.2 函数原型

int WINAPI UhfBlockWriteDataByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uUii, UCHAR* uWriteData, UCHAR* uErrorCode, UCHAR* uStatus, UCHAR* uWritedLen, UCHAR* RuUii, UCHAR flagCrc);

4.7.3 返回值

1：写入标签数据成功；

其他：写入标签数据失败；

4.7.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS Password（4 个字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，标签存储区的地址偏移量；

UCHAR uCnt：输入参数，写入数据长度，以字为单位；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uWriteData：输入参数，所要写入的数据内容；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR* uStatus：返回参数，返回操作状态值（1 字节）；

UCHAR* uWritedLen：返回参数，返回已经成功写入的数据长度，以字为单位；

UCHAR* RuUii：返回参数，返回当前操作的标签 UII；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.8 UhfBlockWriteDataToSingleTag()

4.8.1 功能简介

该函数用于向标签写入多字长的数据（不指定 UII）。

4.8.2 函数原型

int WINAPI UhfBlockWriteDataToSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uWriteData, UCHAR* uUii, UCHAR* uLenUii, UCHAR* uErrorCode, UCHAR* uStatus,UCHAR* uWritedLen, UCHAR flagCrc);

4.8.3 返回值

1：写入标签数据成功；

其他：写入标签数据失败；

4.8.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS Password（4 个字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，标签存储区的地址偏移量；

UCHAR uCnt：输入参数，写入数据长度，以字为单位；

UCHAR* uWriteData：输入参数，所要写入的数据内容；

UCHAR* uUii：返回参数，标签的 UII；

UCHAR* uLenUii：返回参数，返回标签的 UII 长度，以字节为单位；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR* uStatus：返回参数，返回操作状态值（1 字节）；

UCHAR* uWritedLen：返回参数，返回已经成功写入的数据长度，以字为单位；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.9 UhfEraseDataByEPC ()

4.9.1 功能简介

该函数擦除标签数据（指定 UII）。

4.9.2 函数原型

int WINAPI UhfEraseDataByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.9.3 返回值

1：擦除标签数据成功；

其他：擦除标签数据失败；

4.9.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，需要擦除的数据长度（Word 为单位）；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.10 UhfEraseDataFromSingleTag ()

4.10.1 功能简介

该函数擦除标签数据（不指定 UII）。

4.10.2 函数原型

int WINAPI UhfEraseDataFromSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.10.3 返回值

1：擦除标签数据成功；

其他：擦除标签数据失败；

4.10.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR uBank：输入参数，标签的存储区；

UCHAR* uPtr：输入参数，起始地址的偏移量；

UCHAR uCnt：输入参数，需要擦除的数据长度（Word 为单位）；

UCHAR* uUii：返回参数，被擦除标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.11 UhfLockMemByEPC ()

4.11.1 功能简介

该函数锁定标签的指定数据段（指定 UII）。

4.11.2 函数原型

int WINAPI UhfLockMemByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR* uLockData, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.11.3 返回值

1：锁定标签的指定数据段成功；

其他：锁定标签的指定数据段失败；

4.11.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR* uLockData：输入参数，命令的 LOCKDATA 数据段（3 字节）；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.12 UhfLockMemFromSingleTag ()

4.12.1 功能简介

该函数锁定标签的指定数据段（不指定 UII）。

4.12.2 函数原型

int WINAPI UhfLockMemFromSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR* uLockData, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.12.3 返回值

1：锁定标签的指定数据段成功；

其他：锁定标签的指定数据段失败；

4.12.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS PASSWORD（4 字节）；

UCHAR* uLockData：输入参数，命令的 LOCKDATA 数据段（3 字节）；

UCHAR* uUii：返回参数，标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.13 UhfKillTagByEPC ()

4.13.1 功能简介

该函数销毁指定标签（指定 UII）。

4.13.2 函数原型

int WINAPI UhfKillTagByEPC (HANDLE hCom, UCHAR* uKillPwd, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.13.3 返回值

1：销毁指定标签成功；

其他：销毁指定标签失败；

4.13.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uKillPwd：输入参数，标签的 Kill Password（4 字节）；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：返回参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.14 UhfKillSingleTag ()

4.14.1 功能简介

该函数销毁标签（不指定 UII）。

4.14.2 函数原型

int WINAPI UhfKillSingleTag (HANDLE hCom, UCHAR* uKillPwd, UCHAR* uUii, UCHAR* uErrorCode, UCHAR flagCrc);

4.14.3 返回值

1：销毁标签成功；

其他：销毁标签失败；

4.14.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uKillPwd：输入参数，标签的 Kill Password（4 个字节）；

UCHAR* uUii：返回参数，标签的 UII；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.15 UhfBlockWriteEPCByEPC()

4.15.1 功能简介

该函数向 UII 区写入 EPC（指定 UII）。

4.15.2 函数原型

int WINAPI UhfBlockWriteEPCByEPC (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uCnt, UCHAR* uUii, UCHAR* uWriteData, UCHAR* uErrorCode, UCHAR* uStatus, UCHAR* uWritedLen, UCHAR* RuUii, UCHAR flagCrc)

4.15.3 返回值

1：写入 EPC 成功；

其他：写入 EPC 失败；

4.15.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS Password（4 字节）；

UCHAR uCnt：输入参数，写入数据长度，以字为单位；

UCHAR* uUii：输入参数，标签的 UII；

UCHAR* uWriteData：输入参数，所要写入的数据内容；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR* uStatus：返回参数，返回操作状态值（1 字节）；

UCHAR* uWritedLen：返回参数，返回已经成功写入的数据长度，以字为单位；

UCHAR* RuUii：返回参数，返回当前操作的标签 UII；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.16 UhfBlockWriteEPCToSingleTag()

4.16.1 功能简介

该函数向 UII 区写入 EPC（不指定 UII）。

4.16.2 函数原型

int WINAPI UhfBlockWriteEPCToSingleTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uCnt, UCHAR* uWriteData, UCHAR* uUii, UCHAR* uLenUii, UCHAR* uStatus, UCHAR* uErrorCode, UCHAR* uWritedLen, UCHAR flagCrc);

4.16.3 返回值

1：写入 EPC 成功；

其他：写入 EPC 失败；

4.16.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS Password（4 字节）；

UCHAR uCnt：输入参数，写入数据长度，以字为单位；

UCHAR* uWriteData：输入参数，所要写入的数据内容；

UCHAR* uUii：返回参数，标签的 UII；

UCHAR* uLenUii：返回参数，返回标签的 UII 长度，以字节为单位；

UCHAR* uErrorCode：返回参数，标签返回的 Error Code（1 字节）。只在函数返回失败，且 uErrorCode 不等于 0xFF 时有效；

UCHAR* uStatus：返回参数，返回操作状态值（1 字节）；

UCHAR* uWritedLen：返回参数，返回已经成功写入的数据长度，以字为单位；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

4.17 UhfStartReadDataFromMultiTag ()

4.17.1 功能简介

该函数用于开启防碰撞读取标签数据循环。

4.17.2 函数原型

int WINAPI UhfStartReadDataFromMultiTag (HANDLE hCom, UCHAR* uAccessPwd, UCHAR uBank, UCHAR* uPtr, UCHAR uCnt, UCHAR uOption, UCHAR* uPayLoad, UCHAR flagCrc)

4.17.3 返回值

1：开启防碰撞读数据成功；

其他：开启防碰撞读数据失败；

4.17.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uAccessPwd：输入参数，标签的 ACCESS Password（必须为 4 字节的全 0）；

UCHAR uBank：输入参数，标签的存储区（1 级）；

UCHAR* uPtr：输入参数，标签存储区的地址偏移量（1 级）；

UCHAR uCnt：输入参数，要读取的数据长度（1 级），以字为单位（不能为 0，最大支持 220）；

UCHAR uOption：输入参数（0：读取 1 级数据；1：读取 2 级数据）；

UCHAR* uPayLoad：输入参数；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；1：使用 CRC 功能。

【注】当 uOption=0 时，读取 1 级数据，uPayLoad 为 2 字节（第 1 个字节为 Q 值，取值范围 0~15，第 2 个字节为固定字节 0x20）；当 uOption=1 时，读取 2 级数据，uPayLoad 的内容包括第二级数据的存储区、起始地址、数据长度、Q 值（取值 0~15）、固定字节（0x20），其中存储区、起始地址、数据长度同第 1 级数据。

UhfStartReadDataFromMultiTag()函数用于开启防碰撞读取标签数据循环，该函数与UhfGetDataFromMultiTag()、UhfStopOperation()两个函数配合使用，UhfStartReadDataFromMultiTag()只是开启防碰撞读取标签数据循环，之后KLM将读到的数据上传到缓冲区，UhfGetDataFromMultiTag()用于从缓冲区读取一次存储区数据，开启防碰撞读取标签数据循环之后，模块只能响应UhfStopOperation()函数，UhfStopOperation()停止防碰撞读取标签数据循环。

4.18 UhfGetDataFromMultiTag ()

4.18.1 功能简介

该函数用于获取防碰撞读取标签数据所返回的数据。

4.18.2 函数原型

int WINAPI UhfGetDataFromMultiTag (HANDLE hCom, UCHAR* uStatus, UCHAR* ufData_len, UCHAR* ufReadData, UCHAR* usData_len, UCHAR* usReadData, UCHAR* uUii, UCHAR* uLenUii, UCHAR* uErrorCode);

4.18.3 返回值

1：获取数据成功；

其他：获取数据失败；

4.18.4 输入参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* uStatus：返回参数，返回当前操作状态值（1字节）；

UCHAR* ufData_len：返回参数，第一级数据长度，以字节为单位；

CHAR* ufReadData：返回参数，第一级数据内容；

UCHAR* usData_len：返回参数，第二级数据长度，以字节为单位；

CHAR* usReadData：返回参数，第二级数据内容；

UCHAR* uUii：返回参数，被读取标签的UII；

UCHAR* uLenUii：返回参数，标签UII的长度，字节为单位；

UCHAR* uErrorCode：返回参数，返回的错误代码（1字节）。

5. 固件升级命令集

5.1 UhfUpdateInit ()

5.1.1 功能简介

该函数用于通知读写器准备进行升级操作。

5.1.2 函数原型

int WINAPI UhfUpdateInit (HANDLE &hCom, char* cPort, UCHAR* STATUS, UCHAR* RN32, UCHAR flagCrc);

5.1.3 返回值

1：操作成功；

其他：操作失败；

5.1.4 参数

HANDLE &hCom：返回参数，通信端口句柄；

char* cPort：输入参数，串口（例如 \\?\hid#vid_0483&pid_5750#6&20bfa97&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}等）；

UCHAR* STATUS：返回参数，读写器响应返回的状态值（1 字节）；

UCHAR* RN32：返回参数，读写器响应返回的 RN32 数据（4 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

5.2 UhfUpdateSendRN32 ()

5.2.1 功能简介

该函数用于向读写器发送 RN32 数据的反码。

5.2.2 函数原型

int WINAPI UhfUpdateSendRN32 (HANDLE hCom, UCHAR* RN32, UCHAR* STATUS, UCHAR flagCrc);

5.2.3 返回值

1：操作成功；

其他：操作失败；

5.2.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* RN32： 输入参数，RN32 反码（4 字节）；

UCHAR* STATUS：返回参数，读写器响应返回的状态值（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

5.3 UhfUpdateSendSize ()

5.3.1 功能简介

该函数用于向读写器发送升级数据包数据长度。

5.3.2 函数原型

int WINAPI UhfUpdateSendSize (HANDLE hCom, UCHAR* STATUS, UCHAR* FILESIZE, UCHAR flagCrc);

5.3.3 返回值

1：操作成功；

其他：操作失败；

5.3.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* STATUS：返回参数，读写器响应返回的状态值（1 字节）；

UCHAR* FILESIZE：输入参数，升级数据包的数据包长度（4 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

5.4 UhfUpdateSendData ()

5.4.1 功能简介

该函数用于向读写器发送升级数据包。

5.4.2 函数原型

int WINAPI UhfUpdateSendData (HANDLE hCom, UCHAR* STATUS, UCHAR PACKNUM, UCHAR LASTPACK, int Data_len, UCHAR* TRANDATA, UCHAR flagCrc);

5.4.3 返回值

1：操作成功；

其他：操作失败；

5.4.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* STATUS：返回参数，读写器响应返回的状态值（1 字节）；

UCHAR PACKNUM：输入参数，数据包序号；

UCHAR LASTPACK：输入参数，最后一包升级数据包标志：0-该数据包不是最后一包数据包，1-最后一包数据；

int Data_len：输入参数，本次发送数据包数据长度；

UCHAR* TRANDATA：输入参数，数据包数据；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

5.5 UhfUpdateCommit ()

5.5.1 功能简介

该函数用于通知读写器升级操作完成。

5.5.2 函数原型

int WINAPI UhfUpdateCommit(HANDLE hCom, UCHAR* STATUS, UCHAR flagCrc);

5.5.3 返回值

1：操作成功；

其他：操作失败；

5.5.4 参数

HANDLE hCom：输入参数，通信端口句柄；

UCHAR* STATUS：返回参数，读写器响应返回的状态值（1 字节）；

UCHAR flagCrc：输入参数，是否使用 CRC16 验证功能，

0：不使用 CRC 功能；

1：使用 CRC 功能。

6. 其他命令集

6.1 LockGenCode ()

6.1.1 功能简介

该函数生成锁定码。

6.1.2 函数原型

void LockGenCode(UCHAR bkill, UCHAR baccess, UCHAR buii, UCHAR btid, UCHAR buser, UCHAR* LockCode);

6.1.3 返回值

无；

6.1.4 输入参数

UCHAR bkill：输入参数，对 KillPwd 的锁定操作（1：锁定；2：解锁；3：永久锁定；4 永久解锁；其他：保持状态）；

UCHAR baccess：输入参数，对 AccessPwd 的锁定操作（取值同 bkill）；

UCHAR buii：输入参数，对 UII 区的锁定操作（取值同 bkill）；

UCHAR btid：输入参数，对 TID 区的锁定操作（取值同 bkill）；

UCHAR buser： 输入参数，对 USER 区的锁定操作（取值同 bkill）；

UCHAR* LockCode：返回参数，生成的锁定码；

6.2 UhfSearchHids ()

6.2.1 功能简介

该函数搜索 HID 的数量和序列号。

6.2.2 函数原型

int WINAPI UhfSearchHids (char* serials);

6.2.3 返回值

>0：HID 的数量；

其他：搜索失败；

6.2.4 输入参数

char* serials：返回参数，HID 的序列号（当搜索到多个 HID 时，序列号用“|”隔开，如 \\?\hid#vid_0483&pid_5750#6&20bfa97&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}）

附录 A：频率各参数的说明

参数名	参数类型	描述
bFreMode	UCHAR	频率工作模式，取值范围如下： 0：中国标准 (920-925MHz)； 1：中国标准 (840-845MHz)； 2：ETSI 标准； 3：定频模式(922MHz)； 4：用户自定义；
BFreBase	UCHAR	频率基数，取值如下： 0：50MHz； 1：125MHz；
bBaseFre	UCHAR*	起始频率，取值范围：840~960；
bChannNum	UCHAR	频道数，取值范围：1~16；
bChannSpc	UCHAR	频道带宽基数，取值如下： 当 bFreBase = 50 时，取值为：1~20； 当 bFreBase = 125 时，取值为：1~8；
bFreHop	UCHAR	跳频顺序方式，取值如下： 0：随机跳频； 1：从高往低顺序跳频； 2：从低往高顺序跳频； 其他：随机跳频；

起始频率、终止频率、频率基数、频道数、频道带宽基数之间的关系如下：

```
if(频率基数 == 0)
{
    终止频率 - 起始频率 = (频道数 - 1) * 频道带宽基数 * 0.050
}else{
    终止频率 - 起始频率 = (频道数 - 1) * 频道带宽基数 * 0.125
}
```

例如：频率范围为 920.625-924.375MHz
频率基数=1；
起始频率小数部分：0.625；
起始频率整数部分：920；
起始频率的尾数积数 : 0.625÷0.125 = 0x05;
920 的 2 进制表示为：11 1001 1000；

```
bBaseFre[0] = (byte)(11 1001 1000 >> 3);                    --->0x73
bBaseFre[1] = (byte)(11 1001 1000 << 5 | (0x05 & 0x1F));    --->0x05
```

```
终止频率 - 起始频率 = (频道数 - 1) * 频道带宽积数 * 0.125；
```

924.375 -920.625 = （频道数 - 1）* 频道带宽积数 * 0.125；

（频道数-1）* 频道带宽积数 = 30；

当“频道数”= 16，“频道带宽积数”= 2；

bFreMode = 4;
bFreBase = 1;
bFreBase = {0x73 , 0x05};
bChannNum = 16;
bChannSpc = 2;
bFreHop = 0;

附录 B：标签操作各参数的定义

参数类型	参数名称	参数定义
UCHAR*	uAccessPwd	标签的访问密码，长度为 4 的数组
UCHAR*	uKillPwd	标签的销毁密码，长度为 4 的数组
UCHAR	uBank	标签的存储区： 0x00：RESERVED 0x01：UII 0x02：TID 0x03：USER
UCHAR*	uPtr	存储区起始地址的偏移量
UCHAR	uCnt	字长
UCHAR*	uUii	标签 Uii 数据
UCHAR*	uLenUii	Uii 长度
UCHAR*	uReadData	读取的数据
UCHAR*	uWriteData	要写入的数据
UCHAR*	uWritedLen	已经写入的字长
UCHAR*	uStatus	返回状态，长度为 1 的数组
UCHAR*	uErrorCode	错误代码，长度为 1 的数组