MR Reader SDK Development Guide

MR Reader SDK is a software development kit that use for user develop application program. SDK provide to user in dynamic-link library document form.

When user using MR reader to develop their own application development platform, user will be able to complete their application development in high efficiently and correctly based on MR SDK. SDK support Visual C++、VB、C++ Builder and Delphi different language development. Demo program source code of VC++ version is available now, please contact with our engineer if have any needs.

SDK development guide is a reference manual for user secondary development. After review this manual, user will be able to solve their problem in fast way during their development.

According to functions performance, the SDK function can be apart in four segments: reader management functions、ISO18000-6B tag operation functions、EPC GEN2 tag operation functions and tag data management functions.

Remark: In this development guide, there is only once description for case of different functions but same parameter.

1 Read/write device management functions

1.1CommOpen

Functions Description	apiStatus　DrfCommOpen (HANDLE * hCom, char *com_port)
Function	Open PC COMM port
Parameter	Com port: name hCom: COMM port handle pointer
Return Value	Success return 0，fail return not 0

Example	Open COMM Port 1
	if(Comm Open(&ComHandle,”COM1”) = = 0)
	{
	　……
	}

1.2 CommClose

Functions Description	apiStatus　CommClose (HANDLE hCom)
Function	Close PC COMM Port
Parameter	hCom: COMM Handle
Return Value	Success return 0，　Fail return not 0
Example	

1.3 SetBaudRate

Functions Description	apiStatus SetBaudRate(HANDLE hCom, USHORT BaudRate, unsigned char ReaderAddr)
Function	Set reader’s baud rate
Parameter	Baud Rate：Baud rate code. 0~4 separate symbolizes 9600、19200、38400、57600 and115200 Reader Address: use for fixed reader RS485, Default 0XFF（handheld reader and module is invalid parameter）
Return Value	Success return 0, fail return not 0
Example	

1.4 Reset

Functions Description	apiStatus　Reset (HANDLE hCom, unsigned char ReaderAddr)
Function	Reader reset
Parameter	

Return Value	Success return 0, fail return not 0
Example	

1.5 GetFirmwareVersion

Functions Description	apiStatus GetFirmwareVersion (HANDLE hCom, unsigned char *major,unsigned char *minor, unsigned char ReaderAddr)
Function	Read the reader's firmware version number
Parameter	major: major version information pointer Minor: minor version information pointer
Return Value	Success return 0，fail return not 0

Example	To read and output firmware version number
	if(GetFirmwareVersion(ComHandle,&ver1,&ver2,0xff) == 0)
	printf("Firmware Version is:%d.%d",ver1,ver2);

1.6 SetRf

Functions Description	apiStatus SetRf(HANDLE hCom,unsigned char power,unsigned char freq_type,unsigned char ReaderAddr)
Function	Set reader's power and frequency parameter
Parameter	Power value: dereference 0~30, corresponding 0~30dBm. Frequency type: dereference 0 as China standard（920M~925M）, dereference 1 as America standard（902M~928M）others are special type (i.e. 868M)，refer to special specification
Return Value	Success return 0，fail return not 0
Example	Set reader as America frequency, power configure as 0.5W（27dBm） SetRf(ComHandle,27,1,0xff);

1.7 GetRf

Functions Description	apiStatus GetRf(HANDLE hCom,unsigned char * power,unsigned char *freq_type,unsigned char ReaderAddr)
Function	Read the reader's current RF parameter
Parameter	Power: Power value pointer Frequency type: Frequency type pointer
Return Value	Success return 0，fail return not 0
Example	

2 ISO18000-6B Tag Operation Functions

2.1 IsoMultiTagIdentify

Functions	apiStatus IsoMultiTagIdentify(HANDLE hCom, unsigned int * Count,TagIds

Description	*value, unsigned char ReaderAddr)
Function	ISO18000-6B multi-tag identification contains repeat data filtration. Please use ClearIDBuffer functions to clearing reader's internal buffer before restart new operation of multi-tag identification.
Parameter	Count: Tags quantity that being read this time, value: Tags data that being read. Saved as TagIds structure. TagIds structure: Total 14 byte, the 1st byte is tags type（ISO18000-6B tag is 1, EPC GEN2 is 4）the 2nd byte is antenna port **(**specific to multi-port reader**).** The last 12 byte is the data of tag, UID of ISO18000-6B should be front 8 byte.
Return Value	Success return 0，fail return not 0

Example	Start multi-tag identify
	ClearIDBuffer（ComHandle,0xff）;
	While(non-stop event)
	{
	if(IsoMultiTagIdentify（ComHandle，&Cnt,TagData,0Xff）==0)
	{
	for(i = 0; i < Cnt; i++)
	OutputDate(TagData[i]);
	}
	}

2.2 IsoMultiTagRead

Functions Description	apiStatus IsoMultiTagRead(HANDLE hCom, unsigned char iRomAddr,unsigned int * Count,TagIds *value, unsigned char ReaderAddr)
Function	ISO18000-6B multi-tag reading: it's able to read the front 8 byte data of random address started.
Parameter	iRomAddr: read the tags start address, Count: tags quantity being read this time; value: the tag data being read, saved as TagIds structure
Return Value	Success return 0，fail return not 0
Example	

2.3 IsoWriteTag

Functions Description	apiStatus IsoWriteTag(HANDLE hCom ,unsigned char iRomAddr,unsigned char value, unsigned char ReaderAddr);
Function	ISO18000-6B tag's write: one time write one byte data
Parameter	iRomAddr: Tags store address to be written. Value: data to be written.
Return Value	Success return 0, fail return not 0

Example	Write 0xAA in tag address 20
	if(IsoWriteTag(ComHandle,20,0xAA,0Xff) == 0)
	printf("write success");
	else
	printf("write failed");

2.4 IsoReadWithID

Functions Description	apiStatus IsoReadWithID(HANDLE hCom, unsigned char* TagID,unsigned char iRomAddr,unsigned char *AntNum,unsigned char* value, unsigned char ReaderAddr);
Function	Read tag data that specified ID. Read the specified address started 8 byte data of each reading.
Parameter	TagID: 8 byte UID; iRomAddr, read first address; AntNum, read the tag serial number. Value, read the tag data
Return Value	Success return 0, fail return not 0
Example	

2.5 IsoWriteWithID

Functions Description	apiStatus IsoWriteWithID(HANDLE hCom, unsigned char* TagID,unsigned char iRomAddr,unsigned char value, unsigned char ReaderAddr)
Function	Write the data to specified ID tag.
Parameter	Tag ID: 8 byte UID; iRomAddr，write address; value: write data
Return Value	Success return 0, fail return not 0

Example	Write 0xAA to specified tag address 30
	unsigned char Id[8] = {0xE0,0x04,0x00,0x00,0xB5,0x73,0x8C,0x01}
	if(IsoWriteWithID (ComHandle, Id,30,0xAA,0Xff) == 0)
	printf(“write success”);
	else
	printf(“write fail”);

2.6 IsoLockTag

Functions Description	apiStatus IsoLockTag(HANDLE hCom ,unsigned char iRomAddr, unsigned char ReaderAddr)
Function	To write and lock to the specified tag address, this address can’t be unlock after lock.
Parameter	iRomAddr, tag address to be write and lock
Return Value	Success return 0, fail return not 0
Example	

2.7 IsoQueryLock

Functions Description	apiStatus IsoQueryLock(HANDLE hCom, unsigned char iRomAddr,unsigned char *status, unsigned char ReaderAddr)
Function	Inquiry specified address locked or not
Parameter	iRomAddr: inquiry address, status, return status, 0 is un-lock，1 is locked
Return Value	Success return 0, fail return not 0
Example	

2.8 IsoBlockWrite

Functions Description	apiStatus IsoBlockWrite(HANDLE hCom, unsigned char iRomAddr, unsigned char length，unsigned char* value, unsigned char ReaderAddr);
Function	Write 4 byte content to specified address continuously
Parameter	iRomAddr, write address, only the multiple of 4, Value:write data, 4 byte
Return Value	Success return 0，fail return not 0
Example	

3 EPC GEN2 Tag Operation Functions

3.1 Gen2MultiTagIdentify

Functions Description	apiStatus Gen2MultiTagIdentify(HANDLE hCom, unsigned int * Count,TagIds *value, unsigned char ReaderAddr)
Function	EPC GEN2 multi-tag Identification contains repeat data filtration.
Parameter	Count: tag quantity being identified this time; value: tag data
Return Value	Success return 0，fail return not 0
Example	Refer to IsoMultiTagIdentify application example

3.2 Gen2WriteEPC

Functions Description	apiStatus Gen2WriteEPC(HANDLE hCom ,unsigned char WordPtr,unsigned int value, unsigned char ReaderAddr)
Function	EPC GEN2 tag: EPC write data, write 1 word (2byte) data on each time.
Input parameter	WordPtr, write word serial number. Value: write content.
Return Value	Success return 0, fail return not 0

Example	Write 0x55AA to EPC 4 byte.
	if(Gen2WriteEPC (ComHandle,4,0x55AA,0Xff) == 0)
	printf(“write sucess”);
	else
	printf(“write fail”);

3.3 Gen2LockTag

Functions Description	apiStatus Gen2LockTag(HANDLE hCom,unsigned char MemBank = 1, unsigned char ReaderAddr)
Function	Write and lock operation to EPC tag, write and lock one area each time.
Input parameter	MemBank area of write and lock, 0 is reservation, 1 is EPC, 2 is TID, 3 is user area
Return Value	Success return 0, fail return not 0
Example	

3.4 Gen2KillTag

Functions Description	apiStatus Gen2KillTag(HANDLE hCom ,unsigned int PassWord, unsigned char ReaderAddr)
Function	EPC GEN2 Tag's Kill
Input parameter	PassWord,32 bit kill password, the tag can't be kill if the tag password is 0
Return Value	Success return 0, fail return not 0
Example	

3.5 Gen2InitEPC

Functions Description	apiStatus Gen2InitEPC(HANDLE hCom, unsigned char WordCount = 6, unsigned char ReaderAddr)
Function	EPC tag length initialized, Normally, Initialized is 96 bit（6 word）
Input parameter	WordCount**,** initialized number(1 word is 2 byte)
Return Value	Success return 0, fail return not 0
Example	Initialized is 96 bit： Gen2InitEPC（ComHandle,6,0Xff）;

3.6 Gen2Read

Functions Description	apiStatus Gen2Read(HANDLE hCom,unsigned char Membank,unsigned char WordPtr,unsigned char WordCnt, unsigned char *value, unsigned char ReaderAddr)
Function	EPC tag's reading
Input parameter	Membank: read area; WordCnt: read wordcount;　alue:read data
Return Value	Success return 0，fail return not 0

Example	Read 4 byte TID of tag
	unsigned char value[4];
	if(Gen2Read(ComHandle,2,2,&value,0xff) == 0)
	{
	　　Printf("TID:%x%x%x%x",value[0],value[1],value[2],value[3]);
	}

3.7 Gen2Write

Functions Description	apiStatus Gen2Write(HANDLE hCom ,unsigned char Membank,unsigned char WordPtr,unsigned int value, unsigned char ReaderAddr)
Function	EPC GEN2 tag single word write
Input parameter	Membank: Write area; WordPtr: Write address; Value: Write data（2 byte）

Return Value	Success return 0，fail return not 0
Example	

3.8 Gen2BlockWrite

Functions Description	apiStatus Gen2BlockWrite(HANDLE hCom ,unsigned char Membank,unsigned char WordPtr,unsigned char WordCnt,unsigned char *value, unsigned char ReaderAddr)
Function	EPC GEN2 tag block write
Parameter	Membank，write area; WordPtr: write address; WordCnt: write wordcount (max. value depend on tag supporting length, commonly max. support 2 word write)；Value: write data
Return Value	Success return 0，fail return not 0

Example	Write 4 byte data into tag's user area
	unsigned char value[4] = {0x01,0x02,0x03,0x04};
	if(Gen2BlockWrite (ComHandle,3,0,2,value,0xff) == 0)
	{
	Printf("data write sucess");
	}

4 Tag data management function

The segment indicates "internal use" may have been embedded in other functions or internal test use, user don't have to care about it.

4.1 GetTagData

Functions Description	apiStatus GetTagData(HANDLE hCom,int Count,TagIds *value, unsigned char ReaderAddr);
Function	Obtain tag data from reader's buffer store area（internal use）
Parameter	Count: groups of tag data; Value: return tag data
Return Value	Success return 0，fail return not 0
Example	

4.2 ClearIDBuffer

Functions Description	apiStatus ClearIDBuffer(HANDLE hCom,unsigned char ReaderAddr);
Function	Clear readers tag data buffer area，it able to used before multi-tag identification.
Parameter	
Return Value	Success return 0，fail return not 0
Example	Refer to example 2.1

4.3 QueryIDCount

Functions Description	apiStatus QueryIDCount(HANDLE hCom,unsigned char* Count,unsigned char ReaderAddr);
Function	Inquiry tag quantity when in caching status
Parameter	Count, return tag quantity pointer
Return Value	Success return 0，fail return not 0
Example	

4.4 GetIdWithoutDel

Functions Description	apiStatus GetIdWithoutDel(HANDLE hCom,unsigned char *value,unsigned char ReaderAddr);
Function	Take one group data of tag, but not delete data（internal use）
Parameter	Value: obtain the tag data
Return Value	Success return 0，fail return not 0
Example	

4.5 GetIDACK

Functions Description	apiStatus GetIDACK(HANDLE hCom,unsigned char ReaderAddr)
Function	Confirm after get data, reader will delete the data that obtain previous（internal use）
Parameter	
Return Value	Success return 0，fail return not 0
Example	