

# UHF Library Function Interface

## Description

All the operation interface of UHF are packed in file of Maverick.dll, this file is under package of Demo, the functions as following description.

1. int MkOpenAndConnect()  
Function Description : open com port  
Return: Return as 0 when success
2. int MkCallInventory(byte nScanMode, ref int nTagNum, ref byte[] cTagInfo)  
Function Description : scann tag data  
Return: return as 1 when success, otherwise it's failed  
Parameter description: nTagNum is PC code, cTagInfo is the EPC data returned, the first byte is the length of EPC data
3. int MKcallSelectTag(byte byEPCLen, byte[] cTagInfo)  
Function Description : Select TAG UID  
Return: return as 0 when success  
Parameter description: byEPCLen is length of EPC code, cTagInfo is length of EPC code.
4. int MKcallReadFromTag(Byte uMemtype, Byte uAddress, Byte uDataLen, ref Byte[] upData)  
Function Description: Read data.  
Return: return value bigger than 0 when success, otherwise it's failed  
Parameter description: uMemtype is Date area (RESERVE,EPC,TID,USER); uAddress is data address; uDataLen is data length, upData is the data be read from Data area
5. int MKcallWriteToTag(Byte uMemtype, Byte uAddress, Byte[] upAccesPw, Byte uDataLen, Byte[] upData)  
Function Description : to read data  
Return: return as 1 when success, otherwise it's failed  
Function description: uMemtype is data area (RESERVE、EPC、TID、USER); uAddressis data address; uDataLen is data length, upData is the data to be written into
6. int MKcallLockUnlock(Byte uLockUn, Byte uMenSpace, Byte[] upAccessPassWord)

Function Description : Lock tag

Return: return as 1 when success, otherwise it's failed

7. int MKcallConfigGen2(byte link, byte linkFreq, byte code, byte Coding, byte sess, byte session, byte tre, byte trext, byte gen2, byte gen2qbegin, byte sens, byte Sensitivity)

Function Description : configure MKcallConfigGen2 protocol

Function description: the parameter bit of link, code, sess, tre, gen2, which each stands if need to write the parameter bit of linkFreq、Coding、session、trext、gen2qbegin, 1 stands Yes, 0 stands No,

LinkFreq: speed setup of that Tag's reverse link communication, default value is 0x06;

Coding: program method setup of Tag's reverse link communication, default value is 0x01;

Session: Tag communication option, default 0x00;

Trext: preamble and pre-leading note, default 0x01.

8. int MKcallInventoryRSSI(byte nScanMode, ref byte Rssi, ref byte[] cTagInfo)

Function Description : callInventoryRSSI Tag inventory with RSSI

Function Description: nScanMode 0 stands only inventory at one time, if it's 1, that means reader will continuously inventory lasting at 1 second.

Rssi, range from 0 ~255, the value bigger, the signer stronger.

cTagInfo is the EPC data returned, the first bit is the data length.

9. int MKcallPowerdownMode()

Function Description : wake up/break

Return: Return 0 when reader entering into Sleep mode, return 1 reader wake up.

10. int MkcallSetReadAddr(byte set\_or\_read, ref byte new\_ID)

Function Description : callSetReadAddr setup to read Reader Addr;

Function Description: parameter set\_or\_read (1 stands to set new Addr, others stands to read address) new\_ID(the new address need to be set, setup return as 0x00~0xFF)

Return: return 0 when setting succeed

11. int Mk6B\_callReadFromTag6B(byte[] uid, byte addr, byte data, byte len, ref byte[] upData)

Function Description : callReadFromTag6B, read ISO18000-6B tag data;

Function Description; uid- tag data read out, len: data length got in this time

Return: return 0 when reading succeed

12. int Mk6B\_callWriteToTag(byte[] uid, byte addr, byte data, byte len) Function Description :

callWriteToTag6B, write ISO18000-6B tag

Function Descriptionuid: the tag uid to be written uid,

addr: the head address of tag data to be written into

data: tag data to be written into

len: data length to be written into;

Return: return as 2 when success

13. int MKcallChangeFreq(ref byte[] nChangeFreq,byte Dmb) callChangeFreqFunction

Description : change READER frequency parameter

Function Description: nChangeFreq-- frequency value configured

Return: return as 2 when success